

**O'ZBEKISTON RESPUBLIKASI
OLIY VA O'RTA MAXSUS TA'LIM VAZIRLIGI**

**ISLOM KARIMOV NOMIDAGI TOSHKENT
DAVLAT TEXNIKA UNIVERSITETI
TERMIZ FILIALI**

“ANIQ VA TABIIY FANLAR” KAFEDRASI



**“KIMYO INJINERINGIDA AXBOROT
TEXNOLOGIYALARI”
fanidan**

O'QUV-USLUBIY MAJMUA

Termiz – 2020

O‘ZBEKISTON RESPUBLIKASI
OLIY VA O‘RTA MAXSUS TA‘LIM VAZIRLIGI
TOSHKENT DAVLAT TEXNIKA UNIVERSITETI
TERMIZ FILIALI

Ro‘yxatga olindi
№ _____
« _____ » _____ 20 ____ yil

«T A S D I Q L A Y M A N»
O‘quv va tarbiyaviy ishlari bo‘yicha
direktor o‘rinbosari

F.Nosirov
20 ____ yil « _____ » _____

“Aniq va tabiiy fanlar” kafedrası

Kimyo injineringida axborot texnologiyalari
FANIDAN

O‘QUV-USLUBIY MAJMUUA

Bilim sohasi:	300000	– Ishlab chiqarish- texnik soha
Ta‘lim sohasi:	320000	– Ishlab chiqarish texnologiyalari
Ta‘lim yo‘nalishi:	5320400	–Kimyoviy texnologiya (silikat materiallari)
	5320400	–Kimyoviy texnologiya (kamyob, tarqoq va nodir metallar)

Termiz

O'quv-uslubiy majmua O'zbekiston Respublikasi Oliy va o'rta maxsus ta'lim vazirligida № _____ raqam bilan ro'yhatga olingan va 2019 yil "___" _____ da _____ - sonli buyruq bilan tasdiqlangan fan dasturi asosida tuzilgan.

Tuzuvchi(lar): – S.M.Abdullayeva – Toshkent davlat texnika universiteti
Termiz filiali « Aniq va tabiiy fanlar »
kafedrası assistenti;

Kafedra mudiri:

O.Sultonova

Fanning ishchi o'quv dasturi "Yengil sanoat va kimyo texnologiyalari" fakultet kengashida muhokama etilgan va o'quv-uslubiy Kengashda muxokama qilish uchun tavsiya qilingan (20__ yil "___" _____ - sonli bayonnoma)

Fakultet dekani: _____ **J.Raxmonqulov**
(imzo)

Fanning ishchi o'quv dasturi Toshkent davlat texnika universiteti Termiz filialining o'quv-uslubiy Kengashida ko'rib chiqilgan va tavsiya etilgan. (20__ yil "___" _____dagi _____-sonli bayonnoma)

O'quv-uslubiy Kengash kotibi: _____ **O.Qarshiyev**
(imzo)

**O‘ZBEKISTON RESPUBLIKASI
OLIY VA O‘RTA MAXSUS TA‘LIM
VAZIRLIGI**

**TOSHKENT DAVLAT TEXNIKA UNIVERSITETI
TERMIZ FILIALI**

**KIMYO INJINERINGIDA
AXBOROT TEXNOLOGIYALARI
o‘quv fanidan**

MA‘RUZA MASHG‘ULOTLARI

1-ma'ruza. Kimyo injinerida axborot texnologiyalar faniga kirish.

Reja:

- 1. Kimyoviy injinerida axborot texnologiyalari.**
- 2. Axborot texnologiyasi tushunchasi.**
- 3. Axborot texnologiyasining asosiy tavsifi.**
- 4. Axborot texnologiyalari taraqqiy etishining asosiy bosqichlari va rivojlanish tendensiyasi.**
- 5. O'zbekiston Respublikasining axborotlashtirish dasturi.**
- 6. Axborot va uning xossalari.**

Axborot texnologiyalari fanini o'qitishdan maqsad, zamonaviy axborot texnologiyalar haqida tushuncha berish, zamonaviy dasturiy vositalar, zamonaviy kompyuter dasturlarida ishlashni bilishni o'rgatish va ta'lim jarayoniga mos o'quv resrularini kompyuterdan foydalanib, yaratish. Ta'lim jarayoniga mos elektron darsliklarini, o'quv elektron resrularni, on-layn tizimida va off-layn tizimida ishlaydigan vositalarni yaratishni o'rgatishdan iboratdir. Bu fan o'zining oldiga bir necha vazifalarni qo'yib olgan, ya'ni birinchi istiqbol rejalaridan birinchisi, yaratilgan va yaratilayotgan yangi kompyuter dasturlaridan xabardor bo'lib turish, mavjudlarini amaliyotda qo'llashni o'rgatish.

Axborot kommunikatsion texnologiyalar-bu audio, video va kommunikatsiya vositalarini axborotlarni hosil qilish, olish, ishlab chiqish va qayta ishlash hamda tarqatish jarayonidir. Bu jarayonda inson resurlari va texnik resurlar qatnashadi. Biri ikkinchilari uchun xizmat qiladi. Ya'ni texnik resurlar insonlarni axborotni olishlari uchun va boshqa odamlarga tarqatish uchun xizmat qiladi.

Texnologiya-maxorat, san'at degan ma'noga ega, logos esa-ta'lim, — Axborot texnologiyalari kursining maqsadi AKTdan ta'limda, jumladan, xorijiy til va adabiyoti fanini o'qitishda qo'llashni o'rgatish, talabalarning esa ta'limda qo'llaniladigan dasturlar, dasturiy vositalardan o'z mutahassisliklarida foydalanishni bilish ta'limda qo'llashni bilish, ularni o'rganish va foydalanish hamda qo'llashning zamonaviy tendensiyalardan biri, bu axborot texnologiyalarining turlari bilan tanishish, ularning qaysi sohaga oidligini farqlay bilish va ilg'or texnologiyalarni, bir tildan ikkinchi tilga o'girish texnologiyalari, matndan giper-matnga o'girish texnologiyalarini, taqdimot yaratish texnologiyalarini, ekspert tizimlar texnologiyalari, elektron xukumat, masofaviy ta'lim texnologiyalari, video-kurslar, elektron resurslar, on-layn taqdimotlar va boshqa ta'lim resurlarini yaratuvchi dasturlar va dasturiy vositalarni o'rganish maqsadga muvofiqdir. Ta'lim jarayoniga axborot kommunikatsion texnologiyalarini qo'llash ta'lim jarayonini mazmunli, chiroyli, unumli, qulay, vaqtni tejovchi va o'zlashtirish sifatini oshiruvchi etib amalga oshirib beradi. O'qituvchining dars berish jarayonini unumli, vaqtni tejovchi va mazmunli tarzda o'tkazib beradi. O'zlashtirish sifatida vaqtning qo'llanish evaziga, ya'ni kompyuter dasturlari qo'shimcha kanallari orqali ko'z xotirasini qabul qilish jarayonini ishga tushirib, ularga ta'sirni kuchytirib, rangli rasmlar, xarakatli animatsiyalarni o'qitish jarayoniga tadbiq etib, tasavvurni to'g'ri shakllantirish orqali bilish jarayonini, tafakkur etish qobiliyatini rivojlantiradi.

Ta'lim jarayoniga axborot kommunikatsiya texnologiyalarini qo'llashning dolzarb vazifalaridan biri, bu elektron darsliklarni yaratuvchi, elektron resurlarni, taqdimotlarni, zamonaviy ta'lim vositalarini yaratuvchi dasturlarni o'rgatishdan iborat. Axborot katta jadallik bilan kirib kelmoqda, kim axborot bilan birinchi tanish bo'lsa, o'sha voqeani boshqaradi, jarayonni boshqarish uni qo'lida, ya'ni u dunyoni boshqaradi, deyishadi.

Katta xajmdagi axborotlarni yetkazish, uni samarali egasiga berish, vaqtni tejash vazifasini bajarishda axborot texnologiyalar yordamga keladi. Konsepsiya-bu ta'limot deganidir.

1. AKT. O'zbekistonda AKT sohasini rivojlantirishdagi davlat siyosatining asosiy yo'nalishlari, amaldagi qonunlar, farmonlar va qarorlar.

2. Axborot va uning turlari, xossalari, uzatilishi, o'lchov birliklari.

3. Axborotni kodlash.

4. Ma'lumotlarning kompyuter xotirasida tasvirlanishi/ O'zbekistonda AKT sohasini rivojlantirish uchun 2005 yidagi bir qator qarorlar va qonunlar isbot bo'la oladi. 21 asr axborot texnologiyalari asri bo'lgani uchun ta'limni ham bu texnologiyalarisiz tasavvur etish qiyin. Hukumatimiz tomonidan Axborotlashtirish to'g'risidagi, Elektron hukumat tashkil etish, uni rivojlantirish haqidagi bir qator qonunlar jamiyatni rivojlantirish, hukumatni boshqarishga yordam qilishga yo'naltirgan.

1.2. Axborot texnologiyasi tushunchasi.

Texnologiyalar. —Texnologiya grekcha so'z bo'lib mohirlik, ustalik, biror ishni uddalay olishni anglatadi. Bu ma'lum bir jarayonga nisbatan qo'llanilgan. Jarayon deganda esa maqsadga erishishga yo'naltirilgan xatti-harakatlar majmui tushunilgan. Ushbu jarayon inson tomonidan tanlangan strategiya bilan belgilanadi va turli xildagi vositalar, usullar yordamida amalga oshiriladi. Umumiy hollarda texnologiya deganda, mahsulotni ishlab chiqarish jarayonida amalga oshiriladigan xomashyo, material yoki yarim tayyor mahsulot shakli, xususiyati, holatining o'zgarishi, uni qayta ishlash, tayyorlash usullarining majmui tushuniladi. Bu biror bir ishni yuqori darajada uddalash deganidir. Axborot texnologiyalari to'g'risida gap ketganda, material sifati ham, mahsulot sifatida ham axborot ishtirok etadi. Biroq bu ob'ekt, jarayon yoki hodisa to'g'risidagi sifat jihatidan yangi ma'lumot bo'ladi. Texnologiya xodimning axborot bilan ishlash usuli va uslubi hamda texnik vositalar orqali namoyon bo'ladi.

Axborot texnologiyasi – ob'ektning (axborot mahsulotining) holati, jarayon yoki voqeaning yangi xususiyati to'g'risida axborot olish uchun ma'lumotlarni yig'ish, qayta ishlash va uzatish vositalari va usullari majmuidan foydalaniladigan jarayondir.

Moddiy ishlab chiqarish texnologiyasining maqsadi – inson yoki tizimning ehtiyojini qondiruvchi mahsulot ishlab chiqarish sanaladi. Axborot texnologiyasining maqsadi esa – axborot ishlab chiqarish bo'lib, uni tahlil etish va uning asosida biror bir harakatga qo'l urish uchun tegishli qaror qabul qilish.

Axborotlarni yig'ish, uzatish, to'plash, saqlash, taqdim etish va foydalanish uslublari va usullari tizimi **axborot texnologiyasi** deb yuritiladi. Axborot texnologiyasi avtomatlashgan va an'anaviy (qog'oz) ko'rinishiga amalga oshiriladi. Avtomatlashtirish hajmi va texnik vositalardan foydalanish turi aniq bir texnologiyaning mohiyatiga bog'liq.

Avtomatlashtirish – bu inson ish faoliyatini mashina va mexanizmlar bilan almashtirish demakdir. U texnik, iashkiliy va iqtisodiy mazmundagi hattharakatlar hamda tadbirlar kompleksidan iborat bo'ladi, ishlab chiqarish jarayoni, boshqaruv jarayonining u yoki bu ishini amalga oshirishda inson ishtirokini qisman yoki butunlay cheklash imkonini beradi.

Moddiy va axborot texnologiyasining asosiy komponentlarini qiyoslash
1-jadvalda berilgan.

1-jadval.

Mahsulot ishlab chiqarish uchun texnologiya komponentlari	
Moddiy mahsulot	Axborot mahsuloti
Xom ashyo va materiallar tayyorlash	Ma'lumotlar yoki boshlang'ich axborotni yig'ish
Moddiy mahsulot ishlab chiqarish	Ma'lumlarni qayta ishlash va yakuniy axborotga ega bo'lish
Iste'molchilarga ishlab chiqarilgan mahsulotni sotish	Uning asosida qaror qabul qilish uchun yakuniy axborotlar uzatish

1.3.Axborot texnologiyasining asosiy tavsifi.

Avtomatlashtirilgan axborot texnologiyasi (AAT) – boshqaruv vqazifalarini hal etish uchun tizimli tashkil etilgan axborot jarayonlarini amalga oshirish usul va vositalari majmuidir. U hisoblash texnikasi va aloqa vositalaridan foydalaniladigan rivojlangan dasturiy ta'minotni qo'llash bazasida bajariladi. Axborot texnologiyasining amalga oshirishdagi texnik vositalarning asosiy qisini komp yuter texnikasi tashkil etgan uchun axborot texnologiyasi, ayniqsa zamonaviy axborot texnologiyasi deganda komp yuter axborot texnologiyasi tushuniladi.

Zamonaviy axborot texnologiyasi (ZAT) (kompyuter axborot texnologiyasi) – shaxsiy komp yuter va telekommunikatsiya vositalaridan foydalanuvchi uchun qulay —interfeyslli axborot texnologiyasidir. Ma'lum bir turdagi komp yuter uchun mo'ljallangan bir yoki bir necha o'zaro bog'liq dasturiy mahsulotlar zamonaviy axborot texnologiyalarining vositasi sanaladi. Zamonaviy axborot texnologiyasining asosiy elementlari.

ZAT ning asosiy elementlari quyidagicha:

- ma'lum bir vaqt davomida axborotni kiritish va qayta o'zgartirish;
- tasvirni kiritish va unga ishlov berish;
- signal axboroti paydo bo'lgan yerda uni qayta ishlash;
- og'zaki axborotni qayta ishlash;
- foydalanuvchining ShK bilan faol muloqoti;
- turli axborot tizimlarida mashinali modellashtirish;

- axborot almashuvining tarmoq texnologiyasi (dialog yuritish, video va telekommunikatsiya, elektron pocha, vidioteka va x.z.);
- taqsimlangan tarmoq tizimlarida ma'lumotlarni mul'tiprotssessor asosida qayta ishlash;
- mahalliy, mintaqaviy va xalqaro tarmoqlar bo'yicha axborotni tezkor tarqatish.

Axborot texnologiyasining bazaviy texnologiyasi bo'lib quyidagilar hisoblanadi:

- a) texnik ta'minot texnologiyasi;
- b) telekommunikatsiya texnologiyasi;
- v) dasturiy ta'minot texnologiyasi.

Bu texnologiyalar hisoblash tizimlari va tarmoqlari arxitekturasining aniq variantlari doirasida birgalikda harakat qiladi va birlashadi. Ularni ayrimlari axborot texnologiyasi rivojlanishida hal qiluvchi rol o'ynaydi.

1.4.Axborot texnologiyalari taraqqiy etishining asosiy bosqichlari va rivojlanish tendensiyasi.

XIX asrning ikkinchi yarmigacha "qo'l" axborot texnologiyasi rivojlangan bo'lib, uning asosini pero, siyohdon va buxgalteriya daftari tashkil etgan. Kommunikatsiya (aloqa) paket (rasmiy xujjatlar solingan konvert) yuborish orqali amalga oshirilardi.

"Qo'l" axborot texnologiyasi o'rniga XIX asr oxirida "mexanik" texnologiya kirib keldi. Yozuv mashinasi, telefon, diktafonning kashf etilishi, jamoa pochasi tizimining takomillashuvi – bular bari avvaliga axborotni qayta ishlash texnologiyasida, so'ng ish mahsuldorligida sezilarli o'zgarishlar yuz berishiga zamin bo'ldi.

XX asrning 40 - 60–yillarida —elektrli texnologiyasi paydo bo'lib, u yechib almashtiriladigan elementlarga ega elektr yozuv mashinkalari, oddiy qog'ozdan foydalanuvchi nusxa ko'chirish mashinasi, portativ diktafonlardan iborat edi. 60-yillarning ikkinchi yarmidan esa —elektron (yoki kompyuter)texnologiyasi yuzaga kela boshladi va axborotni shaklini emas, mazmunini o'zgartirishga urg'u berila boshlandi.

Axborot texnologiyasining keyingi taraqqiyoti asosan kommunikatsiya vositalari bilan bog'liq.

Xorijiy mutaxassislar axborot texnologiyalari rivojlanishiing beshta asosiy tendentsiyasini ajratib ko'rsatadi:

1. Axborot mahsulotlarining murakkablashuvi. Axborot vositasi ko'rinishidagi axborot mahsuloti, ekspert ta'minoti txizmatining ma'lumotlar bazasi strategik ahamiyat kasb eta boradi. Turli shakldagi (nutq, ma'lumot, tasvir) axborot mahsulotlari eshitish, ko'rish va anglash uchun foydalanuvchining talabiga ko'ra ishlab chiqiladi hamda unga qulay vaqtda va shaklda mahsulotni yetkazib berish vositasi mavjud bo'ladi.

2. Birgalikda harakat qilish qobiliyati. Axborot mahsulotining ahamiyati oshib borishi bilan mazkur mahsulotlarni komp'yuter va inson yoki axborot tizimlari o'rtasida ideal tarzda almashuvini o'tkazish imkoniyati ilg'or texnologik

muammo kasb etadi.

3. Oraliq bo'g'inlarni tugatish. Birgalikda harakatlanish qobiliyatining rivojlanishi axborot mahsulotlari almashish jarayonining takomillashuviga, so'ngra, axborot manbai yo'lidan iste'molchiga qarab oraliq bo'g'inlar tugatiladi.

4. Globallashtirish. Tashkilot yo'ldosh aloqa va Internet tarmog'ida foydalanib axborot texnologiyalari yordamida hohlagan joyda va hohlagan paytda ish olib borish mumkin.

5. Konvergenstsiya (uyg'unlashish). Konvergenstsiya AATning zamonaviy rivojlanish jarayonining oxirigi bosqichi sifatida ko'rib chiqiladi. Bunda mahsulotlaro va xizmatlar, axborot va dam olish, shuningdek, ovozli, raqamli hamda videosignallarni uzatish kabi ish rejimlari o'rtisidagi farq yo'qoladi.

1.5.O'zbekiston Respublikasining axborotlashtirish dasturi.

Kompyuter texnikasi va axborot texnologiyalari rivojlanishi turli xil axborotlardan foydalanishga qaratilgan va axborot jamiyati nomini olgan jamiyatning rivojlanish omili bo'ldi, ya'ni axborot ishlab chiqarish jamiyatni rivojlantirishda harakatlantiruvchi kuchna aylandi. Axborot texnologiyalari bugungi kunda hayotimizning hamma sohalarini qamrab olgan.

O'zbekistonda axborot texnologiyasi rivojlanishini tezlashtirish va zamonaviy darajaga ko'tarish uchun Respublika Vazirlar Mahkamasi tomonidan bir qator qarorlar qabul qilingan. 1994 yilda Vazirlar Mahkamasi O'zbekiston Respublikasini axborotlashtirish kontseptsiyasini qabul qilgan. Kontseptsiya asosida —O'zbekiston Respublikasini axborotlashtirish Dasturi ishlab chiqilgan. Dastur uch maqsadni o'z ichiga oladi:

- 1) milliy axborot hisoblash tarmog'i;
- 2) EHM ni matematik va dasturiy ta'minlash;
- 3) shaxsiy kompyuter.

Milliy axborot hisoblash tarmog'i davlat aloqa tizimi negizida ishlaydigan va yagona qoidalarga rioya qilish asosida qurilgan davlat va idoraviy xususiyatga ega bo'lgan axborot hisoblash tarmoqlarini mujassamlashgan ochiq tizimdir. EHM ni matematik va dasturiy ta'minlash axborotlashtirish masalalarining matematik modeli, hisoblash algoritmi va yechim olish dasturini yaratishdan iboratdir.

Dasturning shaxsiy komp yuter qismi informatika industriyasini yaratish va rivojlantirishdan iborat.

Istalgan biror ob'ektni yoki jarayonni boshqarish uchun oldindan tayyorlangan axborotlardan foydalaniladi, ya'ni axborot tizimi yaratiladi. Axborot tizimi boshqaruv funktsiyasini amalga oshirish uchun turli xil axborotlarni yig'ish, uzatish va qayta ishlash bo'yicha ma'lumotlar va kommunikatsion tizimdir. Axborot tizimining texnik asosi bo'lib kompyuterlar hisoblanadi.

Informatika 1960 – yillarda Frantsiyada EHM yordamida axborotlarni qayta ishlash bilan shug'ullanuvchi sohani ifodalovchi atama sifatida yuzaga keldi. Informatika atamasi lotincha —informatik so'zidan kelib chiqqan bo'lib, tushuntirish, xabar qilish, bayon etish ma'nolarini anglatadi. Informatikaning fan sifatida ajralib chiqishi birinchi navbatda komp yuter

texnikasining rivojlanishi bilan uzviy bog'liqdir. Informatika inson faoliyatining turli jabhalardagi axborotlarni izlash, to'plash, qayta ishlash qonuniyatlarini EHM tizimlariga asoslangan holda o'rganuvchi fandir.

Informatikaning asosiy uchta yo'nalishi mavjud.

Birinchi yo'nalish axborotni uzatish, yig'ish va qayta ishlashning texnik vositalarini rivojlantirish nazariyasi bilan bog'liq. U o'z ichiga hisoblash komplekslarini, lokal va global hisoblash tarmoqlari, aloqa nazariyasini olgan keng ilmiy-ommaviy sohadir.

Ikkinchi yo'nalish ma'lumotlarni qayta ishlash bo'yicha har xil amaliy vazifalarni hal etish yuzasidan turli kategoriyadagi foydalanuvchilar uchun texnik vositalar bilan samarali ishlashni tashkil qilish imkonini beradigan, dasturiy ta'minotni ishlab chiqishga yo'naltirilgan, matematik va amaliy fanlar kompleksini o'z ichiga olgan dasturlashtirishdir.

Bu yo'nalishga algoritmlashtirish tillari nazariyasi, ma'lumotlarni tashkil etish, saqlash, qidirish va qayta ishlash nazariyasi, tizimli hamda amaliy dasturlashtirish nazariyasi kiradi.

Uchinchi yo'nalish – avtomatlashtirilgan usulda turli darajadagi vazifalarni hal etish modellari, algoritmlari, tartibi, texnologiyasini ishlab chiqish va tashkil qilishdir.

Informatika o'zaro aloqador uch qismdan iborat: algoritmik vositalar, dasturiy vositalar va texnik vositalar.

Informatikaning moddiy-texnik bazasi boshqaruv faoliyatining turli sohasi va darajalarida ishlayotgan mutaxassislar uchun avtomatlashtirilgan ish joylaridan keng foydalanish imkonini beradi va shuningdek, ekspert – professional (ekspert tizimi) darajasida qaror qabul qilishga qodir bo'lgan ixtisoslashtirilgan predmet sohasida hisoblash tizimini va axborot-kommunikatsiya tarmog'ini yaratish imkonini yuzaga keladi.

Informatika fanining *asosiy ashyosi* axborotdir. Axborot atrof-muhit, ob'ektlar va hodisalar, ularning o'lchamlari, xosiyatlari va holatlari to'g'risidagi ma'lumotdir. Ma'lumot bilan axborot bir-biridan farq qiladi. Axborot foydalaniladigan ma'lumotdir.

Informatikaning tarkibiy qismi bo'lib *axborot texnologiyasi* va *telekommunikatsiya* hisoblanadi.

1.6. Axborot va uning xossalari.

Axborot texnologiyasi ob'ekt, jarayon yoki hodisa haqida yangi sifat axboroti olish uchun ma'lumotlar yig'ish, qayta ishlash va uzatish (boshlang'ich axborot) vositasi va uslublari jamlanmasidan foydalanadigan jarayondir. *Telekommunikatsiya* kompyuter tarmoqlari va zamonaviy texnik aloqa vositalari negizida ma'lumotlarni masofaga uzatishdir. «Axborot» so'zi lotincha —information so'zidan olingan bo'lib, biror ish holati yoki kishi faoliyati haqida ma'lum qilish, xabar berish, biror narsa haqidagi ma'lumot, degan ma'noni anglatadi. Informatika nazariyasida saqlash, qayta tuzish va uzatish ob'ekti sanalgan barcha ma'lumotlar axborot deb yuritiladi. Axborot – bu, yaratuvchisi doirasida qolib ketmagan va xabarga aylangan, bilimlar noaniqligi, to'liqsizligi darajasini kamaytiradigan hamda og'zaki, yozma yoki boshqa usullar (shartli

signallar, texnik vositalar, hisoblash vositalari va x.k.) orqali ifodalash mumkin bo'lgan atrof-muhit (ob'ektlar, voqea-hodisalar) to'g'risidagi ma'lumotlardir.

Mazkur yo'nalishda quyidagilar muhim sanaladi:

- axborot – bu har qanday ma'lumot emas, balki u mavjud noaniqliklarni kamaytiruvchi yangi bir ma'lumotdir;

- axborot uni yaratuvchisidan tashqarida mavjud bo'ladi, u o'z yaratuvchisidan uzoqlashgan, inson tafakkurida aks etgan bilimdir;

- axborot xabarga aylanadi, chunki u belgilar ko'rinishida ma'lum bir tilda ifodalanadi;

- xabar moddiy tashuvchiga yozib qo'yilishi mumkin (xabar axborotni uzatish shaklidir);

- xabar uning muallifi ishtirokisiz aks ettirilishi mumkin;

- u jamoat kommunikatsiyasi kanallari orqali uzatiladi.

Axborotning asosiy xossalari:

1. to'liqlik;
2. yaroqlilik;
3. ishonchlilik;
4. dolzarblik;
5. tushunarlik.

Axborotning ifodalanish shakllari va uning turlari. Axborotning muhim xarakteristikalaridan biri uning adekvatligi hisoblanadi. Axborotning adekvatligi – olingan axborot yordamida yaratilgan obrazning real ob'ekt, jarayon, hodisa va shunga o'xshashlarga mosligining ma'lum darajasi. Axborotning adekvatligi uchta shaklda ifodalanishi mumkin: semantik, sintaktik va pragmatik.

Semantik (ma'noli) adekvatlik – ob'ektning uning obraziga (qiyofasiga) muvofiqlik darajasini aniqlaydi. Semantik nuqtai nazar axborotning ma'noli mazmunini hisoblashni ko'zlaydi. Bunda axborot aks ettirgan ma'lumotlar tahlil qilinadi, ma'nolar bog'liqligi ko'riladi.. Masalan, axborotni kodlar orqali ifodalashni ko'rsatish mumkin.

Sintaktik adekvatlik – axborotning mazmuniga tegmagan holda, uning rasmiy-strukturaviy xarakteristikalarini ifodalaydi. Sintaktik darajadagi axborotni ifodalash usulida axborot elituvchi turi, uzatish va qayta ishlash tezligi, ifodalash kodining o'lchamlari, bu kodlarni o'zgartirish aniqliligi va ishonchliligi hisobga olinadi. Axborotning mazmuniga ahamiyat berilmaganligi sababi, bunday axborot ma'lumot deb ataladi.

Pragmatik (foydalanuvchanlik) adekvatlik – axborot bilan foydalanuvchining munosabatlarini aks ettiradi, axborotni uning asosida amalga oshiriladigan boshqarish sistemasi maqsadiga muvofiqligini ifodalaydi. Axborotning pragmatik xususiyatlari faqat axborot (ob'ekt), foylanuvchi va boshqarish maqsadlarining umumiylikida namoyon bo'ladi. Adekvatlikning ushbu shakli axborotdan amaliy foylanish bilan bevosita bog'langan. Informatikada asosiy masala bo'lib hisoblash texnikasi qurilmalaridan axborotni saqlash, qayta ishlash va uzatishda qanday foydalanish hisoblanadi.

Shuning uchun informatikada axborotning ikki xil turi bilan ish ko'riladi, ya'ni analog va raqamli. Ko'pgina hisoblash texnikasi qurilmalari raqamli axborotni qayta ishlaydi. Analog axborotni raqamli axborotga o'zgartiruvchi maxsus qurilmalar mavjud bo'lib, bunday o'zgartirishni *analog-raqamli o'zgartirish* deyiladi.

Kompyuter faqat raqamli ko'rinishdagi axborotni qayta ishlaydi. Barcha boshqa turdagi axborot (ovozlar, tasvirlar va boshq.) komp yuterda qayta ishlanishi uchun raqamli ko'rinishga keltirilishi zarur. Ovozni raqamli ko'rinishga o'zgartirish uchun kichik vaqt oralig'ida ovoz intensivligini o'lchash va har bir o'lchash natijalarini raqam ko'rinishida aks ettirish zarur. Kompyuter dasturi yordamida olingan axborotni qayta ishlab, hosil bo'lgan natijani ovoz shakliga qaytarish mumkin. Komp yuterda matni axborotni qayta ishlash uchun matni kompyuterga kiritilayotganda har bir xarf ma'lum bir raqam bilan kodlanadi. Tashqi qurilmalarga (monitor ekrani yoki printer) chiqarilayotganda esa inson qabul qilish uchun ushbu raqamlar orqali xarflarning tasviri quriladi. Xarflar to'plami va raqamlar o'rtasidagi moslik belgilarni kodlashtirish deb ataladi. Kompyuterda barcha raqamlar 0 va 1 orqali ifodalanadi. Kompyuter ikkilik sanoq sistemasida ishlaydi. Kompyuterda axborot birligining o'lchovi bo'lib bit, ya'ni 0 yoki 1 qiymatni qabul qilishi mumkin bo'lgan ikkilik razryad hisoblanadi. Komp yuter komandalari alohida bitlar bilan emas, balki sakkiz bit bilan birgalikda ishlaydi. Sakkizta ketma-ket bit bir baytni tashkil etadi. Baytlar yordamida raqamli ko'rinishda ifodalangan har qanday axborotni kodlashtirish mumkin. Bir baytda 256 xil belgilardan birining qiymatini kodlashtirish mumkin ($256=2^8$) bo'ladi. Baytning qiymati uchun undagi bitlarning joylashgan o'rni muhimdir. Axborotda qatnashgan har qanday belgi 1 bayt hajmli deb hisoblanadi. Masalan, "Sh" harfi – 1 bayt, "Kitob" – 5 bayt hajmga ega. Bir bayt 0 dan 255 qiymatni qabul qilishi mumkin. Axborotning yirik birliklari:

Bayt – ketma-ket yozilgan 8 ta bitdan iborat bo'lib, uning xar birida bittadan belgini ifodalash mumkin. Bunday belgilar soni $2^8=256$ tadan iborat. Bundan tashqari axborotlarning hosilaviy o'lchov birliklari xam mavjud bo'lib, ular *kilobayt(Kb)*, *megobayt(Mb)*, *Gigobayt(Gb)*, *eksobayt(Eb)* lardan iboratdir:

$$1 \text{ Kb} = 2^{10} = 1024 \text{ bayt};$$

$$1 \text{ Mb} = 2^{10} = 1024 \text{ Kb};$$

$$1 \text{ Gb} = 2^{10} = 1024 \text{ Mb};$$

$$1 \text{ Eb} = 2^{10} = 1024 \text{ Gb};$$

Mavzu bo'yicha savollar to'plami:

1. "Kimyo injinerigida axborot texnologiyalari" fanining ta'rifini bering.
2. O'zbekiston Respublikasining axborotlashtirish konsepsiyasida qanday masalalar qo'yilgan?
3. "Axborot" deb nima tushuniladi?
4. "Ma'lumotlar" deb nimani tushunamiz?
5. Axborotni qanday belgilar bo'yicha tavsiflash mumkin?
6. Axborotning qanday turlari bor?

Mavzu bo'yicha glossariy:

Axborot - manbalari va taqdim etilish shaklidan qat'i nazar shaxslar, predmetlar, faktlar, voqealar, hodisalar va jarayonlar to'g'risidagi ma'lumotlar.

Axborot mulkdori - o'z mablag'iga yoki boshqa qonuniy yo'l bilan olingan axborotga egalik qiluvchi, undan foydalanuvchi va uni tasarruf etuvchi yuridik yoki jismoniy shaxs.

Axborotni muhofaza etish - axborot borasidagi xavfsizlikka tahdidlarning oldini olish va ularning oqibatlarini bartaraf etish chora-tadbirlari.

Axborot resurslari - alohida hujjatlar, hujjatlarning alohida to'plamlari, axborot tizimlaridagi (kutubxonalaridagi, arxivlardagi, fondlardagi, ma'lumotlar banklaridagi va boshqa axborot tizimlaridagi) hujjatlar va hujjatlarning to'plamlari.

Axborot sohasi - sub'ektlarning axborotni yaratish, qayta ishlash va undan foydalanish bilan bog'liq faoliyati sohasi.

Axborot borasidagi xavfsizlik - axborot sohasida shaxs, jamiyat va davlat manfaatlarining himoyalanganlik holati.

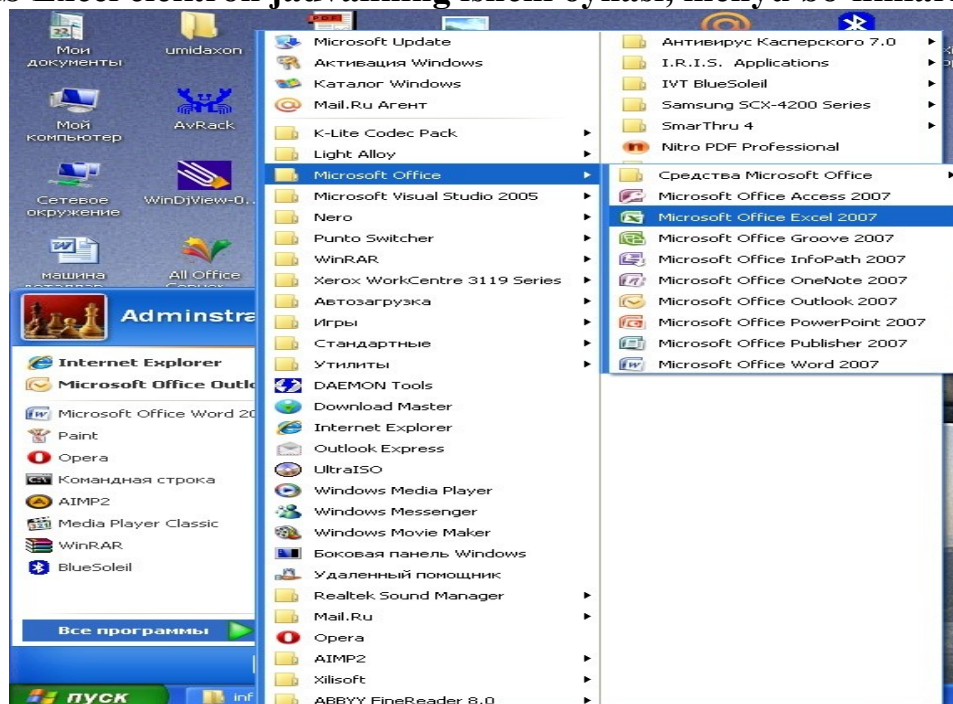
Axborotning ishonchliligi – bu axborotning shunday xususiyatiki, bunda u tizimning faqat ruxsat etilgan va tekshiruvdan o'tgan sub'ektlarigagina (foydalanuvchilar, dasturlar, jarayonlar va h.) ma'lum bo'ladi.

2-ma'ruza. MS Excel elektron jadvali haqida umumiy tushuncha.

Reja:

1. MS Excel elektron jadvalining ishchi oynasi, menyu bo'limlari.
2. Ustun kengligini o'zgartirish. Qatorlar balandligini o'zgartirish.
3. Sonlarni kiritish.

2.1. MS Excel elektron jadvalining ishchi oynasi, menyu bo'limlari.



2.1-rasm. MS Excel dasturiga kirish.

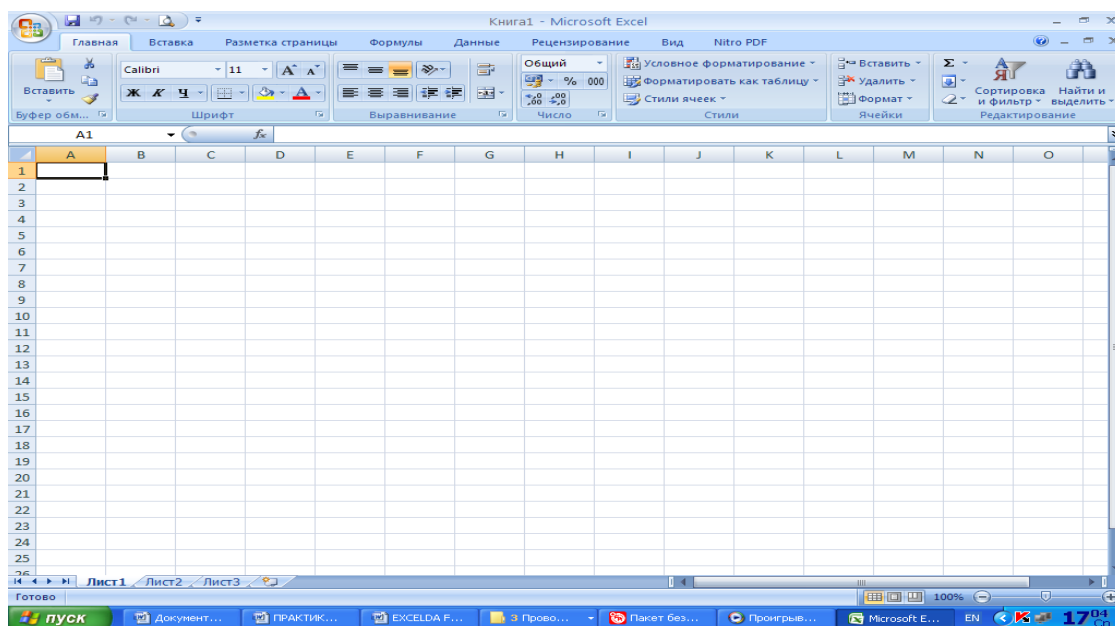
Microsoft Excel Microsoft Office dasturining tarkibidan bir bo'lagi hisoblanadi. U o'zida elektron jadval, ilova, funksiyalar va grafik instrumentlarni aks ettiradi. Excel ning eng muhim ustunligi sanoat va ilmiy sohada keng qamrovda ishlatishidir. Shunday qilib bu mukammal interfeys nafaqat hisob kitobni bajarish uchun balki turli rejali dasturlar ta'minoti uchun ham ishlatiladi. Shunday qilib foydalanuvchi Excel da boshqa dasturlar bilan o'zaro aloqa qilib ham ishlashi mumkin.

Elektron jadvallar asosan iqtisodiy masalalarni yechishga mo'ljallangan bo'lsa-da, uning tarkibiga kiruvchi vositalar boshqa sohaga tegishli masalalarni yechishga ham, masalan, formulalar bo'yicha hisoblash ishlarini olib borish, grafik va diagrammalar qurishga ham katta yordam beradi. Shuning uchun EXCEL dasturini o'rganish muxim ahamiyat kasb etadi va xar bir foydalanuvchidan EXCEL bilan ishlay olish ko'nikmasiga ega bo'lish talab etiladi.

Inson o'z ish faoliyati davomida ko'pincha biror kerakli ma'lumot olish uchun bir xil, zerikarli, ba'zida esa, murakkab bo'lgan ishlarini bajarishga majbur bo'ladi. MICROSOFT EXCEL dasturi mana shu ishlarni osonlashtirish va qiziqarli qilish maqsadida ishlab chiqilgandir.

MICROSOFT EXCEL elektron jadvali hisoblash vositasi sifatida qaralib, iqtisodiy va moliyaviy masalalarni yechishda yordam beribgina qolmay, balki har kungi xarid qilinadigan oziq-ovqatlar, uy-ro'zg'or buyumlari hamda bankdagi hisob raqamlari hisob-kitobini olib borishda ham yordam beruvchi tayyor dasturdir.

Foydalanuvchi Excel dasturini ishga tushrganda ustun va qatorlarni ko'radi. Ustunlar lotin harflar bilan, qatorlar esa sonlar orqali ifodalangan bo'ladi. Ularni birlashgani yacheykalar deb yuritiladi.



2.2-rasm MS Excel ishchi oynasi.

MICROSOFT EXCEL dasturida matnlar — bu raqamlar, bo'sh joylar va boshqa belgilardan iborat bo'lgan ixtiyoriy ketma-ketlikdir. Kiritilayotgan

ma'lumotlarimiz yacheykaga sig'magan holda sig'may qolgan qismi ekran yacheykasining o'ng qismiga o'tkaziladi (agar u bo'sh bo'lsa). Bunday hollarda quyidagi usullardan foydalanib, yacheyka kengligi va balandligini o'zgartirish mumkin. Bir necha so'zdan iborat va yacheykaga sig'magan matnni ko'rish.

2.2. Ustun kengligini o'zgartirish.Qatorlar balandligini o'zgartirish

Bir yoki bir nechta ustun kengligini o'zgartirish usullarini qarab chiqamiz. Buning uchun o'zgartirish kerak bo'lgan ustunlarni belgilab olish kerak. Ular bir xil uzunlikdagi «Format Yacheyki» (Yacheyka bichimi) buyruqlarini tanlash va bayroqcha (flajok) belgisini «переносить по словам» degan ibora ustiga joylashtirish kerak. Yacheykada yangi satrni boshlash uchun Alt + Enter tugmalarini bosish kerak. yacheykalar bo'lishi lozim.

Sichqoncha ko'rsatkichini ustunning o'ng chegara chizig'iga olib kelib, ustun kengligini keragicha uzaytirish.

Bu ish sichqoncha tugmasi bosilgan polatda amalga oshiriladi.

«Формат»—«Столбец» (Ustun) — «Ширина» (Kenglik) buyruqlarini ketma-ket tartib bilan tanlab, «Ширина столбцов» (Ustun kengligi) muloqot oynasida parametriga kerakli qiymat beriladi va OK tugmasi bosiladi. «Формат»—«Столбец »(Ustun) —«Автоподбор ширина» (Kenglikni avtomatik tanlash) buyrug'ini bajarish

Bunda ajratib ko'rsatilgan ustun kengligi shunday tanlanadiki, unga eng uzun ma'lumotlar qatorini kiritish mumkin bo'ladi.

Ustunning o'ng chegarasiga sichqoncha ko'rsatkichini olib kelib, sichqonchaning chap tugmasi ikki marta tez-tez bosilsa, kiritilayotgan qatorning eng uzun ma'lumoti sig'adigan ustun kengligi avtomatik ravishda tanlanadi.

Qatorlar balandligini o'zgartirish

Qatorlarga ma'lumotlarni kiritish uning shriftiga, ya'ni kiritilayotgan belgilarning o'lchamiga bog'liq bo'lib, ko'pincha NORMAL bichim (stil) ishlatiladi. MICROSOFT EXCEL dasturida qatorlarning balandligi kiritilayotgan ma'lumotlarning qanday o'lchamdagi shriftda yozilganiga qarab avtomatik ravishda o'zgarib boradi. Bundan tashqari qatorlar balandligini yuqoridagi ustunlar kengligini o'zgartirish buyruqlari orqali xam o'zgartirish mumkin. Buning uchun buyruqlardagi «Столбец» (Ustun) buyrug'i o'rniga «Строка» (Satr) buyrug'ini olinadi.

2.3.Sonlarni kiritish

Ular bir yacheykaga kiritilayotgan sonlar o'zgarmaslar (konstantalar) sifatida qabul qilinadi. MICROSOFT EXCEL dasturida sonlar quyidagi 1,2,3,...9,0,Q,—,(),G', \$,%,E,e belgilari orqali kiritiladi. Tugmalarning boshqa barcha kombinatsiyalari orqali kiritilayotgan ma'lumotlar raqamli va raqamsiz belgilardan iborat bo'lib, matn sifatida qaraladi. Manfiy sonlar kiritilayotganda son oldiga «—» ishorasi qo'yiladi yoki son qavs ichiga olinib yoziladi. Masalan, -5 yoki (5). Kiritilayotgan sonlar xamma vaqt yacheykaning o'ng chegarasidan boshlab yoziladi.

Kiritilayotgan sonli qiymatlar bichimlanmagan holatda bo' ladi, boshqacha aytganda, ular oddiy raqamlar ketma- ketligidan iborat bo'ladi. Shuning uchun sonlarni bichimlash zarur. Sonlarni bunday bichimlashdan maqsad — yacheykadagi ma'lumotlarni, ya'ni sonlarni oson o'qish imkonini berishdir.

MICROSOFT EXCEL dasturida sonli qiymatlar 12 xil bichimlanadi .

1. Umumiy.
2. Sonli.
3. Pul bilan bog'lik.
4. Moliyaviy.
5. Kun,oy (sana).
6. Vaqt.
7. Foiz bilan bog'lik.
8. Kasrli.
9. Eksponentsial.
10. Matnli.
11. Qo'shimcha (barcha bichimlar).

Agar yacheykadagi belgilar o'rnida «reshetka» (#####) paydo bo'lsa, tanlangan bichimdagi sonlar ustun kengligiga sig'magan hisoblanadi. Bunday hollarda ustun kengligini o'zgartirish yoki boshqa sonli bichimlashga o'tish kerak bo'ladi.

Agar sonli qiymatlar bichimlangandan keyin yacheykadagi jadval kursori sonli qiymatlari bilan boshqasiga o'tkazilsa, formulalar qatorida sonli qiymatlarning bichimlanmagan ko'rinishi hosil bo'ladi, chunki bichimlash sonli qiymatning yacheykadagi ko'rinishigagina ta'sir etadi.

Mavzu bo'yicha savollar to'plami:

- 1.Elektron jadvallari qaysi maqsadlarda ishlatiladi?
- 2.Elektron jadvallar asosan qaysi sohalarda qo'llaniladi?
- 3.Microsoft Excel elektron jadvalida nechta ustun bo'lishi mumkin?
- 4.Piktogrammalar guruhini sanab bering?
- 5.Jadvalga rasm kiritish uchun qanday ishlar bajariladi?

Mavzu bo'yicha glossariy:

Matnlar — bu raqamlar, bo'sh joylar va boshqa belgilardan iborat bo'lgan ixtiyoriy ketma-ketlik.

Yacheyka – ustun va satr kesishmasi

Sonlarni bichimlash — yacheykadagi ma'lumotlarni, ya'ni sonlarni oson o'qish imkonini berish.

Axborotni muhofaza etish - axborot borasidagi xavfsizlikka tahdidlarning oldini olish va ularning oqibatlarini bartaraf etish chora-tadbirlari.

Axborot resurslari - alohida hujjatlar, hujjatlarning alohida to'plamlari, axborot tizimlaridagi (kutubxonalardagi, arxivlardagi, fondlardagi, ma'lumotlar banklaridagi va boshqa axborot tizimlaridagi) hujjatlar va hujjatlarning to'plamlari.

Axborot sohasi - sub'ektlarning axborotni yaratish, qayta ishlash va undan foydalanish bilan bog'liq faoliyati sohasi.

3-ma'ruza. MS Excel asoslari.

Reja:

1.Sonlarni bichimlash.

2.Bosh menyu yordamida bichimlash.

3.Jadvalga formula kiritish.

1.Sonlarni bichimlash.

Bichimlashdan oldin kerakli yacheykalarni ajratib olish zarur.

Uskunalar paneli yordamida bichimlash. Uskunalar panelida bir qancha bichimlash tugmalari joylashgan bo'lib, ular kerakli bichimlash turini tez tanlab olish va foydalanish imkonini beradi.

Yacheykalar ajratilgandan keyin tanlangan tugmalarga sichqoncha ko'rsatkichini olib borib, tugmasini bosish kerak. Shundan so'ng yacheykada tanlangan bichim asosida ish olib boriladi.

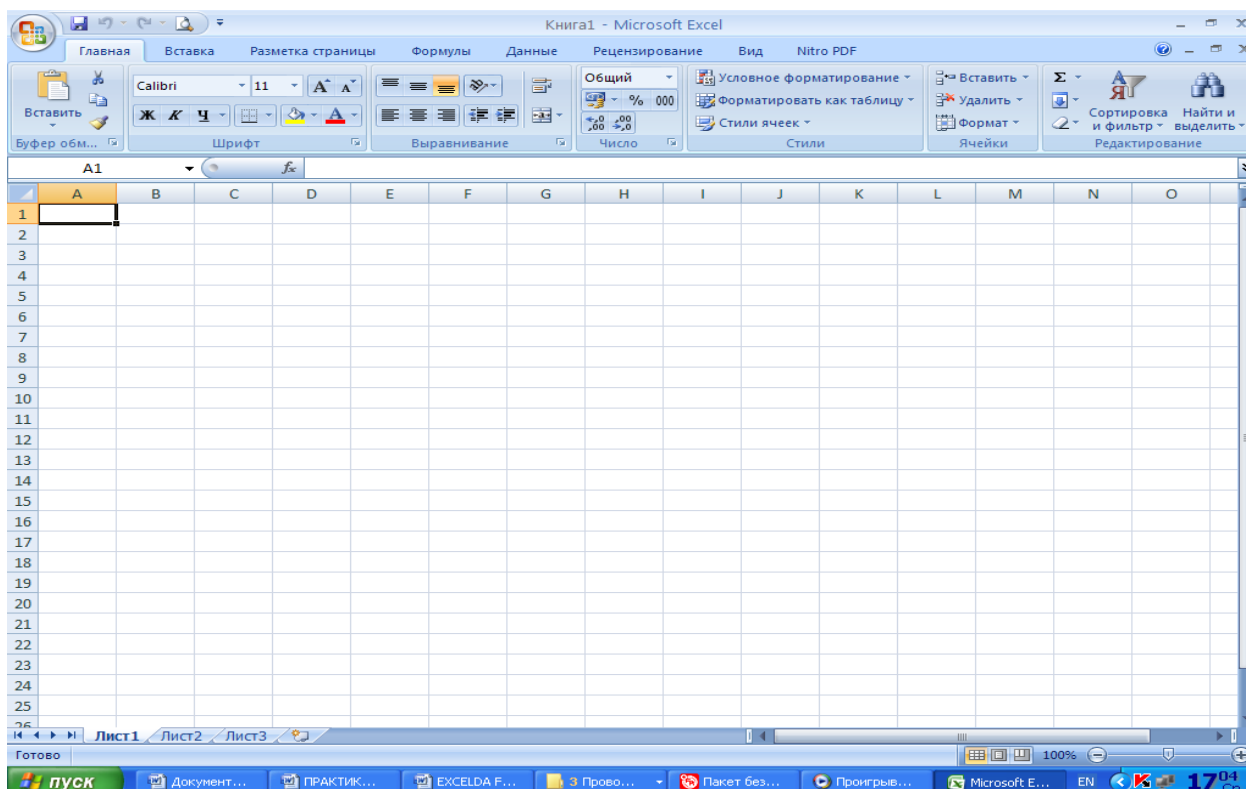
Bosh menyu yordamida bichimlash. Bosh menyuda bichimlash buyrug'i quyidagi tartibda amalga oshiriladi.

«Формат»—«Формат ячейки» (Yacheyka bichimi) buyruqlari tanlanib, muloqot oynasidan «число» (Son)ning kerakli parametrli bichimi tanlanadi.

Tekislash. Matnlar yacheykaga kiritilganda, ko'pincha, yacheykaning chap chegarasida tekislanadi. Sonli qiymatlar esa yacheykaning o'ng chegarasidan boshlab tekis xolatda yoziladi. Yacheykadagi yozuv chegaralarini turli tartibda o'zgartirish mumkin. Buning uchun kerakli yacheykalar ajratib olinadi va uskunalar panelida joylashgan turli tomonli bichimlash piktogrammalaridan biri tanlab olinadi. Bundan tashqari gorizontal menyudagi «Формат»—«Ячейка»—«Выравнивание» (Tekislash) buyruqlari orqali xam o'zgartirish, ya'ni tekislash mumkin.

Excel Microsoft Office paketi tarkibidagi dastur bo'lib, u Windows amaliyot tizimi boshqaruvida ishlovchi hamda ma'lumotli elektron jadvallarni tayyorlash va qayta ishlashga mo'ljallangan.

Excelda tayyorlangan har bir hujjat (ma'lumotli jadval) ixtiyoriy nom va . XLS kengaytmadan iborat fayl bo'ladi. Excelda odatda bunday fayl "Ish kitobi" (Workbook) deb yuritiladi. Microsoft Excelning asosiy ish sohasi – bu "Ish kitobi" bo'lib, u bir yoki bir necha ish varaqalardan iborat. Ish varag'ida buxgalter kitobi kabi, sonlar, matnlar, arifmetik ifodalar, hisoblar qator va ustunlardan joylashgan bo'ladi. Excelning buxgalter kitobidan asosiy farqi barcha hisob ishlarini uning o'zi bajaradi. Lekin ma'lumotlarni kiritish foydalanuvchi zimmasida qoladi.



3.1-rasm. MS Excelda ustunlar va qatorlar.

3.2.Bosh menyu yordamida bichimlash.

Excel elektron jadvali 1048576 (Excel – 2007da) qator (row) va 16384 ta ustun (column) dan iborat. Qatorlar 1dan 1048576 gacha bo'lgan butun sonlar bilan tartiblangan, ustunlar esa lotin alifbosining bosh harflari (A, B , ... , Z, AA, AB,, IV,.....XFD) bilan belgilangan.

Asosiy tarkibiy elementi – yacheyka (cell) joylashgan. Har bir yacheykaga son, matn yoki formula tarzidagi ma'lumotlar kiritiladi. Ustun kengligini va qator balandligini o'zgartirish ham mumkin. Jadvalning tanlangan yacheykasiga o'tish uchun aniq manzil (adres) ko'rsatilishi kerak. U qator va ustun kesishmasida , masalan A1, B4, F9, AB3 kabi ko'rsatiladi.

Zamonaviy kompyuterlarning dasturiy ta'minotining tarkibiy qismiga kiruvchi MICROSOFT OFFICE paketidagi asosiy vositalardan biri jadval protsessori deb ataluvchi EXCEL dasturidir. EXCEL WINDOWS operatsion qobig'i boshqaruvida elektron jadvallarni tayyorlash va ularga ishlov berishga mo'ljallangan.

Elektron jadvallar asosan iqtisodiy masalalarni yechishga mo'ljallangan bo'lsa-da, uning tarkibiga kiruvchi vositalar boshqa soxaga tegishli masalalarni yechishga xam, masalan, formulalar bo'yicha xisoblash ishlarini olib borish, grafik va diagrammalar qurishga xam katta yordam beradi. Shuning uchun EXCEL dasturini o'rganish muxim ahamiyat kasb etadi va xar bir foydalanuvchidan EXCEL bilan ishlay olish ko'nikmasiga ega bo'lish talab etiladi.

Inson o'z ish faoliyati davomida ko'pincha biror kerakli ma'lumot olish uchun bir xil, zerikarli, ba'zida esa, murakkab bo'lgan ishlarini bajarishga majbur

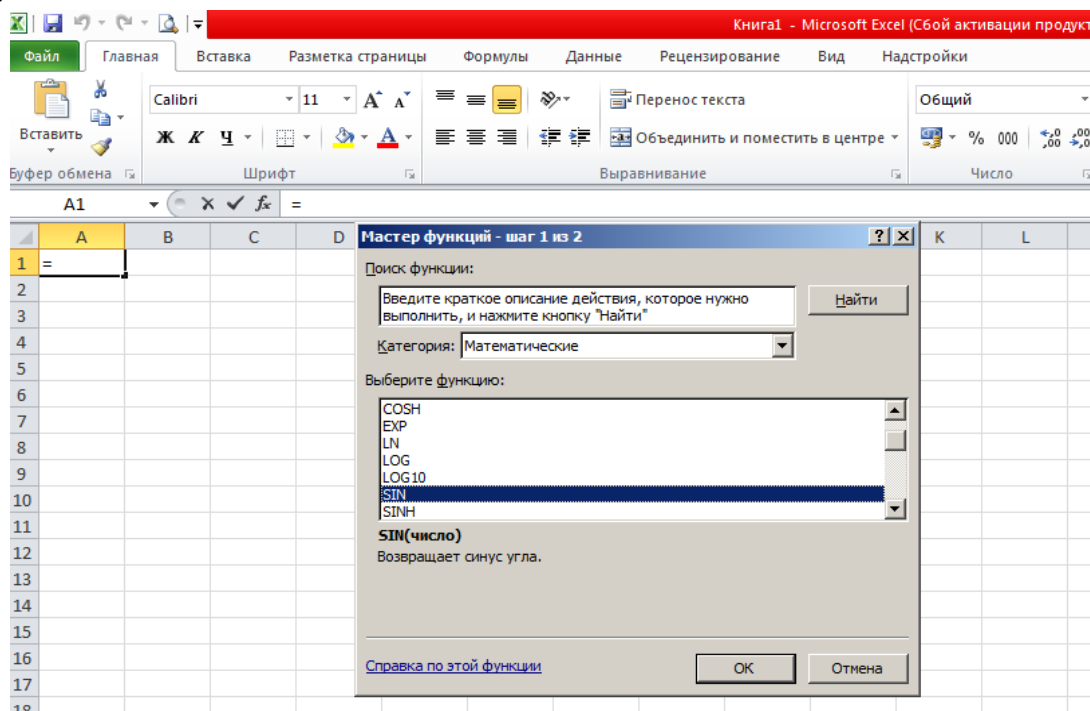
bo'ladi. MICROSOFT EXCEL dasturi mana shu ishlarni osonlashtirish va qiziqarli qilish maqsadida ishlab chiqilgandir.

MICROSOFT EXCEL elektron jadvali xisoblash vositasi sifatida qaralib, iqtisodiy va moliyaviy masalalarni yechishda yordam beribgina qolmay, balki xar kungi xarid qilinadigan oziq-ovqatlar, uy-ruzg'or buyumlari xamda bankdagi xisob raqamlari xisob-kitobini olib borishda xam yordam beruvchi tayyor dasturdir.

Formula — bu mavjud qiymatlar asosida yangi qiymatlarni hisoblovchi tenglamadir. Formular yordamida elektron jadvalda ko'pgina foydali ishlarni amalga oshirish mumkin. Elektron jadvallar formulalarsiz oddiy matn muxarririga aylanib qoladi. Formulalarsiz elektron jadvallarni tasavvur qilish qiyin.

Jadvalga formulani qo'yish uchun uni kerakli yacheykaga kiritish kerak. Formulalarni xam boshqa ma'lumotlar singari o'zgartirish, saralash, ulardan nusxa ko'chirish va o'chirish mumkin. Formuladagi arifmetik amallar sonli qiymatlarni xisoblashda, maxsus funksiyalar matnlarni qayta ishlashda xamda yacheykadagi boshqa qiymatlarni xisoblashda ishlatiladi.

Bundan tashqari bir qator vazifalarni bajaruvchi funksiyalar mavjud. Masalan trigonometric (sin, cos, tan va boshqalar) statistika (среднее значение, среднее va boshqalar) algebraik (sum, krugliy, log va boshqalar) logik (if, false). Kerakli funksiyadan foydalanish uchun foydalanuvchi $f(x)$ tugmasini bosish kerak.



3.2-rasm. Funksiyalar ustasi

3.3.Jadvalga formula kiritish.

Sonlar va matnlar. Formuladagi xisoblashlarda qatnashayotgan sonlar va matnlar boshqa yacheykalarda joylashgan bo'lishi mumkin bo'lsa-da, ularning ma'lumotlarini oson almashtirish mumkin. Masalan, EXCEL boshlang'ich

ma'lumotlar o'zgartirilsa, formulalarni qayta xisoblab chiqadi.

Formula quyidagi elementlardan ixtiyoriysini o'z ichiga olishi mumkin:

— Operatorlar. Bittadan oshiq operatorlardan tuzilgan formulani tuzishda EXCEL bu operatorlarni taxlil qiladi. Bunda standart matematik qoidalarga asoslanadi. (Arifmetik amallarni bajarish tartibi saqlanib qoladi.)

Excelda formulalarni hisoblash va bajarish quyidagi tartib asosida amalga oshiriladi:

Birinchi bo'lib qavs ichidagi ifodalar qarab chiqiladi.

Undan keyin amallar bajarish tartibi saqlangan xolda operatorlar bajariladi.

Agar formulalarda bir xil tartibli bir necha operatorlar bo'lsa, ular ketma-ket chapdan o'ngga qarab bajariladi.

Quyidagi jadvalda formulalarda qo'llaniladigan operatorlarning bajarilish tartibi ko'rsatilgan:

3.1-jadval

Belgilar	Operatorlar	Bajarilish tartibi
^	Darajaga ko'tarish	1
*	Ko'paytirish	2
/	Bo'lish	2
+	Ko'shish	3
-	Ayirish	3
&	Konkatenasiya	4
=	Tenglik	5
>	Dan katta	5
<	Dan kichik	5

— Diapazon va yacheykalarga yuborish — kerakli ma'lumotlarni saqlovchi diapazon va yacheykalar nomi yoki manzili ko'rsatiladi. Masalan: D10 yoki A1:YE8.

— Sonlar.

— Ishchi jadval funksiyalari. Masalan, SUM.

Agar formula yacheykaga kiritilsa, unda yacheykada kiritilgan formula asosidagi hisob-kitob natijasi ko'rinadi. Lekin formulaning o'zi tegishli yacheyka faollashtirilsa formulalar qatorida paydo bo'ladi.

Formulalar xar doim «=» belgisi bilan boshlanadi. Ushbu belgi yordamida EXCEL matn va formulalarni farqlaydi.

Yacheykaga formulalarni kiritishning ikkita usuli mavjud:

1. Formulani klaviatura orqali kiritish: «=» belgisini quyib, keyin formulalar kiritiladi. Kiritish paytida belgilar formulalar qatorida xamda faollashgan yacheykada paydo bo'ladi. Formulalarni kiritishda odatdagi taxrirlash tugmalaridan foydalanish mumkin.

2. Yacheykalar manzilini ko'rsatish yo'li bilan formulalar kiritish: Bu usulda

xam formulalar klaviaturadan kiritish orqali, lekin kamroq foydalangan holda amalga oshiriladi. Ushbu usulda yacheykalar manzilini kiritish o'rniga ular ko'rsatiladi, xolos. Masalan, $A3 = A1 + A2$ formulasini kiritish uchun quyidagilarni bajarish kerak.

3· jadval kursori A3 yacheykaga utkaziladi;

4· « $\Rightarrow$ » belgisi kiritiladi. Formulalar qatori yonida «kiritish» (ВВОД) yozuvi paydo bo'ladi;

5· sichqoncha ko'rsatkichi A1 yacheykaga olib boriladi va chap tugmachasi bosiladi. Natijada yacheyka ajratib ko'rsatiladi, ya'ni uning atrofida xarakatlanuvchi ramka (rom) paydo bo'ladi. A3 yacheykasi formulalar qatorida — A1 yacheyka manzili ko'rinadi. U qatorida esa «Укажите» (Ko'rsating) yozuvi paydo bo'ladi:

6· « $\Rightarrow$ » belgisi kiritiladi. Natijada xarakatlanuvchi rom yo'qolib, yana «ВВОД» (Kiritish) so'zi chiqadi;

7· sichqoncha ko'rsatkichi A2 yacheykaga o'tkaziladi va tugmachasi bosiladi. Formulaga A2 yacheyka qo'shiladi;

8· ENTER tugmasini bosish bilan formulani kiritish yakunlanadi.

Yacheyka manzilini ko'rsatish usuli klaviatura yordamida kiritish usulidan oson va tez bajariladi.

Formulalarni boshqa ishchi jadvallar yacheykalariga xam yuborish mumkin, boshqacha aytganda, formulalar bir necha joyda takrorlanishi mumkin. Hattoki, boshqa ishchi kitobdagi ishchi jadvallarda ham. Buning uchun EXCELda maxsus yozuv ishlatiladi.

Yacheykalardagi ma'lumotlarni boshqa ishchi jadvallarga yuborish (o'tkazish)

Joriy ishchi kitobdagi ma'lumotlarni boshqa ishchi kitobdagi yacheykaga yuborish quyidagi usullardan foydalanib hal qilinadi:

Joy nomi. Yacheyka manzili.

Boshqacha qilib aytganda, yacheyka manzili oldiga joyning nomi undov belgisi bilan qo'yiladi. Masalan, $=A1*list1!A2$

Bu formulada joriy ishchi jadvaldagi A1 yacheyka qiymati A2 yacheyka qiymatiga ko'paytiriladi va «List2» ishchi varaqida joylashadi. Agar jo'natishda ishchi jadvalning nomi bir yoki bir nechta bo'shlikni o'z ichiga olsa, jadvalning nomi bittali qushtirnoq ichiga olinib ko'rsatiladi.

Mavzu bo'yicha savollar to'plami:

1. Elektron jadvallari qaysi maqsadlarda ishlatiladi?
2. Elektron jadvallar asosan qaysi sohalarda qo'llaniladi?
3. Microsoft Excel elektron jadvalida nechta ustun bo'lishi mumkin?
4. Piktogrammalar guruhini sanab bering?
5. Jadvalga rasm kiritish uchun qanday ishlar bajariladi?

Mavzu bo'yicha glossariy:

Matnlar — bu raqamlar, bo'sh joylar va boshqa belgilardan iborat bo'lgan ixtiyoriy ketma-ketlik.

Yacheyka — ustun va satr kesishmasi

Sonlarni bichimlash — yacheykadagi ma'lumotlarni, ya'ni sonlarni oson o'qish imkonini berish.

Formula — bu mavjud qiymatlar asosida yangi qiymatlarni hisoblovchi tenglama.

Axborot resurslari - alohida hujjatlar, hujjatlarning alohida to'plamlari, axborot tizimlaridagi (kutubxonalaridagi, arxivlardagi, fondlardagi, ma'lumotlar banklaridagi va boshqa axborot tizimlaridagi) hujjatlar va hujjatlarning to'plamlari.

Axborot sohasi - sub'ektlarning axborotni yaratish, qayta ishlash va undan foydalanish bilan bog'liq faoliyati sohasi.

5-ma'ruza. Algoritmash asoslari. Algoritm turlari va xossalari.

Reja:

1. "Algoritm" tushunchasi va unga misollar.

2. Algoritmning asosiy xossalari.

3. Algoritmni ishlab chiqish.

1. "Algoritm" tushunchasi va unga misollar.

Algoritm deb, masalani yechish uchun bajarilishi lozim bo'lgan amallar ketma-ketligini aniq tavsiflaydigan qoidalar tizimiga aytiladi.

Boshqacha aytganda, algoritm – boshlang'ich va oraliq ma'lumotlarni masalani yechish natijasiga aylantiradigan jarayonni bir qiymatli qilib, aniqlab beradigan qoidalarining biror bir chekli ketma-ketligidir.

Buning mohiyati shundan iboratki, agar algoritm ishlab chiqilgan bo'lsa, uni echilayotgan masala bilan tanish bo'lmagan biron bir ijrochiga, shu jumladan kompyuterga xam bajarish uchun topshirsa bo'ladi va u algoritmning qoidalariga aniq rioya qilib masalani yechadi.

Masalan, ko'rib o'tilgan birinchi misolni yechish algoritmini quyidagicha bayon qilsa bo'ladi:

- 1) kompyuter xotirasiga V_0 va g o'zgaruvchilarning sonli qiymatlari kiritilsin;
- 2) t ning qiymati $t = V_0 / g$ formula bilan xisoblansin;
- 3) h ning qiymati $h = V_0 t - g t^2 / 2$ (1) formula bilan xisoblansin;
- 4) t va h o'zgaruvchilarning sonli qiymatlari ekranga yoki qog'ozga chiqarilsin;
- 5) hisoblash to'xtatilsin.

Masalaning qo'yilishida koptok 29, 5 m /sek bilan tepilsa, degan shart bor edi. ya'ni, $V_0 = 29, 5$ va $g = 9, 81$ bo'lsa, t va h qancha bo'ladi? (Talabalarning o'ziga yechish taklif etiladi: $t = 3$ sek, $h = 43, 35$ m.) Natija xammada bir xil chiqadi.

Ikkinchi misolning yechish algoritmi quyidagicha bo'ladi:

- 1) kompyuter xotirasiga a va b ning qiymati kiritilsin;
- 2) to'g'ri to'rtburchaklar soni n kiritilsin;
- 3) to'rtburchaklar asosi (eni) xisoblansin: $h = (b-a)/n$
- 4) 1-to'rtburchak balandligi (bo'yi) aniqlansin: $x_1 = a$
- 5) 1-to'rtburchak yuzi xisoblansin: $S_1 = \text{sqr}(x_1) * h$
- 6) S_1 ning qiymati eslab qolinsin;

- 7) 2-to'rtburchakka o'tilsin; $x_2 = x_1 + h$ (balandligi shunga bog'liq)
 - 8) 2-to'rtburchak yuzi xisoblansin: $S_2 = \text{sqr}(x_2) \cdot h$
 - 9) S_2 ning qiymati S_1 ning qiymatiga qo'shib qo'yilsin va yig'indi eslab qolinsin;
 - 10) n -to'rtburchakka o'tilsin: $x_n = x_{n-1} + h = b$
 - 11) n -to'rtburchak yuzi xisoblansin: $S_n = \text{sqr}(b) \cdot h$
 - 12) S_n ning qiymati $S_1, S_2, \dots, S_{n-1}$ lar qiymatiga qo'shilsin;
- Algoritmni ishlab chiqish uchun avvalo masalaning yechish yo'lini yaxshi tasavvur qilib olish, keyin esa uni formallashtirish, yani aniq qoidalar ketma-ketligi ko'rinishida yozish kerak.

5.2. Algoritmning asosiy xossalari.

“Algoritmash” deganda masalani biri ketidan boshqasini bajariladigan hamda oldingisining natijalari keyingilarining bajarilishida ishlatiladigan bosqichlar ketma-ketligiga keltirish tushuniladi. Ayni paytda bu bosqichlardagi amallarni kompyuter bajara olishi ko'zda tutilishi kerak.

Kengroq ma'noda qaraydigan bo'lsak algoritmash, o'zidan oldingi bosqich – masalani yechish usulini tanlash bosqichi ham, o'zidan keyingi bosqich – kompyuterining xususiyatlarini hisobga olgan holda boshlang'ich, oraliq va natijaviy axborotlarni tuzilishining ifoda shakllarini tanlashni ham o'z ichiga oladi.

Algoritmash bosqichining natijasi masalani yechish algoritmi bo'ladi, yani bu bosqichda masalani yechish algoritmi ishlab chiqiladi. Bunda masalani matematik qo'yilishi va tanlangan usul qidirilayotgan natijani olishga olib keladigan harakatlar ketma-ketligini aniqlash uchun asos bo'lib xizmat qiladi.

Algoritmning 5 ta asosiy xossasi bor.

1. **Diskretlilik (Cheklilik).** Bu xossaning mazmuni algoritmarni doimo chekli qadamlardan iborat qilib bo'laklash imkoniyati mavjudligida. Ya'ni uni chekli sondagi oddiy ko'rsatmalar ketma-ketligi shaklida ifodalash mumkin. Agar kuzatilayotgan jarayonni chekli qadamlardan iborat qilib qo'llay olmasak, uni algoritm deb bo'lmaydi.

2. **Tushunarlilik.** Biz kundalik hayotimizda berilgan algoritmlar bilan ishlayotgan 23soat23 soatlar, mashinalar, dastgohlar, kompyuterlar, turli avtomatik va mexanik qurilmalarni kuzatamiz. Ijrochiga tavsiya etilayotgan ko'rsatmalar uning uchun tushinarli mazmunda bo'lishi shart, aks holda, ijrochi oddiygina amalni ham bajara olmaydi. Bundan tashqari, ijrochi har qanday amalni bajara olmasligi ham mumkin. Har bir ijrochining bajarishi mumkin bo'lgan ko'rsatmalar yoki buyruqlar majmuasi mavjud, u ijrochining ko'rsatmalar tizimi (sistemi) deyiladi. Demak, ijrochi uchun berilayotgan har bir ko'rsatma ijrochining ko'rsatmalar tizimiga mansub bo'lishi lozim. Ko'rsatmalarni ijrochining ko'rsatmalar tizimiga tegishli bo'ladigan qilib ifodalay olishimiz muhim ahamiyatga ega. Masalan, quyi sinfning a'lochi o'quvchisi “son kvadratga oshirilsin” degan ko'rsatmani tushunmasligi natijasida bajara olmaydi, lekin “son o'zini o'ziga ko'paytirilsin” shaklidagi ko'rsatmani bimalol bajaradi, chunki u ko'rsatma mazmunidan ko'paytirish amalini bajarish kerakligini anglaydi.

3. **Aniqlik.** Ijrochiga berilayotgan ko'rsatmalar aniq va mazmunli

bo'lishi zarur. Chunki ko'rsatmadagi noaniqliklar mo'ljaldagi maqsadga erishishga olib kelmaydi. Inson uchun tushunarli bo'lgan "3-4 marta silkitilsin", "5-10 daqiqa qizdirilsin", "1-2 qoshiq solinsin", "tenglamalardan biri echilsin" kabi noaniq ko'rsatmalar kompyuterni qiyin ahvolga solib qo'yadi. Bundan tashqari, ko'rsatmalarning qaysi ketma-ketlikda bajarilishi ham muhim ahamiyatga ega. Demak, ko'rsatmalar aniq berilishi va faqat algoritmda ko'rsatilgan tartibda bajarilishi shart ekan.

4. **Ommaviylik.** Har bir algoritm mazmuniga ko'ra bir turdagi masalalarning barchasi uchun ham o'rinli bo'lishi kerak. Ya'ni masaladagi boshlang'ich ma'lumotlar qanday bo'lishidan qat'i nazar algoritm shu xildagi har qanday masalani yechishga yaroqli bo'lishi kerak. Masalan, ikki oddiy kasrning umumiy maxrajini 24oppish algoritmi, kasrlarni turlicha o'zgartirib bersangiz ham ularning umumiy maxrajlarni aniqlab beraveradi. Yoki uchburchakning yuzini 24oppish algoritmi, uchburchakning qanday bo'lishidan qat'i nazar, uning yuzasini hisoblab beraveradi.

5. **Natijaviylik.** Har bir algoritm chekli sondagi qadamlardan so'ng, albatta, natija berishi shart. Bajariladigan amallar ko'p bo'lsa ham baribir natijaga olib kelishi kerak. Chekli qadamdan so'ng qo'yilgan masalae chimga ega emasligini aniqlash ham natija hisoblanadi. Agar ko'rilyotgan jarayon cheksiz davom etib natija bermasa, uni algoritm deb atay olmaymiz.

5.3.Algoritmni ishlab chiqish.

Algoritmni ishlab chiqishda masalani yechish jarayonini shunday formallashtirish kerakki, bu jarayon etarli darajadagi oddiy qoidalarining chekli ketma-ketligi ko'rinishiga keltirilsin.

Masalan, biz ko'pincha ko'p xonali sonlar ustida asosiy arifmetik amallarni bajarishda vatandoshimiz Al-Xorazmiyning IX asrda yaratgan qoidalarini ishlatamiz. "Algoritm" atamasi ham ana shu buyuk matematik nomidan kelib chiqqan.

Masala yechimining algoritmi ishlab chiqilayotgan davrda asosan uch xil turdagi algoritmlardan foydalanib, murakkab ko'rinishdagi algoritmlar yaratiladi.

Algoritmning asosiy turlariga chiziqli (a), tarmoqlanadigan (b) va takrorlanadigan (c) ko'rinishlari kiradi.

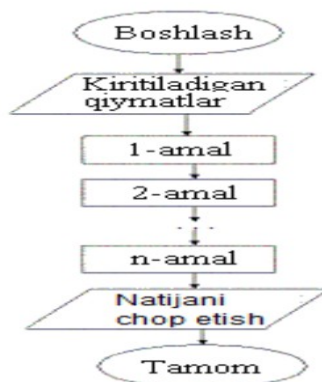
Murakkab masalalarning yechimini olish algoritmlari yuqoridagi turlarining barchasini o'z ichiga olishi mumkin.

Chiziqli turdagi algoritmlarda bloklar biri ketidan boshqasi joylashgan bo'lib, berilgan tartibda bajariladi. Bunday bajarilish tartibi "tabiiy tartib" deb ham yuritiladi.

Har qanday murakkab algoritmni ham uch asosiy struktura yordamida tasvirlash mumkin. Bular ketma-ketlik, ayri va takrorlash strukturalaridir. Ushbu strukturalar asosida chiziqli, tarmoqlanuvchi va takrorlanuvchi hisoblash jarayonlarining algoritmlarini tuzish mumkin. Umuman olganda, algoritmlarni shartli ravishda quyidagi turlarga ajratish mumkin:

- **chiziqli** algoritmlar;

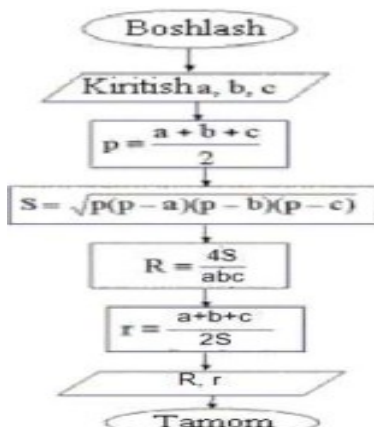
- tarmoqlanuvchi algoritmlar;
- takrorlanuvchi algoritmlar;
- ichma-ich joylashgan takrorlanuvchi algoritmlar;
- rekurrent algoritmlar;
- takrorlanishlar soni oldindan no'malum algoritmlar;
- ketma-ket yaqinlashuvchi algoritmlar.



5.1.-rasm. Chiziqli algoritmlar blok–sxemasining umumiy tuzilishi.

Faqat ketma-ket bajariladigan amallardan tashkil topgan algoritmlarga - *chiziqli* algoritmlar deyiladi. Bunday algoritmni ifodalash uchun ketma-ketlik strukturasi ishlatiladi. Strukturada bajariladigan amal mos keluvchi shakl bilan ko'rsatiladi. Chiziqli algoritmlar blok-sxemasining umumiy tuzilishi 5.1.-rasmda keltirilgan.

1-misol. Uchburchak tomonlarining uzunligi bilan berilgan. Uchburchakka ichki r va tashqi R chizilgan aylanalar radiuslarini hisoblang. Ichki chizilgan aylana radiusi $r = (a+b+c)/2S$, tashqi chizilgan aylana radiusi $R = 4S/abc$ formulalar orqali hisoblanadi. Bu yerda S - uchburchakning yuzi, a, b, c – uchburchak tomonlarining uzunliklari. Masala yechimining blok-sxemasi 5.2.-rasmida keltirilgan.



5.2.-rasm. Uchburchakka ichki va tashqi chizilgan aylanalar radiuslarini hisoblash blok sxemasi

2-misol. Quyida keltirilgan munosabatni hisoblash algoritmini ko'rib chiqaylik. Jarayon amallarni ketma-ket bajarilishidan iborat.

$$z = x^2 + \sqrt{|\sin(x+y)|},$$

bu yerda

$$x = \cos(a - b), y = \ln(a^2 - x^2), a = 0.7, b = 2.1.$$

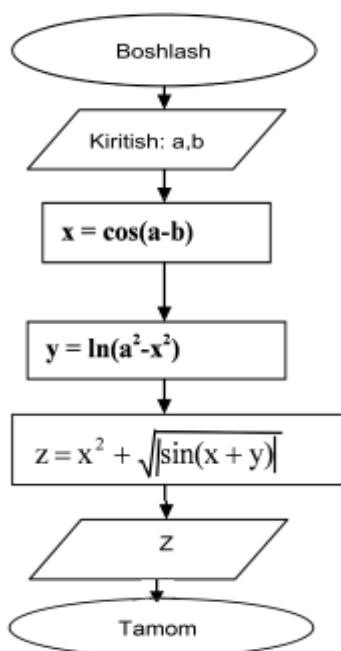
Bunda:

a, b - aniq qiymatga ega bo'lgan boshlang'ich ma'lumotlar;

x, y - oraliq ma'lumotlar;

z - natija.

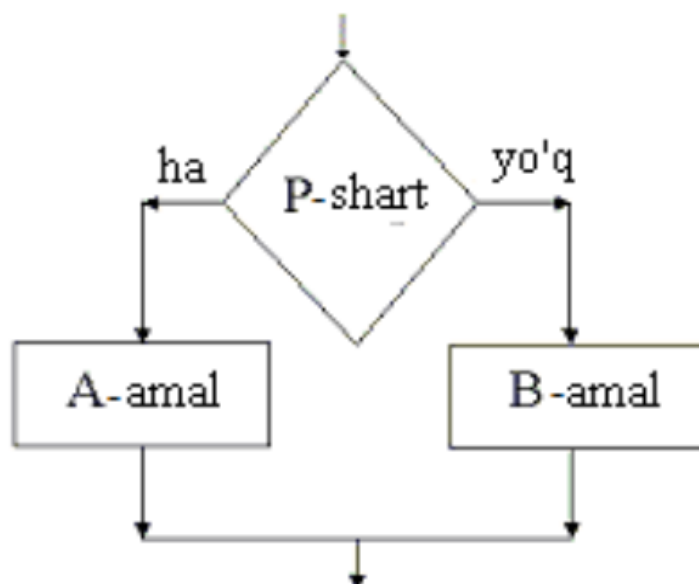
Masalani yechish jarayoni chiziqli hisoblanadi, chunki boshlang'ich ma'lumotlar kiritilgach, munosabatlarning qiymati dasturda joylashgan tartibda hisoblanadi, ya'ni dastlab x, so'ng - y qiymati va nihoyat z natija hisoblanadi. Mazkur jarayonning blok-sxemasi 5.3.-rasmda keltirilgan.



5.3.-rasm. Hisoblash blok-sxemasi

5.4. Tarmoqlanuvchi algoritmlar.

Agar hisoblash jarayoni biror-bir berilgan shartning bajarilishiga qarab turli tarmoqlar bo'yicha davom ettirilsa va hisoblash jarayonida har bir tarmoq faqat bir marta bajarilsa, bunday hisoblash jarayonlari tarmoqlanuvchi algoritmlar deyiladi. Tarmoqlanuvchi algoritmlarni tasvirlash uchun "ayri" tuzilmasi ishlatiladi. Tarmoqlanuvchi tuzilmasi berilgan shartning bajarilishiga qarab ko'rsatilgan tarmoqdan faqat bittasining bajarilishi ta'minlanadi (5.4-rasm).



5.4. Tarmoqlanishning umumiy ko'rinishi.

Berilgan *R-shart* romb figurasi ichida tasvirlanadi. Agar *shart* bajarilsa, "ha" tarmoq bo'yicha A-amal, aks holda (*shart* bajarilmasa) "yo'q" tarmoq bo'yicha V-amal bajariladi.

1-misol. Tarmoqlanuvchi algoritmgga *misol* sifatida quyidagi sodda *masala* keltiriladi:

$$y = \begin{cases} x^2, & \text{agar } x \geq 0 \\ 2x, & \text{aks holda.} \end{cases}$$

Natijaviy qiymat y berilgan x ning qiymatiga bog'liq holda bo'ladi: agar $x \geq 0$ shart *rost* bo'lsa, tarmoq bo'yicha $y = x^2$ munosabatning qiymati, aks holda, $y = 2x$ munosabatning qiymati hisoblanadi. Bu masala bajarilishining so'z bilan ifodalangan algoritmi quyidagicha:

agar $(x \geq 0)$ shart bajarilsa, u holda $u = x^2$, aks holda $u = 2x$.

5.5. Takrorlanuvchi algoritmlar.

Agar biror masalani yechish uchun zarur bo'lgan amallar ketma-ketligining ma'lum bir qismi biror parametrga bog'liq holda ko'p marta qayta bajarilsa, bunday jarayon takrorlanuvchi algoritm deyiladi. Takrorlanuvchi algoritmlarga misol sifatida odatda qatorlarning yig'indisi yoki ko'paytmasini hisoblash jarayonlarini qarash mumkin.

1-misol. Birdan n gacha bo'lgan natural sonlarning yig'indisini hisoblash algoritmini tuzaylik. Masalaning matematik modeli quyidagicha:

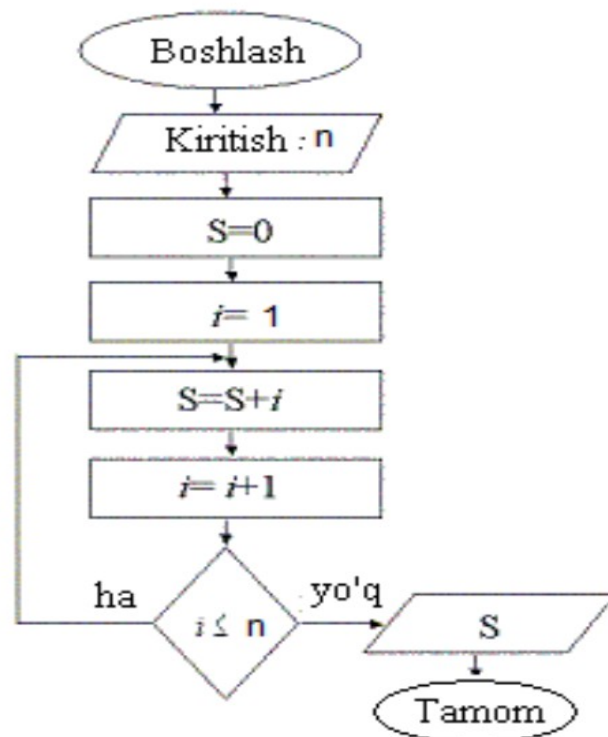
$$S = 1 + 2 + 3 + \dots + n = \sum_{i=1}^n i$$

Bu yig'indini hisoblash uchun, avvalo, natiga boshlang'ich qiymatini $S=0$ va indeksning boshlang'ich qiymatini $i=1$ deb olamiz va joriy amallar $S=S+i$

va $i = i + 1$ hisoblanadi. Bu erda birinchi va ikkinchi qadamlar uchun yig'indi hisoblandi va keyingi qadamda i parametr yana bittaga orttiriladi va navbatdagi qiymat avvalgi yig'indi S ga qo'shiladi. Mazkur jarayon shu tartibda indeksning joriy qiymati $i \leq n$ sharti bajarilmaguncha davom ettiriladi va natijada, izlangan yig'indiga ega bo'lamiz. Ushbu fikrlarni quyidagi so'zlar orqali ifodalangan algoritm bilan ifodalash mumkin:

- 1) kiritish (n);
- 2) $S=0$ - natijaning boshlang'ich qiymati;
- 3) $i=1$ - indeksning boshlang'ich qiymati;
- 4) $S=S+i$ - natijaning joriy qiymatini hisoblang;
- 5) $i=i+1$ - indeksning joriy qiymatini hisoblang;
- 6) agar $(i \leq n)$ sharti tekshirilsin va u bajarilsa $\rightarrow$ (4) ;
- 7) muhrlash (S).

Bu jarayonga mos keladigan blok-sxemaning ko'rinishi quyida tasvirlangan



5.5-rasm. 1 dan n -gacha bo'lgan sonlar yig'indisini hisoblash blok-sxemasi.

Mavzu bo'yicha savollar to'plami:

1. "Algoritm" tushunchasining ta'rifini bering.
2. Blok-sxemaning ta'rifini bering.
3. Blok-sxema asosiy figuralar ma'nosini ayting bering.
4. Qanday algoritmlar chiziqli deb ataladi.
5. Qanday algoritmlar tarmoqlanuvchi deb ataladi?
6. Qanday algoritmlar takrorlanuvchi deb ataladi?
7. Dastur tuzilishida qaysi bloklardan foydalaniladi?
8. Algoritmndagi xatolar qanday qilib to'g'rilanadi?

- 9.Chiziqli algoritmning ishlash prinsiplari.
- 10.Shartli algoritmning ishlash prinsiplari.
- 11.Takrorlanuvchi algoritmning ishlash usullari.
- 12.Ichma-ich joylashgan takrorlanuvchi algoritmlar nimadan iborat?

Mavzu bo'yicha glossariy:

Algoritm - masalani yechish uchun bajarilishi lozim bo'lgan amallar ketma-ketligini aniq tavsiflaydigan qoidalar tizimi.

Chiziqli algoritmlar - faqat ketma-ket bajariladigan amallardan tashkil topgan algoritmlar.

Axborotni muhofaza etish - axborot borasidagi xavfsizlikka tahdidlarning oldini olish va ularning oqibatlarini bartaraf etish chora-tadbirlari.

Axborot resurslari - alohida hujjatlar, hujjatlarning alohida to'plamlari, axborot tizimlaridagi (kutubxonalardagi, arxivlardagi, fondlardagi, ma'lumotlar banklaridagi va boshqa axborot tizimlaridagi) hujjatlar va hujjatlarning to'plamlari.

Axborot sohasi - sub'ektlarning axborotni yaratish, qayta ishlash va undan foydalanish bilan bog'liq faoliyati sohasi.

6-ma'ruza. C++ algoritmik tiliga kirish.

Reja:

- 1.C++ algoritmik tili alfaviti va xizmatchi so'zlari.**
- 2.C++ tilida o'zgaruvchilarni ta'riflash.**
- 3.Maxsus belgilar va satrli konstantalar.**

Keyingi yillarda amaliy dasturchilarga juda ko'p integratsion dastur tuzish muhitlari taklif etilayapti. Bu muhitlar u yoki bu imkoniyatlari bilan bir-biridan farq qiladi. Aksariyat dasturlashtirish muhitlarining fundamental asosi C++ tiliga borib taqaladi. C++ tili B.Straustrup tomonidan yaratilgan. Amerika milliy standartlar instituti (American National Standards Institute – ANSI) rahbarligi ostidagi Ctandartlarni akkreditivlash komiteti (Accredited Standards Committee) C++ tilining xalqaro standartini tuzdi.

C++ standarti ayni vaqtda ISO – International Standards Organization (Ctandartlash bo'yicha xalqaro tashkilot) standarti deb ham nomlanadi.

Vaqt o'tishi bilan dasturchilar oldiga quyilgan masalalar o'zgarib boryapti. Bundan yigirma yil oldin dasturlar katta hajmdagi ma'lumotlarni qayta ishlash uchun tuzilar edi. Bunda dasturni yozuvchi ham, uning foydalanuvchisi ham kompyuter sohasidagi bilimlar bo'yicha professional bo'lishi talab etilardi. Hozirda esa ko'pgina o'zgarishlar ro'y berdi. Kompyuter bilan ko'proq uning apparat va dasturiy ta'minoti haqida tushunchalarga ega bo'lmagan kishilar ishlashyapti. Kompyuter odamlar tomonidan uni chuqur o'rganish vositasi emas, ko'proq o'zlarining oldilariga qo'yilgan, o'zlarining ishlariga tegishli bo'lgan muamolarini yechish instrumenti bo'lib qoldi.

Dasturlashga talabni o'zgarishi nafaqat tillarning o'zgarishiga balki uni yozish texnologiyasini ham o'zgarishiga olib keldi. Dasturlash evolyutsiyasi

tarixida ko'pgina bosqichlar bo'lishiga qaramay biz bu qo'llanmada protsedurali dasturlashdan ob'ektlarga mo'ljallangan dasturlashga o'tishni qaraymiz.

Alfavit va xizmatchi so'zlar:

Alfavit. C++ alfavitiga quyidagi simvollar kiradi.

Katta va kichik lotin alfaviti harflari (A,B,...,Z,a,b,...,z)

Raqamlar: 0,1,2,3,4,5,6,7,8,9

Maxsus simvollar: “ , { } | [] () + - / % \ ; ‘ . : ? < = > _ ! & * # ~ ^

Ko'rinmaydigan simvollar (“umumlashgan bo'shlik simvollar”).

Leksemalarni o'zaro ajratish uchun ishlatiladigan simvollar (misol uchun bo'shlik, tabulyatsiya, yangi qatorga o'tish belgilari).

Izoxlarda, satrlarda va simvolli konstantalarda boshqa literallar, masalan rus harflari ishlatilishi mumkin.

C++ tilida besh xil turdagi leksemalar ishlatiladi: erkin tanlanadigan va ishlatiladigan identifikatorlar, xizmatchi so'zlar, konstantalar(konstanta satrlar), amallar(amallar belgilari), ajratuvchi belgilar.

Identifikator. Identifikatorlar lotin harflari,ostki chiziq belgisi va sonlar ketma ketligidan iborat bo'ladi. Identifikator lotin harfidan yoki ostki chizish belgisidan boshlanishi lozim.

Misol uchun:

A1, _MAX, adress_01, RIM, rim

Katta va kichik harflar farqlanadi, shuning uchun oxirgi ikki identifikator bir biridan farq qiladi.

Borland kompilyatorlaridan foydalanilganda nomning birinchi 32 harfi ,ba'zi kompilyatorlarda 8 ta harfi inobatga olinadi. Bu holda NUMBER_OF_TEST va NUMBER_OF_ROOM identifikatorlari bir biridan farq qilmaydi.

Xizmatchi so'zlar. Tilda ishlatiluvchi ya'ni dasturchi tomonidan o'zgaruvchilar nomlari sifatida ishlatish mumkin bo'lmagan identifikatorlar xizmatchi so'zlar yoki kalit so'zlar deyiladi. Kompilyatorning texnik dokumentatsiyasida barcha zahiralangan so'zlarning ro'yxati turadi.

O'zgaruvchilarni ta'riflash. C++ tilida o'zgaruvchini aniqlash uchun kompyuterga uning tipi (masalan, int, char yoki float) hamda ismi xaqida haqida ma'lumot beriladi. Bu axborot asosida kompilyatorga o'zgaruvchi uchun qancha joy ajratish lozim va bu o'zgaruvchida qanday turdagi qiymat saqlanishi mumkinligi haqida ma'lumot aniq bo'ladi. O'zgaruvchi nomi identifikator bo'lib, xizmatchi so'zlardan farqli bo'lishi kerak.

Har bir yacheyka bir bayt o'lchovga ega. Agar o'zgaruvchi uchun ko'rsatilgan tip 4 baytni talab qilsa, uning uchun to'rtta yacheyka ajratiladi. Aynan o'zgaruvchini tipiga muvofiq ravishda kompilyator bu o'zgaruvchi uchun qancha joy ajratish kerakligini aniqlaydi.

Kompyuterda qiymatlarni ifodalash uchun bitlar va baytlar qo'llaniladi va xotira baytlarda hisoblanadi.

6.2.C++ tilida o'zgaruvchilarni ta'riflash.

O'zgarmlar turlari. O'zgaruvchilar kabi o'zgarmlar ham ma'lumotlarni saqlash uchun mo'ljallangan xotira yacheykalarini o'zida ifodalaydi.

O'zgaruvchilardan farqli ravishda ular dasturni bajarilishi jarayonida qiymati o'zgarmaydi. O'zgarmas e'lon qilinishi bilan unga qiymat berish lozim, keyinchalik bu qiymatni o'zgartirib bo'lmaydi.

C++ tilida ikki turdagi, literal va nomlangan o'zgarmaslar aniqlangan.

Literalli o'zgarmaslar to'g'ridan-to'g'ri dasturga kiritiladi. Masalan:

```
int myAge =39;
```

Bu ifodada MyAge int tipidagi o'zgaruvchi, 39 soni esa literal o'zgarmasdir.

Belgili o'zgarmaslar. Belgili o'zgarmaslar odatda bir bayt joyni egallaydi va bu 256 xil belgini saqlash uchun yetarlidir. Char tipi qiymatlarini 0..255 sonlar to'plamiga yoki ASCII belgilar to'plamiga interpretatsiya qilish mumkin.

ASCII belgilari deganda kompyuterlarda qo'llaniladigan standart belgilar to'plami tushuniladi. ASCII - bu American Standard Code for Information Interchange (Amerikaning axborot almashinishi uchun standart kodi) degan ma'noni anglatadi.

Maxsus belgilar. C++ kompilyatorida tekstlarni formatlovchi bir nechta maxsus belgilardan foydalaniladi. (Ulardan eng ko'p tarqalgani jadvalda keltirilgan). Bu belgilarni dasturda ishlatishda «teskari slesh»dan foydalanamiz. Teskari sleshdan keyin boshqaruvchi belgi yoziladi. Masalan, tabulyatsiya belgisini dasturga qo'yish uchun quyidagicha yozuvni yozish kerak.

```
char tab ="\t";
```

Bu misoldagi char tipidagi o'zgaruvchi \t qiymatini qabul qiladi. Maxsus belgilar axborotlarni ekranga, faylga va boshqa chiqarish qurilmalariga chiqarishda formatlash uchun qo'llaniladi.

Maxsus '\ simvolidan boshlangan simvollar eskeyp simvollar deyiladi. Simvolli konstanta qiymati simvolning kompyuterda qabul qilingan sonli kodiga tengdir.

ESC (eskeyp) simvollar jadvali:

Yozilishi	Ichki kodi	Simvoli(nomi)	Ma'nosi
\a	0x07	bel (audible bell)	Tovush signali
\b	0x08	bs (bascpase)	Bir qadam qaytish
\f	0x0C	ff (form feed)	Saxifani o'tkazish
\n	0x0A	lf (line feed)	Qatorni o'tkazish
\r	0x0D	cr (carriage return)	Karetkani qaytarish
\t	0x09	ht (horizontal tab)	Gorizontal tabulyatsiya
\v	0x0B	vt (vertical tab)	Vertikal tabulyatsiya
\\	0x5C	\ (bacslash)	Teskari chizik
\'	0x27	' (single out)	Apostrif (oddiy qavs)
\"	0x22	“ (double quote)	Ikkilik qavs
\?	0x3F	? (question mark)	Savol belgisi
\000	000	ixtiyoriy (octal number)	Simvol sakkizlik kodi
\xhh	0xhh	ixtiyoriy(hex number)	Simvol unoltilik kodi

Ma'lumotlarning butun son turi. Butun sonlar o'nlik, sakkizlik yoki o'n oltilik sanoq sistemalarida berilishi mumkin.

O'nlik sanoq sistemasida butun sonlar 0-9 raqamlari ketma ketligidan iborat bo'lib, birinchi raqami 0 bo'lishi kerak emas.

Sakkizlik sanoq sistemasida butun sonlar 0 bilan boshlanuvchi 0-7 raqamlaridan iborat ketma ketlikdir.

O'n oltilik sanoq sistemasida butun son 0x yoki 0X bilan boshlanuvchi 0-9 raqamlari va a-f yoki A-F harflaridan iborat ketma ketlikdir.

Masalan 15 va 22 o'nlik sonlari sakkizlikda 017 va 026, o'n oltilikda 0xF va 0x16 shaklda tasvirlanadi.

Ma'lumolarning uzun butun son turi.

Oxiriga l yoki L harflari qo'yilgan o'nlik, sakkizlik yoki o'n oltilik butun son.

Ma'lumotlarning ishorasiz (unsigned) butun son turi:

Oxiriga u yoki U harflari qo'yilgan o'nlik, sakkizlik yoki o'n oltilik oddiy yoki uzun butun son.

Ma'lumotlarning xaqiqiy son turi. Ma'lumotlarning xaqiqiy son turi olti qismdan iborat bo'lishi mumkin: butun qism, nuqta, kasr qism, e yoki E belgisi, o'nlik daraja, F yoki f suffikslari.

Masalan : 66. .0 .12 3.14F 1.12e-12

Ma'lumolarning uzun xaqiqiy son turi :

Oxiriga L yoki l suffikslari quyilgan xaqiqiy son.

Masalan: 2E+6L;

Mantiqiy konstanta. Mantiqiy konstantalar true(rost) va false(yolg'on) qiymatlardan iborat. Ichki ko'rinishi false – 0, ixtiyoriy boshqa qiymat true deb qaraladi.

6.3.Maxsus belgilar va satrli konstantalar.

Satrli konstanta. Satrli konstantalar C++ tili konstantalariga kirmaydi, balki leksemalari aloxida tipi hisoblanadi. SHuning uchun adabiyotda satrli konstantalar satrli leksemalar deb ham ataladi.

Satrli konstanta bu ikkilik qavslarga olingan ixtiyoriy simvollar ketma ketligidir. Misol uchun "Men satrli konstantaman".

Satrlar orasiga eskeyp simvollar ham kirishi mumkin. Bu simvollar oldiga \ belgisi qo'yiladi. Misol uchun:

"\n Bu satr \n uch katorga \n joylashadi".

Satr simvollarini xotirada ketma ket joylashtiriladi va har bir satrli konstanta oxiriga avtomatik ravishda kompilyator tomonidan '\0' simvoli qo'shiladi. SHunday satrning xotiradagi xajmi simvollar soni+1 baytga tengdir.

Ketma-ket kelgan va bo'shliq, tabulyatsiya yoki satr oxiri belgisi bilan ajratilgan satrlar kompilyatsiya davrida bitta satrga aylantiriladi. Misol uchun:

"Salom" "Toshkent "

satrlari bitta satr deb qaraladi.

"Salom Toshkent"

Bu qoidaga bir necha qatorga yozilgan satrlar ham bo'ysinadi. Misol uchun :

"O'zbekistonga "

"baxor "

"keldi"

qatorlari bitta qatorga mos:

"O'zbekistonga baxor keldi"

Agar satrda '\ ' belgisi uchrasa va bu belgidan so'ng to '\n' satr oxiri belgisigacha bo'shlik belgisi kelsa bu bo'shlik belgilari '\ ' va '\n' belgisi bilan birga satrdan o'chiriladi. Satrning o'zi keyingi satrda kelgan satr bilan qo'shiladi.

"O'zbekistonga \ "

" baxor\

" keldi"

qatorlari bitta qatorga mos:

"O'zbekistonga baxor keldi"

Nomlangan o'zgarmlar. Belgili o'zgarmlar – bu nomga ega bo'lgan o'zgarmlardir. C++ tilida belgili o'zgarmlarni aniqlashning ikki usuli mavjud:

#define direktivasi yordamida o'zgarmlarni aniqlash.

const kalitli so'zi orqali o'zgarmlarni aniqlash.

An'anaviy usul hisoblangan #define direktivasi orqali o'zgarmlarni aniqlashni quyidagi misolda ko'rishimiz mumkin.

```
#define StudentsPerClass 15
```

Bu holda StudentsPerClass o'zgarmlar hech qanday tipga tegishli bo'lmaydi.

Preprocessor StudentsPerClass so'ziga duch kelganida uni 15 literaliga almashtiradi.

C++ tilida #define direktivasidan tashqari o'zgarmlarni aniqlashning nisbatan qulayroq bo'lgan yangi usuli ham mavjud:

```
const unsigned short int StudentsPerClass=15
```

Bu misolda ham belgili konstanta StudentsPerClass nomi bilan aniqlanayapti va unga unsigned short int tipi berilyapti. Bu usul bir qancha imkoniyatlarga ega bo'lib u sizning dasturingizni keyingi himoyasini yengillashtiradi. Bu o'zgarmlarni oldingisidan eng muhim afzalligi uning tipga egaligidir.

Bu konstantalar qiymatlarini dasturda o'zgartirish mumkin emas. Konstantalar nomlari dasturchi tomonidan kiritilgan va xizmatchi so'zlardan farqli bo'lgan identifikatorlar bo'lishi mumkin. Odatda nom sifatida katta lotin harflari va ostiga chizish belgilari kombinatsiyasidan iborat identifikatorlar ishlatiladi. Nomlangan konstantalar quyidagi shaklda kiritiladi:

```
const tip konstanta_nomi=konstanta_qiymati.
```

Misol uchun:

```
const double EULER=2.718282;
```

```
const long M=99999999;
```

```
const R=765;
```

Oxirgi misolda konstanta tipi ko'rsatilmagan, bu konstanta int tipiga tegishli deb hisoblanadi.

Belgili o'zgarmlarni literal o'zgarmlarga nisbatan ishlatish qulayroqdir.

Chunki agarda bir xil nomli literalli o'zgaruvchini qiymatini o'zgartirmoqchi bo'lsangiz butun dastur bo'yicha uni o'zgartirishga to'g'ri keladi, belgili o'zgarmaslarni esa faqatgina birining qiymatini o'zgartirish yetarli.

To'plam o'zgarmaslari. Bunday o'zgarmaslarni hosil qilish uchun yangi berilgan ma'lumotlar tiplari tuziladi va undan so'ng bu tipga tegishli o'zgarmasli qiymatlar to'plami bilan chegaralangan o'zgaruvchilar aniqlanadi.

Sanovchi konstantalar enum xizmatchi so'zi yordamida kiritilib, int tipidagi sonlarga qulay so'zlarni mos qo'yish uchun ishlatiladi.

Misol uchun:

```
enum{one=1,two=2,three=3};
```

Agar son qiymatlari ko'rsatilmagan bo'lsa eng chapki so'zga 0 qiymati berilib kolganlariga tartib bo'yicha o'suvchi sonlar mos qo'yiladi:

```
enum{zero,one,two};
```

Mavzu bo'yicha savollar to'plami:

1. Yuqori darajadagi dasturlash tillari nima?
2. Kompilyatsiya jarayonini necha bosqichdan tashkil topadi?
3. Qanday belgilar bilan tugagan barcha belgilar ketma-ketligi izoh hisoblanadi?
4. C++ tilining kalit so'zlariga qaysilar kiradi?
5. Protsessor registrlarini belgilash uchun qaysi so'zlar ishlatiladi?
6. Kompilyatsiya jarayoni nima?
7. Kiritilgan qiymatni o'zgaruvchi turiga mos kelishini qanday tekshirish mumkin?
8. Identifikator nima?
9. Sintaksis – qanday tilning qoidalar?
10. C++ instruksiyalari qanday belgi bilan tugallanishi zarur?
11. Tilning ma'nosini beruvchi qoidalar to'plami qanday nomlanadi?

Mavzu bo'yicha glossariy:

Algoritm - masalani yechish uchun bajarilishi lozim bo'lgan amallar ketma-ketligini aniq tavsiflaydigan qoidalar tizimi.

Chiziqli algoritmlar - faqat ketma-ket bajariladigan amallardan tashkil topgan algoritmlar.

Kompilyator – yuqori darajali dasturlash tilida yozilgan programma kodini mashina tiliga o'giradigan dastur.

Kompilyatsiya jarayoni – dastur kodini mashina tiliga o'girish jarayoni.

O'zgaruvchi – xotiraning nomlangan qismi bo'lib, o'zida ma'lum bir toifadagi qiymatlarni saqlaydi.

O'zgarmaslar – hisoblash jarayonida qiymatini o'zgartirmaydigan kattalik.

Identifikator – dasturchi tomonidan dastur elementlari (funktsiya, o'zgaruvchilar, o'zgarmaslar,...) uchun ixtiyoriy tanlangan nom.

7-ma'ruza. Asosiy ma'lumot turlari.

Reja:

1.C++ tilida oddiy dastur.

2.cout ob'ekti haqida qisqacha ma'lumot.

3.Izohlar.

C++ tilida tuzilgan dastur ob'ektlar, funktsiyalar, o'zgaruvchilar va boshqa elementlardan tashkil topadi.

C++ tilida yozilgan dastur tarkibini o'rganish uchun oddiygina SALOM.CPP dasturi bilan tanishamiz. Bu dastur kichik bo'lishiga qaramasdan bizda qiziqish uyg'otuvchi bir nechta elementdan iboratdir.

SALOM.CPP dasturi misolida C++ tilida tuzilgan dastur qismlarini namoyish qilish

```
#include <iostream.h>
int main( )
{
    cout << "Salom!\n";
    return 0;
}
```

Natija:

Salom!

– satrda iostream fayli joriy faylga biriktilyapti. Dasturda birinchi funta (#) belgisi joylashgan. U preprotssorga signal uzatadi.

Kompilyatorning har safar ishga tushirilishida preprotssor ham ishga tushiriladi. U dasturdagi funta (#) belgisi bilan boshlanuvchi qatorlarni o'qiydi.

include - preprotssorning komandasi bo'lib, u quyidagicha tarjima qilinadi: «Bu komandani ortidan fayl nomi keladi. Ushbu nomdagi faylni topish va fayldagi mazmunni dasturning joriy qismiga yozish lozim».

Burchakli qavs ichidagi faylni mos fayllar joylashtirilgan barcha papkalardan izlash lozimligini ko'rsatadi. Agarda kompilyator to'g'ri sozlangan bo'lsa burchakli kavslar iostream faylini sizning kompilyatoringiz uchun fayllarni o'zida saqlovchi papkadan izlashi kerakligini ko'rsatadi. Iostream (input–output stream – kiritish–chiqarish oqimi) faylida ekranga ma'lumotlarni chiqarish jarayonini ta'minlaydigan cout ob'ekti aniqlangan. Birinchi qator bajarilgandan so'ng iostream fayli joriy dasturga xuddi uning mazmunini qo'l bilan yozganimizdek biriktiriladi. Preprotssor kompilyatordan keyin yuklanadi va funt (#) belgisi bilan boshlanuvchi barcha qatorlarni bajaradi, dastur kodlarini kompilyatsiyaga tayyorlaydi.

Dasturning asosiy kodi main() funktsiyasini chaqirish bilan boshlanadi. C++ tilidagi har bir dastur main() funktsiyasini o'zida saqlaydi. Funktsiya bu bir yoki bir necha amalni bajaruvchi dastur blokidir. Odatda funktsiyalar boshqa funktsiyalar orqali chaqiriladi, lekin main() funktsiyasi alohida xususiyatga ega bo'lib u dastur ishga tushirilishi bilan avtomatik tarzda chaqiriladi.

main() funktsiyasini boshqa funktsiyalar kabi qaytaradigan qiymati tipini

e'lon qilish lozim. SALOM.cpp dasturida main() funksiyasi int (integer – butun so'zidan olingan) tipli qiymat qaytaradi, ya'ni bu funksiya ishini tugatgandan so'ng operatsion sistemaga butun sonli qiymat qaytaradi. Operatsion sistemaga qiymat qaytarish unchalik muhim emas, umuman sistema bu qiymatdan foydalanmaydi, lekin C++ tili standarti main() funksiyasi barcha qoidalarga muvofiq e'lon qilinishini talab qiladi.

Ayrim kompilyatorlar main() funksiyasini void tipidagi qiymat qaytaradigan qilib e'lon qilish imkonini beradi. C++ da bundan foydalanmaslik kerak, chunki hozirda bunday uslub eskirgan. main() funksiyasini int tipini qaytaradigan qilib aniqlash lozim va buning hisobiga funktsiyaning oxiriga return 0 ifodasi yoziladi.

Barcha funktsiyalar ochiluvchi figurali qavs ({} bilan boshlanadi va {}) yopiluvchi qavs bilan tugaydi. Figurali qavslarni ichida joylashgan barcha satrlar funksiya tanasi deb aytiladi.

Bizning oddiy dasturimizda sout ob'ekti ekranga ma'lumotni chiqarish uchun qo'llaniladi. cin va cout ob'ektlari mos ravishda ma'lumotlarni kiritish (masalan, klaviatura orqali) va ularni chiqarish (ekranga chiqarish) uchun qo'llaniladi.

7.2.cout ob'ekti haqida qisqacha ma'lumot.

sout ob'ekti haqida qisqacha ma'lumot beramiz. Ekranga ma'lumotni chiqarish uchun cout so'zini, undan so'ng chiqarish operatorini (<<) kiritish lozim. C++ kompilyatori (<<) belgisini bitta operator deb qaraydi. Quyidagi dasturni tahlil qilamiz.

cout ob'ektini qo'llanilishiga misol

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Bu son 5 ga teng:" << 5 << "\n";
    cout << "endl operatori ekranda yangi";
    cout << " satrga o'tish amalini bajaradi";
    cout << endl;
    cout << "Bu katta son:\t" << 70000 << endl;
    cout << "Bu 5 va 8 sonlarining yig'indisi:" << "\t" << 8+5 << endl;
    cout << "Bu kasr son:\t" << (float)5/8 << endl;
    cout << "Bu esa juda katta son:" << "\t";
    cout << (double) 7000*7000 << endl; return 0;
};
```

Natija:

Bu son 5 ga teng: 5

endl operatori ekranda yangi satrga o'tish amalini bajaradi

Bu katta son: 70000

Bu 5 va 8 sonlarining yig'indisi: 13

Bu kasr son: 0.625

Bu esa juda katta son: 4.9e+07

Ayrim kompilyatorlarda sout ob'ektidan keyin matematik operatsiyalarni bajarish uchun figurali qavslarni ishlatish talab qilinadi.

endl operatori end line (catr oxiri) degan so'zdan olingan bo'lib «end-el» deb o'qiladi.

7.3.Izohlar.

Siz dastur yozayotgan vaqtingizda nima ish qilmoqchi ekanligingiz doimo aniq bo'ladi. Lekin bir oydan so'ng bu dasturga qaytish lozim bo'lsa dasturga tegishli detallar va ularning vazifalari nimadan iborat ekanligini bilmasligingiz mumkin.

Dasturni butunlay xotirangizdan o'chirib yubormaslik va boshqalarga ham tushunarli bo'lishi uchun izohlardan foydalanish lozim. Izohlar kompilyator tomonidan tushirib qoldiriladigan dasturning alohida satrida yoki butun bir blokida qo'llaniladi. Quyidagi listingni ko'rib chiqamiz.

Salom.cpp dasturi misolida izohlarni namoyish qilish.

```
#include <iostream>
using namespace std;
Int main()
{
cout << "Salom!\n";
/* bu izox toki izoxning
oxirini ko'rsatuvchi belgi, ya'ni yulduzcha
va slesh belgisi uchramaguncha davom etadi */
cout << "Bu kommentariy tugadi\n";
// bu izox satrni oxirida tugaydi.
// Ikkita sleshdan so'ng xech kanday tekst
// bulmasligi mumkin.
return 0;
}
```

Natija:

Salom

Bu kommentariy tugadi

Ifoda.C++ tilida ifodalar biror bir hisoblash natijasini qaytaruvchi boshqa ifodalar ketma-ketligini boshqaradi yoki hech nima qilmaydi (nol ifodalar).

C++ tilida barcha ifodalar nuqtali vergul bilan yakunlanadi. Ifodaga misol qilib o'zlashtirish amalini olish mumkin.

```
x=a+b;
```

Algebradan farqli ravishda bu ifoda $x + b$ ga teng ekanligini anglatmaydi. Bu ifodani quyidagicha tushunish kerak:

a va b o'zgaruvchilarni qiymatlarini yig'ib natijasini x o'zgaruvchiga beramiz yoki x o'zgaruvchiga $a+b$ qiymatni o'zlashtiramiz. Bu ifoda birdaniga ikkita amalni bajaradi, yig'indini hisoblaydi va natijani o'zlashtiradi. Ifodadan

so'ng nuqtali vergul qo'yiladi. (=) operatori o'zidan chap tomondagi operandga o'ng tomondagi operandlar ustida bajarilgan amallar natijasini o'zlashtiradi.

Bo'sh joy (probel) belgisi. Bo'sh joy belgilariga nafaqat probel, balki yangi satrga o'tish va tabulyatsiya belgilari ham kiradi. Yuqorida keltirilgan ifodani quyidagicha ham yozish mumkin:

$$x = a$$
$$+ b ;$$

Bu variantda keltirilgan ifoda ko'rimsiz va tushunarsiz bo'lsa ham to'g'ridir.

Bo'sh joy belgilari dasturning o'qilishligini ta'minlaydi.

Mavzu bo'yicha savollar to'plami:

1. *int* turidagi son, odatda, nima uchun ishlatiladi?
2. Agar dasturda sintaktik xatolar bo'lsa, kompilyator bu haqida xabar beradimi?
3. Arifmetik operatorlar bajarilish ketma-ketligi qoidasi qanday qoida?
4. Figurali qavslar nima uchun ishlatiladi?
5. Vergul (","), odatda, nima uchun ishlatiladi?
6. "*double*" kalit so'zi nima uchun ishlatiladi?
7. Haqiqiy son turi nima?
8. Berilganlarning strukturalashgan turlari qanday?
9. Butun son turlari nima?
10. Suzuvchi nuqtali turlar nima?
11. Sanab o'tiluvchi tur nima?
12. Haqiqiy sonlar qanday kalit so'z bilan e'lon qilinadi?

Mavzu bo'yicha glossariy:

Inkrement - C++ tilida o'zgaruvchi qiymatini birga oshirish.

Dekrement - C++ tilida o'zgaruvchi qiymatini birga kamaytirish.

Chiziqli algoritmlar - faqat ketma-ket bajariladigan amallardan tashkil topgan algoritmlar.

Kompilyator – yuqori darajali dasturlash tilida yozilgan programma kodini mashina tiliga o'giradigan dastur.

Kompilyatsiya jarayoni – dastur kodini mashina tiliga o'girish jarayoni.

O'zgaruvchi – xotiraning nomlangan qismi bo'lib, o'zida ma'lum bir toifadagi qiymatlarni saqlaydi.

O'zgarmaslar – hisoblash jarayonida qiymatini o'zgartirmaydigan kattalik.

Identifikator – dasturchi tomonidan dastur elementlari (funktsiya, o'zgaruvchilar, o'zgarmaslar,...) uchun ixtiyoriy tanlangan nom.

8-ma'ruza.Operatorlar.

Reja:

- 1.C++ da operatorlar.
2. Mantiqiy operatorlar
- 3.Qo'shma operatorlar.
- 4.Shart operatorlari.
5. Takrorlash operatorlari

1.C++ da operatorlar.

C++ tilida dastur tuzilishi tushuntirish uchun sodda dastur keltiramiz [3, 4-8 b.].

```
#include <iostream.h>          // sarlavha faylni qo'shish
int main()                     // bosh funksiya tavsifi
{
    cout<<"Kompyuter olami!\n"; // satrni chop etish
    return 0;                  // funksiya qaytaradigan qiymat
}
```

Dasturning 1-satrida “*#include*” preprotssessor ko'rsatmasi bo'lib, dastur kodiga standart oqimli o'qish-yozish funksiyalari va uning o'zgaruvchilari e'loni joylashgan «*iostream.h*» sarlavha faylini qo'shadi (mnemonika: 'i' (*input*) - kiritish (o'qish); 'o' (*ouput*) - chiqarish (yozish); “*stream*” - oqim; 'h' (*head*) – sarlavha). Kelishuv bo'yicha standart oqim ekranga chiqarish hisoblanadi. Keyingi qatorlarda dasturning yagona, asosiy funksiyasi - *main()* funksiyasi tavsifi keltirilgan. Shuni qayd etish kerakki, C++ dasturida, albatta, *main()* funksiyasi bo'lishi shart va dastur shu funksiyani bajarish bilan o'z ishini boshlaydi. Funksiya nomi oldidagi “*int*” kalit so'zi funksiya bajarilishi natijasida qaytaraladigan qiymat turini bildiradi. Bunday holat funksiyaning matematikadagi tavsifiga mos keladi. Keyingi qatordan funksiya tanasi - figurali qavsga olingan amallar ketma-ketligi keladi. Bizning holda funksiya tanasi ikkita amaldan iborat. Birinchisi, konsol rejimida belgilar ketma-ketligini oqimga chiqarish amali qo'llangan. Buning uchun «*iostream.h*» sarlavha faylida aniqlangan *cout* obyektidan foydalanilgan. Uning formati quyidagi ko'rinishda:

cout << <ifoda>;

Bu yerda “<<” – ma'lumot uzatish amali («..ga joylashtir»), <ifoda> sifatida o'zgaruvchi yoki sintaksisi to'g'ri yozilgan va qandaydir qiymat qabul qiluvchi til ifodasi kelishi mumkin (keyinchalik, burchak qavs ichiga olingan o'zbekcha satr ostini til tarkibiga kirmaydigan tushuncha deb qabul qilish kerak). Ikkinchisi, funksiya o'z ishini tugatganligini anglatuvchi va undan chiqishni amalga oshiruvchi “**return 0;**” operatoridir. Odatda, bajarilishi normal tugagan funksiyalar operatsion sistemaga 0 qiymatini qaytaradi. Shu qoidagi rioya qilgan

holda dastur ham 0 qiymatini qaytaradi.

Bajariluvchi dasturni hosil qilish uchun dastur matni kompilyatsiya qilinishi kerak. Kompilyatsiya jarayonining o'zi ham ikkita bosqichdan tashkil topadi. Boshida preprotssessor ishlaydi, u matndagi kompilyatsiya direktivalarini bajaradi, xususan *#include* direktivasi bo'yicha ko'rsatilgan kutubxonalardan C++ tilida yozilgan modullarni dastur tarkibiga kiritadi. Shundan so'ng kengaytirilgan dastur matni kompilyatorga uzatiladi. Kompilyator o'zi ham dastur bo'lib, uning uchun kiruvchi ma'lumot bo'lib, C++ tilida yozilgan dastur matni hisoblanadi. Kompilyator dastur matnini leksema (atomar) elementlarga ajratadi va uni leksik, keyinchalik sintaktik tahlil qiladi. Leksik tahlil jarayonida u matnni leksemalarga ajratish uchun «probel ajratuvchisini» ishlatadi. Probel ajratuvchisiga - probel belgisi (' '), '\t' - tabulyatsiya belgisi, '\n' - keyingi qatorga o'tish belgisi, boshqa ajratuvchilar va izohlar kiradi.

Dastur matni tushunarli bo'lishi uchun izohlar ishlatiladi. Izohlar dastur amal qilishiga hech qanday ta'sir qilmaydi.

C++ tilida izohlar ikki ko'rinishda yozilishi mumkin. Birinchisida */** dan boshlanib, **/* belgilar oralig'ida joylashgan barcha belgilar ketma-ketligi izoh hisoblanadi, ikkinchisi «*satriy izoh*» deb nomlanadi va u *//* belgilardan boshlangan va satr oxirigacha yozilgan belgilar ketma-ketligi bo'ladi. Izohning birinchi ko'rinishida yozilgan izohlar bir necha satr bo'lishi va ulardan keyin C++ operatorlari davom etishi mumkin.

Berilganlarni standart oqimdan (odatda, klaviaturadan) o'qish quyidagi formatda amalga oshiriladi:

```
cin >> <o'zgaruvchi>;
```

Bu yerda <o'zgaruvchi> oqimdan qiymat qabul qiluvchi o'zgaruvchining nomi.

```
int n;  
cout << " n = ";  
cin >> n;
```

Butun turdagi *n* o'zgaruvchisi kiritilgan qiymatni o'zlashtiradi [3, 11-12 b.]. Bir paytning o'zida probel vositasida bir nechta va har xil turdagi qiymatlarni oqimdan kiritish mumkin. Qiymat kiritish "Enter" tugmasini bosish bilan tugaydi. Agar kiritilgan qiymatlar soni o'zgaruvchilar sonidan ko'p bo'lsa, «*ortiqcha*» qiymatlar bufer xotirada saqlanib qoladi.

```
int x, y;  
float z;  
cin >> x >> y >> z;
```

SHuni qayd etish kerakki, oqimga qiymat kiritishda probel ajratuvchi hisoblanadi. Haqiqiy sonning butun va kasr qismlari '.' belgisi bilan

ajratiladi.

8.2. Mantiqiy operatorlar

Mantiqiy ko'paytirish operatori ikkita ifodani hisoblaydi, agar ikkala ifoda *true* qiymat qaytarsa, *BA* operatori ham *true* qiymat qaytardi [3, 22 b.]:

Operator	Belgi	Namuna
<i>BA</i>	&&	<i>ifoda1 && ifoda2</i>
<i>YOKI</i>	 	<i>ifoda1 ifoda2</i>
<i>INKOR</i>	!	<i>!ifoda</i>

Mantiqiy ko'paytirish operatori **&&** belgi orqali belgilanadi.

ifoda1	ifoda1	ifoda1 && ifoda2
<i>false</i> (0)	<i>false</i> (0)	<i>false</i> (0)
<i>false</i> (0)	<i>true</i> (0 emas)	<i>false</i> (0)
<i>true</i> (0 emas)	<i>false</i> (0)	<i>false</i> (0)
<i>true</i> (0 emas)	<i>true</i> (0 emas)	<i>true</i> (1)

Mantiqiy qo'shish operatori ham ikkita ifoda orqali hisoblanadi. Agarda ulardan birortasi *true* bo'lsa, mantiqiy qo'shish operatori *true* qiymat qaytaradi. Mantiqiy qo'shish operatori **||** belgi orqali belgilanadi.

ifoda1	ifoda1	ifoda1 ifoda2
<i>false</i> (0)	<i>false</i> (0)	<i>false</i> (0)
<i>false</i> (0)	<i>true</i> (0 emas)	<i>true</i> (1)
<i>true</i> (0 emas)	<i>false</i> (0)	<i>true</i> (1)
<i>true</i> (0 emas)	<i>true</i> (0 emas)	<i>true</i> (1)

Mantiqiy inkor operatori tekshirilayotgan ifoda *yo'l g'on* bo'lsa, *true* qiymat qaytaradi. Agarda tekshirilayotgan ifoda *rost* bo'lsa, inkor operatori *false* qiymat qaytaradi.

Mantiqiy inkor operatori **!** belgi orqali belgilanadi.

ifoda1	!ifoda1
<i>false</i> (0)	<i>true</i> (1)
<i>true</i> (0 emas)	<i>false</i> (0)

Ma'lumotlarning xaqiqiy son turi. Ma'lumotlarning xaqiqiy son turi olti qismdan iborat bo'lishi mumkin: butun qism, nuqta, kasr qism, e yoki E belgisi, o'nlik daraja, F yoki f suffikslari.

Masalan : 66. .0 .12 3.14F 1.12e-12

Ma'lumolarning uzun xaqiqiy son turi :

Oxiriga L yoki l suffikslari qo'yilgan xaqiqiy son.

Masalan: 2E+6L;

Mantiqiy konstanta. Mantiqiy konstantalar true(rost) va false(yolg'on) qiymatlardan iborat. Ichki ko'rinishi false – 0, ixtiyoriy boshqa qiymat true deb qaraladi.

Satrlı konstanta. Satrlı konstantalar C++ tili konstantalariga kirmaydi, balki leksemalari aloxida tipi hisoblanadi. SHuning uchun adabiyotda satrlı konstantalar satrlı leksemalar deb ham ataladi.

Satrlı konstanta bu ikkilik qavslarga olingan ixtiyoriy simvollar ketma ketligidir. Misol uchun "Men satrlı konstantaman".

Satrlar orasiga eskeyp simvollar ham kirishi mumkin. Bu simvollar oldiga \ belgisi qo'yiladi. Misol uchun:

"\n Bu satr\n uch qatorga\n joylashadi".

Satr simvollari xotirada ketma ket joylashtiriladi va har bir satrlı konstanta oxiriga avtomatik ravishda kompilyator tomonidan '\0' simvoli qo'shiladi. SHunday satrning xotiradagi xajmi simvollar soni+1 baytga tengdir.

Ketma-ket kelgan va bo'shliq, tabulyatsiya yoki satr oxiri belgisi bilan ajratilgan satrlar kompilyatsiya davrida bitta satrga aylantiriladi. Misol uchun:

"Salom" "Toshkent "

satrlari bitta satr deb qaraladi.

"Salom Toshkent"

Bu qoidaga bir necha qatorga yozilgan satrlar ham bo'ysinadi. Misol uchun :

"O'zbekistonga "

"baxor "

"keldi"

qatorlari bitta qatorga mos:

"O'zbekistonga baxor keldi"

Agar satrda '\ ' belgisi uchrasa va bu belgidan so'ng to '\n' satr oxiri belgisigacha bo'shlik belgisi kelsa bu bo'shlik belgilari '\ ' va '\n' belgisi bilan birga satrdan o'chiriladi. Satrning o'zi keyingi satrda kelgan satr bilan qo'shiladi.

"O'zbekistonga \ "

" baxor\

" keldi"

qatorlari bitta qatorga mos:

"O'zbekistonga baxor keldi"

Nomlangan o'zgarmaslar. Belgili o'zgarmas – bu nomga ega bo'lgan o'zgarmasdir. C++ tilida belgili o'zgarmasni aniqlashning ikki usuli mavjud:

#define direktivasi yordamida o'zgarmasni aniqlash.

const kalitli so'zi orqali o'zgarmasni aniqlash.

An'anaviy usul hisoblangan `#define` direktivasi orqali o'zgarmaning aniqlashni quyidagi misolda ko'rishimiz mumkin.

```
#define StudentsPerClass 15
```

Bu holda `StudentsPerClass` o'zgarmaning hech qanday tipga tegishli bo'lmaydi. Preprocessor `StudentsPerClass` so'ziga duch kelganida uni 15 literaliga almashtiradi.

C++ tilida `#define` direktivasidan tashqari o'zgarmaning aniqlashning nisbatan qulayroq bo'lgan yangi usuli ham mavjud:

```
const unsigned short int StudentsPerClass=15
```

Bu misolda ham belgili konstanta `StudentsPerClass` nomi bilan aniqlanayapti va unga `unsigned short int` tipi berilyapti. Bu usul bir qancha imkoniyatlarga ega bo'lib u sizning dasturingizni keyingi himoyasini yengillashtiradi. Bu o'zgarmaning oldingisidan eng muhim afzalligi uning tipga egaligidir.

Bu konstantalar qiymatlarini dasturda o'zgartirish mumkin emas. Konstantalar nomlari dasturchi tomonidan kiritilgan va xizmatchi so'zlardan farqli bo'lgan identifikatorlar bo'lishi mumkin. Odatda nom sifatida katta lotin harflari va ostiga chizish belgilari kombinatsiyasidan iborat identifikatorlar ishlatiladi. Nomlangan konstantalar quyidagi shaklda kiritiladi:

```
const tip konstanta_nomi=konstanta_qiymati.
```

Misol uchun:

```
const double EULER=2.718282;
```

```
const long M=999999999;
```

```
const R=765;
```

Sanovchi konstantalar `enum` xizmatchi so'zi yordamida kiritilib, `int` tipidagi sonlarga qulay so'zlarni mos qo'yish uchun ishlatiladi.

Misol uchun:

```
enum{one=1,two=2,three=3};
```

Agar son qiymatlari ko'rsatilmagan bo'lsa eng chapki so'zga 0 qiymati berilib kolganlariga tartib bo'yicha o'suvchi sonlar mos qo'yiladi:

```
enum{zero,one,two};
```

Bu misolda avtomatik ravishda konstantalar quyidagi qiymatlarni qabul qiladi:

```
zero=0, one=1, two=2;
```

Konstantalar aralash ko'rinishda kiritilishi ham mumkin:

```
enum(zero,one,for=4,five,seeks}.
```

Bu misolda avtomatik ravishda konstantalar quyidagi qiymatlarni qabul qiladi:

```
zero=0, one=1, for=4,five=5,seeks=6;
```

Sanoqli tiplarni hosil qilish uchun `enum` kalitli so'zi va undan keyin tip nomi hamda figurali qavs ichida vergullar bilan ajratilgan o'zgarmaning qiymatlari ro'yxati ishlatiladi. Masalan, `RANG` nomli sanoqli tip deb e'lon qilaylik va uning uchun 5 ta `QIZIL`, `KO'K`, `YASHIL`, `OQ`, `QORA` qiymatlarini aniqlaylik.

enum RANG { QIZIL, KO'K, YASHIL, OQ, QORA };

Bunda ifoda ikkita ishni bajaradi:

RANG nomli yangi sanoqli tip hosil qiladi;

Quyidagi belgili o'zgarmlarni aniqlaydi.

0 qiymat bilan QIZIL;

1 qiymat bilan KO'K;

2 qiymat bilan YASHIL va hokazo;

Matematik amallar. C++ tilida 5 ta asosiy matematik amallar qo'llaniladi: qo'shish (+), ayirish (-), ko'paytirish (*), butun songa bo'lish (/) va modul bo'yicha bo'lish (%) (qoldiqni olish). Amallar odatda unar ya'ni bitta operandga qo'llaniladigan amallarga va binar ya'ni ikki operandga qo'llaniladigan amallarga ajratiladi.

Binar amallar additiv ya'ni + qo'shish va – ayirish amallariga, hamda multiplikativ ya'ni * ko'paytirish, / bo'lish va % modul olish amallariga ajratiladi.

Butun songa bo'lish odatdagi bo'lishdan farq qiladi. Butun songa bo'lishdan hosil bo'lgan bo'linmaning faqatgina butun qismi olinadi.

Butun sonni butun songa bo'lganda natija butun songacha yaxlitlanadi. Misol uchun $20/3=6$; $(-20)/3=-6$; $20/(-3)=-6$.

Modul amali butun sonni butun songa bo'lishdan xosil bo'ladigan qoldiqqa tengdir. Agar modul amali musbat operandlarga qo'llanilsa, natija ham musbat bo'ladi, aks holda natija ishorasi kompilyatorga bog'liqdir.

Masalan, 21 sonini 4 ga bo'lsak 5 soni va 1 qoldiq hosil bo'ladi. 5 butun songa bo'lishni qiymati, 1 esa qoldiqni olish qiymati hisoblanadi.

Inkrement va dekrement. Dasturlarda o'zgaruvchiga 1 ni qo'shish va ayirish amallari juda ko'p hollarda uchraydi. C++ tilida qiymatni 1 ga oshirish inkrement, 1 ga kamaytirish esa dekrement deyiladi. Bu amallar uchun maxsus operatorlar mavjuddir.

Inkrement operatori (++) o'zgaruvchi qiymatini 1 ga oshiradi, dekrement operatori (—) esa o'zgaruvchi qiymatini 1 ga kamaytiradi. Masalan, s o'zgaruvchisiga 1 qiymatni qo'shmoqchi bo'lsak quyidagi ifodani yozishimiz lozim.

C++ //s o'zgaruvchi qiymatini 1 ga oshirdik.

Bu ifodani quyidagicha yozishimiz mumkin edi.

s=s+1;

Bu ifoda o'z navbatida quyidagi ifodaga teng kuchli:

s+=1;

Prefiks va postfiks. Inkrement operatori ham, dekrement operatori ham ikki

variantda ishlaydi: prefiksli va postfiksli. Prefiksli variantda ular o'zgaruvchidan oldin (++Age), postfiksli variantda esa o'zgaruvchidan keyin (Age++) yoziladi.

Oddiy ifodalarda bu variantlarni qo'llanishida farq katta emas, lekin bir o'zgaruvchiga boshqa o'zgaruvchining qiymatini o'zlashtirishda ularning qo'llanilishi boshqacha harakterga ega. Prefiksli operator qiymat o'zlashtirilguncha, postfiksli operator esa qiymat o'zlashtirilgandan keyin bajariladi. Misol uchun i qiymati 2 ga teng bo'lsin, u holda 3+(++i) ifoda qiymati 6 ga, 3+i++ ifoda qiymati 5 ga teng bo'ladi. Ikkala holda ham i qiymati 3 ga teng bo'ladi.

Amallar ustivorligi. Murakkab ifodalarda qaysi amal birinchi navbatda bajariladi, qo'shishmi yoki ko'paytirishmi? Masalan:

$x=5+3*8;$

ifodada agarda birinchi qo'shish bajarilsa natija 64 ga, agarda ko'paytirish birinchi bajarilsa natija 29 ga teng bo'ladi.

Murakkab qiymat berish amali. C++ tilida murakkab qiymat berish amali mavjud bo'lib, umumiy ko'rinishi quyidagichadir:

O'zgaruvchi_nomi amal= ifoda;

Bu yerda amal quyidagi amallardan biri $*$, $/$, $\%$, $+$, $-$, $\&$, $^$, $|$, $<<$, $>>$.

Misol uchun:

$x+=4$ ifoda $x=x+4$ ifodaga ekvivalentdir;

$x*=a$ ifoda $x=x*a$ ifodaga ekvivalentdir;

$x/=a+b$ ifoda $x=x/(a+b)$ ifodaga ekvivalentdir;

$x>>=4$ ifoda $x=x>>4$ ifodaga ekvivalentdir;

Imlo belgilari amal sifatida. C++ tilida ba'zi bir imlo belgilari ham amal sifatida ishlatilishi mumkin. Bu belgilardan oddiy () va kvadrat [] qavslardir. Oddiy qavslar binar amal deb qaralib ifodalarda yoki funktsiyaga murojaat qilishda foydalaniladi. Funktsiyaga murojaat qilish quyidagi shaklda amlga oshiriladi:

<funktsiya nomi> (<argumentlar ro'yxati>). Misol uchun $\sin(x)$ yoki $\max(a,b)$.

Murakkab ifodalarni tuzishda ichki qavslardan foydalaniladi.

Quyidagi mantiq asosida yuqori darajada tashkil qilingan signalizatsiya sistemasini tasavvur qiling. Agarda eshikda signalizatsiya o'rnatilgan bo'lsa VA kun vaqti kech soat olti VA bugun bayram YoKI dam olish kuni BO'LMASA politsiya chaqirilsin. Barcha shartlarni tekshirish uchun C++ tilining uchta mantiqiy operatori ishlatiladi. Ular jadvalda keltirilgan

Amal	Belgi	Misol
VA	&&	1ifoda && 2ifoda
YoKI		1ifoda 2ifoda
INKOR	!	!ifoda

8.3.Qo'shma operatorlar.

Bir necha operatorlar { va } figurali qavslar yordamida qo'shma operatorlarga yoki bloklarga birlashtirilishi mumkin. Blok yoki qo'shma operator sintaksis jixatdan bitta operatorga ekvivalentdir. Blokning qo'shma operatoridan farqi shundaki blokda ob'ektlar ta'riflari mavjud bo'lishi mumkin.

Quyidagi dastur qismi qo'shma operator:

```
{  
n++;  
summa+=(float)n;  
}
```

Bu fragment bo'lsa

```
blok:  
{  
int n=0;  
n++;  
summa+=(float)n;  
}
```

8.4.Shart operatorlari.

Tarmoqlanuvchi algoritmlarda tarmoqlanishni amalga oshirish uchun shartli operatoridan foydalaniladi.

if operatori. Odatda dastur satrma–satr tartib bilan bajariladi. If operatori shartni tekshirish (masalan, ikki o'zgaruvchi tengmi) va uning natijasiga bog'liq ravishda dasturni bajarilish tartibini o'zgartirish imkonini beradi. If operatorining oddiy shakli quyidagi ko'rinishdadir:

```
if (shart)  
ifoda;
```

Qavs ichidagi shart ixtiyoriy ifoda bo'lishi mumkin. Agarda bu ifoda false qiymatini qaytarsa undan keyingi ifoda yoki blok tushirib qoldiriladi. Agarda shart true qiymat qaytarsa navbatdagi ifoda bajariladi. Quyidagi misolni qaraymiz:

```
If (kattaSon>kichikSon)  
kattaSon=kichikSon;
```

Bu yerda kattaSon va kichikSon o'zgaruvchilari taqqoslanayapti. Agarda kattaSon o'zgaruvchisi qiymati katta bo'lsa, bu navbatdagi qatorda unga qiymat sifatida kichikSon o'zgaruvchisining qiymati o'zlashtiriladi.

if operatorida figurali qavs ichiga olingan ifodalar blokini ham ishlatish mumkin.

```
if (shart)  
{  
- ifoda  
- ifoda  
- ifoda  
}
```

Quyida ifodalar blokining qo'llanilishiga oid misol keltirilgan

```
if(kattaSon>kichikSon)
{ kattaSon=kichikSon
  cout<<"kattaSon:"<<kattaSon << "\n";
  cout<<"kichikSon:"<<kichikSon<< "\n";
}
```

Bu holda kattaSon o'zgaruvchisiga nafaqat kichikSon o'zgaruvchisi o'zlashtirilayapti, balki ekranga bu haqida axborot ham chiqarilayapti.

8.5. Takrorlash operatorlari

Takrorlash operatori "*takrorlash sharti*" deb nomlanuvchi ifodaning rost qiymatida dasturning ma'lum bir qismidagi operatorlarni (takrorlash tanasini) ko'p marta takror ravishda bajaradi (iterativ jarayon). Takrorlash shartini tekshirish takrorlash tanasidagi operatorlarni bajarishdan oldin tekshirilishi mumkin (**for**, **while** takrorlashlari) yoki takrorlash tanasidagi operatorlari bir marta bajarilgandan keyin tekshirilishi mumkin (**do-while**).

for takrorlash operatori. **for** takrorlash operatorining sintaksisi qo'yidagi ko'rinishga ega:

for (<ifoda1>; <ifoda2>; <ifoda3>)
 <operator yoki blok>;

Bu operator o'z ishini <ifoda1> ifodasini bajarishdan boshlaydi. Keyin takrorlash qadamlari boshlanadi. Har bir qadamda <ifoda2> bajariladi, agar natija 0 qiymatidan farqli yoki *true* bo'lsa, takrorlash tanasi - <operator yoki blok> bajariladi va oxirida <ifoda3> bajariladi. Agar <ifoda2> qiymati 0 (*false*) bo'lsa, takrorlash jarayoni to'xtaydi va boshqaruv takrorlash operatoridan keyingi operatorga o'tadi.

Takrorlash operatorlarining qavs ichidagi ifodalariga izoh berish mumkin: <ifoda1> - takrorlash sanagichi vazifasini bajaruvchi o'zgaruvchiga boshlang'ich qiymat berishga xizmat qiladi va u takrorlash jarayoni boshida faqat bir marta hisoblanadi. Ifodada o'zgaruvchi e'loni uchrashi mumkin va bu o'zgaruvchi takrorlash operatori tanasida amal qiladi va takrorlash operatoridan tashqarida «ko'rinmaydi»;

<ifoda2> - takrorlashni bajarish yoki yo'qligini aniqlab beruvchi mantiqiy ifoda, agar shart *rost* bo'lsa, takrorlash davom etadi, aks holda yo'q. Agar bu ifoda bo'sh bo'lsa, shart doimo *rost* deb hisoblanadi; <ifoda3> - odatda takrorlash sanagichining qiymatini oshirish (kamaytirish) uchun xizmat qiladi yoki unda takrorlash shartiga ta'sir qiluvchi boshqa amallar bo'lishi mumkin.

C++ tilining qurilmalari operatorlarni blok ko'rinishida tashkil qilishga imkon beradi. *Blok* – '{' va '}' belgilari oralig'iga olingan operatorlar ketmaketligi bo'lib, u kompilyator tomonidan yaxlit bir operator deb qabul qilinadi. Blok ichida e'lon operatorlari ham bo'lishi mumkin va ularda e'lon qilingan

o'zgaruvchilar faqat shu blok ichida ko'rinadi (amal qiladi), blokdan tashqarida ko'rinmaydi. Blokdan keyin ';' belgisi qo'yilmasligi mumkin, lekin blok ichidagi har bir ifoda ';' belgisi bilan yakunlanishi shart.

Mavzu bo'yicha savollar to'plami:

1. *int* turidagi son, odatda, nima uchun ishlatiladi?
2. Agar dasturda sintaktik xatolar bo'lsa, kompilyator bu haqida xabar beradimi?
3. Arifmetik operatorlar bajarilish ketma-ketligi qoidasi qanday qoida?
4. Figurali qavslar nima uchun ishlatiladi?
5. Vergul (","), odatda, nima uchun ishlatiladi?
6. "*double*" kalit so'zi nima uchun ishlatiladi?
7. Haqiqiy son turi nima?
8. Berilganlarning strukturalashgan turlari qanday?
9. Butun son turlari nima?
10. Suzuvchi nuqtali turlar nima?
11. Sanab o'tiluvchi tur nima?
12. Haqiqiy sonlar qanday kalit so'z bilan e'lon qilinadi?

Mavzu bo'yicha glossariy:

Inkrement - C++ tilida o'zgaruvchi qiymatini birga oshirish.

Dekrement - C++ tilida o'zgaruvchi qiymatini birga kamaytirish.

Chiziqli algoritmlar - faqat ketma-ket bajariladigan amallardan tashkil topgan algoritmlar.

Kompilyator – yuqori darajali dasturlash tilida yozilgan programma kodini mashina tiliga o'giradigan dastur.

Kompilyatsiya jarayoni – dastur kodini mashina tiliga o'girish jarayoni.

O'zgaruvchi – xotiraning nomlangan qismi bo'lib, o'zida ma'lum bir toifadagi qiymatlarni saqlaydi.

O'zgarmaslar – hisoblash jarayonida qiymatini o'zgartirmaydigan kattalik.

Identifikator – dasturchi tomonidan dastur elementlari (funktsiya, o'zgaruvchilar, o'zgarmaslar,...) uchun ixtiyoriy tanlangan nom.

9-ma'ruza.Standart funksiyalar.

Reja:

- 1.Funksiyalarni joriy qilish, parametrni o'tkazish, qaytish qiymatlari, qaytish qiymatlarisiz funksiyalar.
- 2.Funksiyada ishlatiladigan operatorlar.
- 3.Funksiyaga parametrlarni qiymat sifatida uzatish.
- 4.Funksiyaning qaytaradigan qiymatlari.
- 5.Bir xil nomli har xil funksiyalar.

Funksiya bu – C++ tilida masala yechishdagi kalit elementlardan biridir. Funksiyalar modullar deb ham ataladi. Funksiyalar oldindan aniqlangan va foydalanuvchi tomonidan aniqlanadigan funksiyalarga bo'linadi. Oldindan aniqlangan funksiyalar, asosan, tilning turli kutubxona fayllari

orqali aniqlanadi. Ularga matematik funksiyalar, turlarni tekshirish funksiyalari, belgi va satrlar bilan ishlash funksiyalari misol bo'ladi. Masalan:

9.1.-jadval

Funksiya ishlatilishi	Kutubxona fayli	Bajaradigan amali
abs(x)	<cmath>	x butun sonining absolyut qiymatini qaytaradi
fabs(x)	<cmath>	x haqiqiy sonining absolyut qiymatini qaytaradi
log(x)	<cmath>	x sonining natural logarifmini qaytaradi
pow(x, y)	<cmath>	x^y hisoblaydi
sqrt(x)	<cmath>	x sonining kvadrat ildizini qaytaradi
islower(x)	<cctype>	x qiymatining kichik harfligini tekshiradi
isupper(x)	<cctype>	x qiymatini katta harfligini tekshiradi
tolower(x)	<cctype>	x qiymatini kichik harf ko'rinishiga aylantiradi
toupper(x)	<cctype>	x qiymatini katta harf ko'rinishiga aylantiradi

Dasturda ishlatiladigan har qanday foydalanuvchi tomonidan aniqlanadigan funksiyalar e'lon qilinishi kerak. Funksiyalar qiymat qaytaruvchi va qiymat qaytarmaydigan funksiyalar ko'rinishida bo'ladi. Odatda funksiyalar e'loni sarlavha fayllarda e'lon qilinadi va **#include** direktivasi yordamida dastur matniga qo'shiladi. Funksiya e'lonini *funksiya prototipi* tavsiflaydi (ayrim hollarda *signatura* deyiladi). Funksiya prototipi quyidagi ko'rinishda bo'ladi: **<qaytaruvchi qiymat turi> <funksiya nomi>(<parametrlar ro'yxati >);** Bu yerda **<qaytaruvchi qiymat turi>** - funksiya ishlashi natijasida u tomonidan qaytaradigan qiymatning turi. Agar qaytariladigan qiymat turi ko'rsatilmagan bo'lsa, kelishuv bo'yicha funksiya qaytaradigan qiymat turi **int** deb hisoblanadi, **<parametrlar ro'yxati>** - vergul bilan ajratilgan funksiya parametrlarining turi va nomlari ro'yxati. Parametr nomini yozmasa ham bo'ladi. Ro'yxat bo'sh bo'lishi ham mumkin. Funksiya prototiplariga misollar:

```
int almashsin(int, int);
double max(double x, double y);
void func();
```

Funksiya prototipi tushirib qoldirilishi mumkin, agar dastur matnida funksiya aniqlanishi uni chaqiradigan funksiyalar matnidin oldin yozilgan bo'lsa. Lekin bunday holat yaxshi uslub hisoblanmaydi, ayniqsa, o'zaro bir-biriga murojaat qiluvchi funksiyalarni e'lon qilishda muammolar yuzaga kelishi mumkin. *Funksiya aniqlanishi* - funksiya sarlavhasida va figurali qavsga ('{' , '}') olingan qandaydir amaliy mazmunga ega tanadan iborat bo'ladi. Agar funksiya

qaytaruvchi turi **void** turidan farqli bo'lsa, uning tanasida, albatta, mos turdagi parametrga ega **return** operatori bo'lishi shart. Funksiya tanasida bittadan ortiq **return** operatori bo'lishi mumkin. Ularning ixtiyoriy birortasini bajarish orqali funksiyadan chiqib ketiladi. Agar funksiyaning qiymati dasturda ishlatilmaydigan bo'lsa, funksiyadan chiqish uchun parametrsiz **return** operatori ishlatilishi mumkin yoki umuman, **return** ishlatilmaydi. Oxirgi holda funksiyadan chiqish - oxirgi yopiluvchi qavsga yetib kelganda ro'y beradi.

Funksiya dasturning birorta modulida yagona ravishda aniqlanishi kerak, uning e'loni esa funksiyani ishlatadigan modullarda bir necha marta yozilishi mumkin. Funksiya aniqlanishida sarlavhadagi barcha parametrlar nomlari yozilishi shart.

Funksiyani ichida aniqlangan o'zgaruvchilar lokal ko'rinish sohasiga ega deyiladi. Yuqorida aytib o'tilganidek, bu o'zgaruvchilarni faqatgina funksiyaning ichidagina qo'llash mumkinligini anglatadi. C++da o'zgaruvchilarni nafaqat dasturning boshida balki ixtiyoriy joyda aniqlash mumkin. Agarda o'zgaruvchi funksiya tanasidagi biror bir blok ichida aniqlangan bo'lsa, bu o'zgaruvchi faqatgina shu blok ichidagina ta'sirga ega bo'lib butun funksiyaning ichida ko'rinish sohasiga ega bo'lmaydi.

Lokal o'zgaruvchini ko'rinish sohasi.

```
1:  #include <iostream.h>
2:  void MeningFunktsiyam();
3:  int main()
4:  {
5:      int x=5;
6:      cout<<"\n\n main()dagi x ning qiymati:"
7:      <<x;
8:
9:      Mening Funktsiyam();
10:
11: cout<<"\n main()dagi x ning qiymati:"<< x;
12: return 0;
13: }
14: void Mening Funktsiyam();
15: {
16: int x = 8;
17: cout <<"\n\n Mening Funktsiyam()dagi"
18: <<"local x ning qiymati: "<< x << endl;
19: {
20: cout << "\ n\ MeningFunktsiyam() "
21: <<"funksiyasi blokidagi x ning qiymati"
22: << " x:"<<x;
23: int x = 9;
24: cout<<"\n Blok ichida aniqlangan"
25: << x ning qiymati:"<<x;
26: }
```

```

27: cout<<"\n MeningFunktsiyam()dagi"
28: << "blockdan tashqarisida x ning qiymati:"
29: <<x<<endl;
30: }

```

HATIJA:

main()dagi x ning qiymati: 5
Mening Funktsiyam()dagi lokal x ning qiymati: 8
Mening Funktsiyam() funktsiyasi blokidagi x ning qiymati: 8
Blok ichida aniqlangan x ning qiymati: 9
Mening Funktsiyam()dagi blockdan tashqarisida x ning qiymati: 8
main()dagi x ning qiymati: 5

9.2.Funksiyada ishlatiladigan operatorlar.

Funksiyada ishlatiladigan operatorlar soni va turiga hech qanday chegara yo'q. Funksiya ichida istalgancha boshqa funktsiyalarni chaqirish mumkin bo'lsada, lekin funktsiyalarni aniqlash mumkin emas.

C++ tilida funktsiyaning hajmiga hech qanday talab qo'yilmasa ham uni kichikroq tarzda tuzgan ma'quldir. Har bir funksiya tushunish oson bo'ladigan bitta masalani bajarishi lozim. Agarda funksiya hajmi kattalashayotganligini sezsangiz, bu funktsiyani bir nechta kichikroq funktsiyalarga ajratish haqida o'ylashingiz kerak.

Funksiya argumentlari.

Funksiyaning argumentlari turli tipda bo'lishi mumkin. SHuningdek, argument sifatida C++ tilidagi biror bir qiymat qaytaradigan o'zgarmaslar, matematik va mantiqiy ifodalar va boshqa ixtiyoriy funktsiyalarni berish mumkin.

Misol sifatida bizda biror bir qiymat qaytaruvchi double(), triple(), square() va cube() funktsiyalari berilgan bo'lsin. Biz quyidagi ifodani yozishimiz mumkin:

```
Answer=(double(triple(square(my Value))))
```

Bu ifodada myValue o'zgaruvchisini qabul qilinadi va u cube() funktsiyasiga argument sifatida uzatiladi.

cube() funktsiyasi qaytargan qiymat esa square() funktsiyasiga argument sifatida uzatiladi. square() funktsiyasi qiymat qaytargandan keyin, buning qiymati o'z navbatida triple() funktsiyasiga argument sifatida beriladi. triple() funktsiyasining qiymati esa double() funktsiyasiga argument qilib beriladi va u Answer o'zgaruvchisiga o'zlashtiriladi.

Parametrlar bu lokal o'zgaruvchilardir.

Funksiyaga uzatilgan har bir argument, bu funktsiyaga nisbatan lokal munosabatda bo'ladi. Funktsiyani bajarilishi jarayonida argumentlar ustida bajarilgan o'zgartirishlar funktsiyaga qiymat sifatida berilgan o'zgaruvchilarga ta'sir qilmaydi.

9.3.Funksiyaga parametrlarni qiymat sifatida uzatish.

```
1: // Funksiyaga parametrlarni qiymat sifatida
```

```

2: // uzatish
3: #include <iostream.h>
4: void Almashtirish(int x, int y);
5: int main()
6: { int x = 5, y = 10;
7:   cout << "Main(). Almashtirishdan oldin,x:"
8:   <<x <<" y:" <<y <<"\n";
9:   Almashtirish(x,y);
10:  cout<<"Main(). Almashtirishdan keyin, x:"
11:  <<x<< "y:" <<y<< "\n";
12:  return 0;
13: }
14: void Almashtirish(int x, int y)
15: {
16:  int temp;
17:  cout<<"Almashtirish().Almashtirishdan "
18:  "oldin, x: " <<x<<" y: " <<y<< "\n";
19:  temp = x ;
20:  x = y;
21:  y = temp;
22:  cout<<"Almashtirish().Almashtirishdan "
23:  "keyin, x: " <<x<<" y: " <<y<< "\n";
24: }

```

HATIJA:

```

Main(). Almashtirishdan oldin,x: 5 y:10
Almashtirish().Almashtirishdan oldin, x:5 y:10
Almashtirish().Almashtirishdan keyin, x:10 y:5
Main(). Almashtirishdan keyin, x:5 y:10 x:

```

9.4.Funksiyaning qaytaradigan qiymatlari.

Funksiya yo biror bir real qiymatni, yo kompilyatorga hech qanday qiymat qaytarilmasligi haqida xabar beruvchi void tipidagi qiymatni qaytaradi.

Funksiyaning qiymat qaytarishi uchun return kalitli soʻzidan foydalaniladi. Bunda oldin return kalitli soʻzi, keyin esa qaytariladigan qiymat yoziladi. Qiymat sifatida esa oʻzgarmaslar kabi butun bir ifodalarni ham berish mumkin. Masalan:

```

return 5 ;
return (x > 5) ;
return (MyFunction()) ;

```

MyFunction() funktsiyasi biror bir qiymat qaytarishidan kelib chiqsak, yuqoridagi barcha ifodalar toʻgʻri keltirilgan. return(x>5) ifodasi esa x 5 dan katta boʻlsa true, kichik yoki teng boʻlsa false mantiqiy qiymatini qaytaradi.

Agarda funktsiyada return kalit soʻzi uchrasa undan keyingi ifoda bajariladi va uning natijasi funksiya chaqirilgan joyga uzatiladi. return operatori bajarilgandan keyin dastur funksiya chaqirilgan satrdan keyingi ifodaga oʻtadi.

return kalitli so'zidan keyingi funksiya tanasidagi operatorlar bajarilmaydi.

Funksiya bir nechta return operatorlarini o'zida saqlashi mumkin. Bu g'oya 5.6 – listingda namoyish qilingan.

Bir nechta return operatorini qo'llanilishi

```
1: // 5.6. – listing.
2: #include <iostream.h>
3: int IkkigaKo'paytirish(int Ko'paytSon);
4:
5: int main()
6: {
7:     int natija=0;
8:     int input;
9:     cout << "Ikkiga ko'paytiriladigan sonni"
10: << "kiriting(0 dan 10000 gacha):";
11: cin >> input;
12:
13: cout << "\n IkkigaKo'paytirish() funksiyasi"
14: << "chaqirilishidan oldin\n";
15: cout<<"Kiritilgan qiymat:" <<input
16: <<"Ikkilangani:"<<natija<< "\n";
17:
18: result = IkkigaKo'paytirish(input);
19: cout<<"\nIkkigaKo'paytirish() funksiyasidan"
20: <<"qaytgandan so'ng...\n";
21:
22: return 0;
23: }
24: int IkkigaKo'paytirish(int original)
25: {
26:     if (original <= 10000)
27:         return original*2;
28:     else
29:         return -1;
30: }
```

NATIJA:

Ikkiga ko'paytiriladigan sonni kiriting (0 dan 10000 gacha): 9000

IkkigaKo'paytirish() funksiyasi chaqirilishidan oldin

Kiritilgan qiymat: 9000 Ikkilangani: 0

Ikkiga_ko'paytirish() funktsiyasidan qaytgandan so'ng

Kiritilgan qiymat:9000 Ikkilangani: 18000

Ikkiga ko'paytiriladigan sonni kiriting (0 dan 10000 gacha): 11000

IkkigaKo'paytirish() funksiyasi chaqirilishidan oldin

Kiritilgan qiymat: 11000 Ikkilangani: 0

Ikkiga_ko'paytirish() funksiyasidan qaytgandan so'ng

Kiritilgan qiymat:11000 Ikkilangani: -1

9.5.Bir xil nomli har xil funksiyalar.

C++ tilida bir nomdagi bir nechta funksiyalarni tuzish imkoniyati mavjuddir. Bu bir xil nomdagi xar xil funksiyalar deyiladi. Qayta yuklanuvchi funksiyalar bir – birlari bilan parametrlari ro'yxati bilan farq qilishi lozim. Bunda parametrlar yoki turli sonda aniqlanishi, yoki tiplari bilan farqlanishi kerak. Quyidagi misollarni ko'rib chiqamiz:

```
int myFunction( int, int )
```

```
int myFunction( long, long)
```

```
int myFunction( long )
```

myFunction() funksiyasi uchta turli parametrlar ro'yxati bilan xar xil nomda aniqlanyapti. Funksiyaning birinchi va ikkinchi versiyalari parametrlar tipi bilan, uchinchi esa parametrlar soni bilan farq qilayapti.

Ushbu funksiyalarning qaytaradigan qiymatlarining tiplari bir xil yoki turlicha bo'lishi mumkin. Lekin, parametrlari ro'yxati bir xil bo'lgan ikkita funktsiya turli tipdagi qiymat qaytaradigan qilib aniqlansa dasturni kompilyatsiya qilish jarayonida xatolik yuz beradi.

Funksiyalarni bir xil nom bilan aniqlanishi ularning polimorfizmi deb ham ataladi. Polimorfizm so'zi poli (gr. poly) - ko'p, morfe (gr. morphyo) - shakl co'zidan olingan bo'lib, ko'pshakllilik degan ma'noni anglatadi.

Funksiyaning polimorfizmi deganda dasturda bir nechta turli vazifalarni bajaruvchi bir xil nomdagi funksiyalar bo'lishi tushuniladi. Parametrlari soni va tipini o'zgartirish orqali bir nechta bir xil nomdagi funksiyalarni aniqlash mumkin. Bunday holda funktsiyani chaqirishda hech qanday noaniqlik bo'lmaydi, kerakli funktsiya parametriga muvofiq tarzda aniqlanadi.

Faraz qilamiz, siz ixtiyoriy berilgan qiymatni ikkiga ko'paytiradigan funktsiya yozmoqchisiz. Bu funktsiyaga int, long, float yoki double tipidagi qiymatlarni uzatish imkoniyati bo'lishini hohladingiz. Bir xil nomli xar xil funksiyalarsiz buni bajarish uchun to'rtta turli nomdagi funksiyalarni hosil qilishingiz kerak bo'ladi:

```
int DoubleInt(int);
```

```
long DoubleLong(long);
```

```
float DoubleFloat(float);
```

```
double DoubleDouble(double);
```

Bir xil nomli funksiyalar yordamida esa ularni o'rniga ushbu funksiyalarni

aniqlashimiz mumkin.

```
int Double(int);  
long Double(long);  
float Double (float);  
double Double(Double);
```

Qayta yuklangan funksiyalar chaqirilganda kompilyator aynan qaysi variantdagi funksiyani ishlatish kerakligini avtomatik tarzda uzatilayotgan parametrlarning tiplariga muvofiq aniqlaydi. Bir xil nomli funksiyalarning aniqlanishi.

Mavzu bo'yicha savollar to'plami:

- 1.Nima uchun literalli o'zgarmasga nisbatan belgili o'zgarmasni ishlatish yaxshiroq?
- 2.const kalitli so'zini #define direktivasi o'rniga qo'llashni afzalligi nimada?
- 3.#include direktivasi qanday vazifani bajaradi.
- 4.main() funksiyasining o'ziga xos xususiyati nimadan iborat?
- 5.Qanday izoh turlarini bilasiz ? Ular nima bilan farq qiladi?
- 6.Izohlar bir necha qatorda yozilishi mumkinmi?

Mavzu bo'yicha glossariy:

Inkrement - C++ tilida o'zgaruvchi qiymatini birga oshirish.

Dekrement - C++ tilida o'zgaruvchi qiymatini birga kamaytirish.

Chiziqli algoritmlar - faqat ketma-ket bajariladigan amallardan tashkil topgan algoritmlar.

Kompilyator – yuqori darajali dasturlash tilida yozilgan programma kodini mashina tiliga o'giradigan dastur.

Kompilyatsiya jarayoni – dastur kodini mashina tiliga o'girish jarayoni.

O'zgaruvchi – xotiraning nomlangan qismi bo'lib, o'zida ma'lum bir toifadagi qiymatlarni saqlaydi.

O'zgarmaslar – hisoblash jarayonida qiymatini o'zgartirmaydigan kattalik.

Identifikator – dasturchi tomonidan dastur elementlari (funksiya, o'zgaruvchilar, o'zgarmaslar,...) uchun ixtiyoriy tanlangan nom.

10-ma'ruza. MathCad dasturi haqida umumiy tushuncha.

Reja:

1. MATHCAD dasturining interfeysi.
2. Formulalarni kiritish va tahrirlash.
- 3.Diskret o'zgaruvchilar va sonlarni formatlash.
- 4.Sonlarni formatlash.

Tayanch so'z va iboralar: MATHCAD, diagramma, Calculator, Graph, Calculus, Untitled, Greek– Symbolik, Programming.

MATHCAD oddiy, yetarlicha qayta ishlangan va tekshirilgan matematik hisoblashlar tizimidir. MATHCAD – bu kompyuter matematikasining zamonaviy

sonli usullarini qo'llashning unikal majmuasidir. U o'z ichiga yillar ichidagi matematikaning rivojlanishi natijasida yig'ilgan tajribalar, qoidalar va matematik hisoblash usullarini olgan.

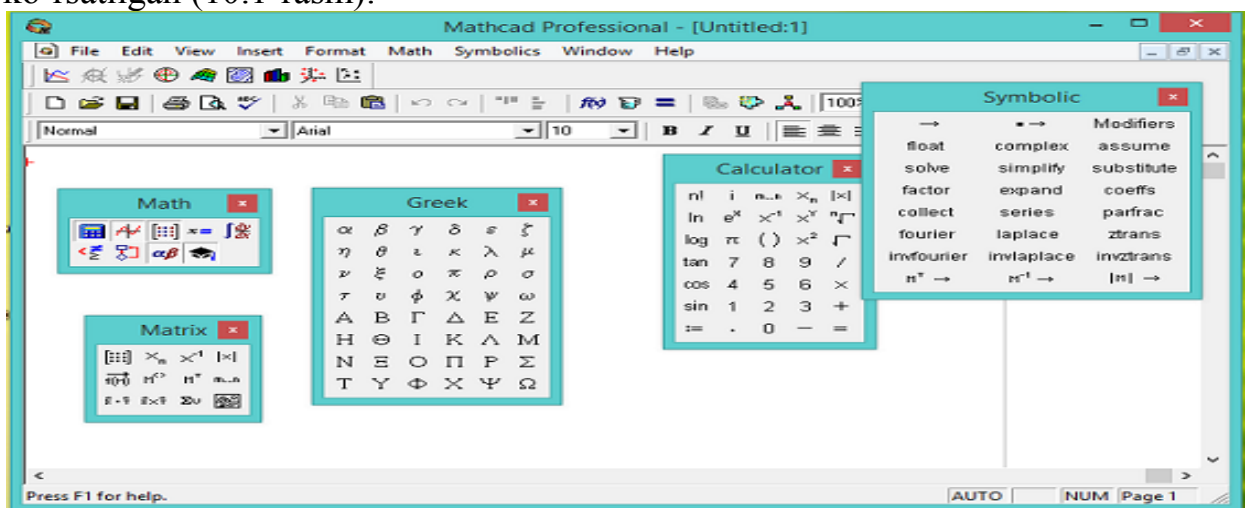
MATHCAD paketi muhandislik hisob-kitob ishlarini bajarish uchun dasturiy vosita bo'lib, u professional matematiklar uchun mo'ljallangan. Uning turli versiyalari mavjud.

Uning yordamida o'zgaruvchi va o'zgarmas parametrlar algebraik va differensial tenglamalarni yechish, funksiyalarni tahlil qilish va ularning ekstremumini izlash, topilgan yechimlarni tahlil qilish uchun jadvallar va grafiklar qurish mumkin.

MATHCAD murakkab masalalarni yechish uchun o'z dasturlash tiliga ham ega.

MATHCAD dasturining interfeysi: Windowsning barcha dasturlari interfeysiga o'xshash. MATHCAD ishga tushurilgandan so'ng uning oynasida bosh menyu va uchta panel vositasi chiqadi. MATHCAD ishga tushganda avtomatik ravishda uning ishchi hujjat fayli Untitled 1 nom bilan ochiladi va unga Worksheet (Ish varag'i) deyiladi.

Standart (Standart) vositalar paneli bir necha fayllar bilan ishlash uchun buyruqlar to'plamini o'z ichiga oladi. Formatting (Formatlash) formula va matnlarni formatlash bo'yicha bir necha buyruqlarni o'z ichiga oladi. Math (Matematika) matematik vositalarini o'z ichiga olgan bo'lib, ular yordamida simvollar va operatorlarni hujjat fayli oynasiga joylashtirish uchun qo'llaniladi. Quyidagi rasmda MATHCADning oynasi va uning matematik panel vositalari ko'rsatilgan (10.1-rasm):



10.1-rasm. MATHCAD paketi oynasi va uning matematik panel vositalari.

Calculator– asosiy matematik amallar shabloni; (Kalkulyator)

Graph– grafiklar shabloni; (Grafik)

Matrix (Massiv) – massiv va massiv amallarini bajarish shabloni; (Massiv)

Evaluation– qiymatlarni yuborish operatori va natijalarni chiqarish operatori; (Baholash)

Calculus– differensiallash, integrallash, summani hisoblash shabloni; (Hisoblash)

Boolean– mantiqiy operatorlar; (Mantiqiy operatorlar)

Programming– dastur tuzish uchun kerakli modullar yaratish o'preatorlari; (Dasturlashtirish)

Greek– Symbolik - belgililar ustida ishlash uchun operatorlar. (Grek alifbosi harflari)

Formulalarni kiritish va tahrirlash

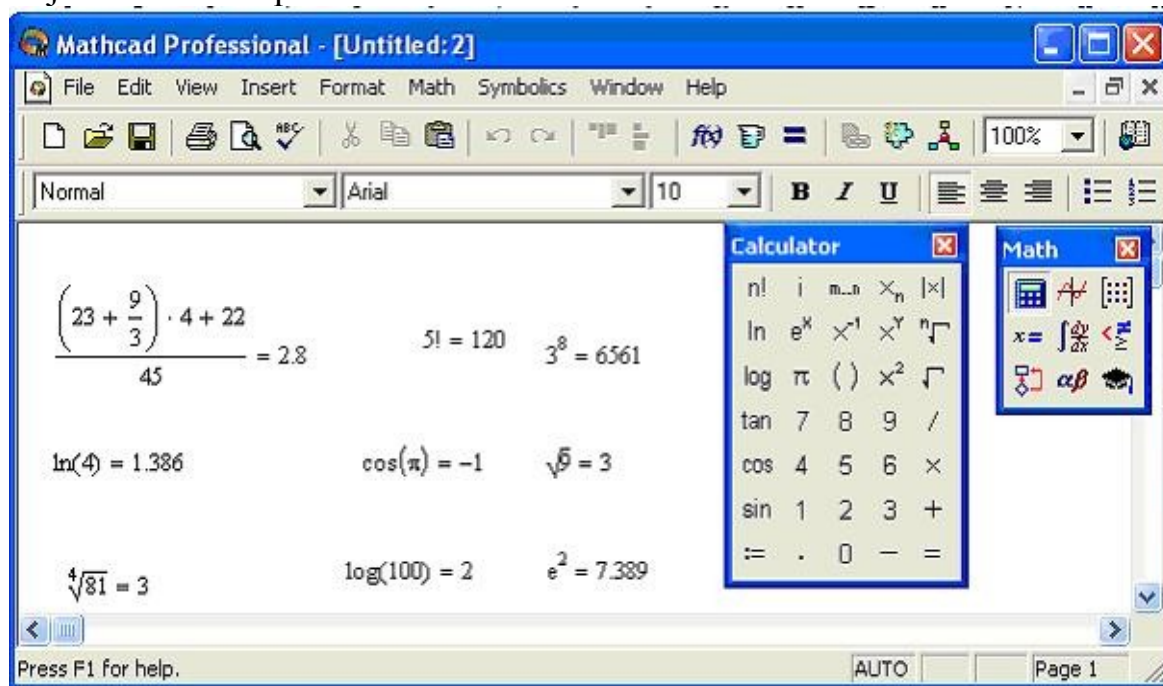
Boshlang'ich holatda ekranda kursor krestik ko'rinishda bo'ladi. Ifodani kiritishda u kiritilayotgan ifodani egallab olgan ko'k burchakli holatga o'tadi. MATHCADning har qanday operatorini kiritishni uchta usulda bajarish mumkin:

- menyu buyrug'idan foydalanib;
- klaviatura tugmalaridan foydalanib;
- matematik paneldan foydalanib.

O'zga'uvchilarga qiymat berish uchun yuborish operatori “:=” ishlatiladi. Hisoblashlarni amalga oshirish uchun oldin formuladagi o'zgaruvchi qiymatlari kiritiladi, keyin matematik ifoda yozilib tenglik “=” belgisi kiritiladi, natijada ifoda qiymati hosil bo'ladi (10.2-rasm).

Oddiy va matematik ifodalarni tahrirlashda menyu standart buyruqlaridan foydalaniladi. Tahrirlashda klaviaturadan ham foydalanish mumkin, masalan

- qirqib olish – Ctrl+X;
- nusxa olish – Ctrl+C;
- qo'yish – Ctrl+V;
- bajarishni bekor qilish – Ctrl+Z



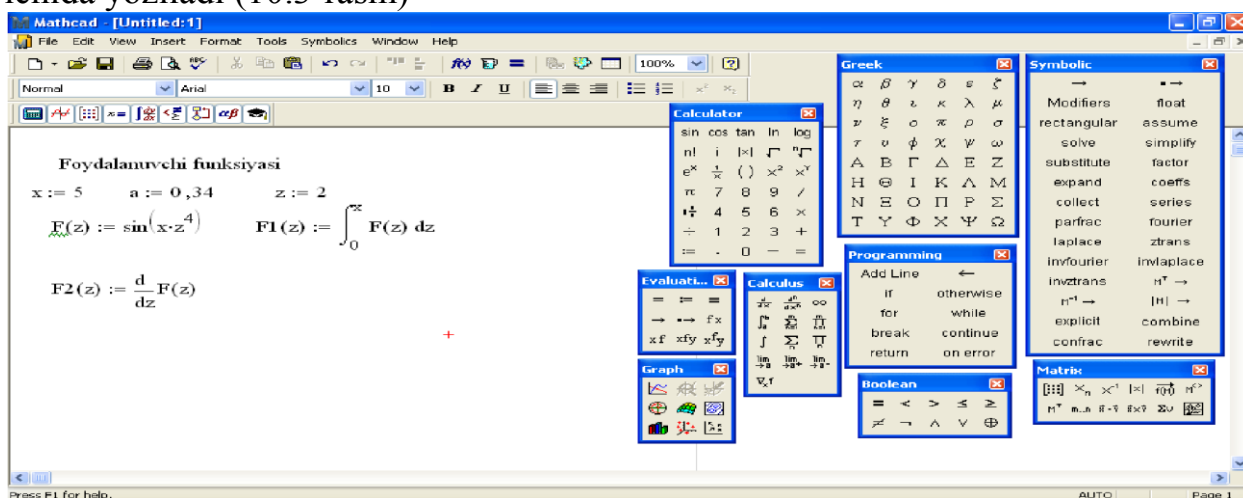
10.2-rasm. Oddiy matematik ifodalarni hisoblash.

MATHCAD dasturida 200 dan ortiq o'zida qurilgan funksiyalariga ega bo'lib, ularni matematik ifodalarda ishlatish uchun standart panel vositasidagi Insert Function (Funksiyani qo'yish) tugmasiga bog'langan muloqot oynasidan foydalaniladi.

MATHCAD 2000 hujjatiga matn kiritish uchun bosh menyudan

Insert→Text Region (Qo'yish→Matn maydoni) buyrug'ini berish yoki yaxshisi klaviaturadan ikkitali kavichka (“) belgisini kiritish kerak. Bunda matn ma'lumotini kiritish uchun ekranda matn kiritish maydoni paydo bo'ladi. Matn kiritish maydoniga matematik ifodani yozish uchun matematik maydonni ham qo'yish mumkin. Buning uchun shu matn maydonida turib Insert→Math Region (Qo'yish→Matematik maydoni) buyrug'ini berish kifoya. Bu maydondagi kiritilgan matematik ifodalar ham oddiy kiritilgan matematik maydon kabi hisoblashni bajaradi.

MATHCADda foydalanuvchi funksiyasini tuzish hisoblashlarda qulaylikni va uning effektivligini oshiradi. Funksiya chap tomonda ko'rsatilib, undan keyin yuborish operatori (:=) va hisoblanadigan ifoda yoziladi. Ifodada ishlatiladigan o'zgaruvchi kattaliklari funksiya parametri qilib funksiya nomidan keyin qavs ichida yoziladi (10.3-rasm)



10.3-rasm.

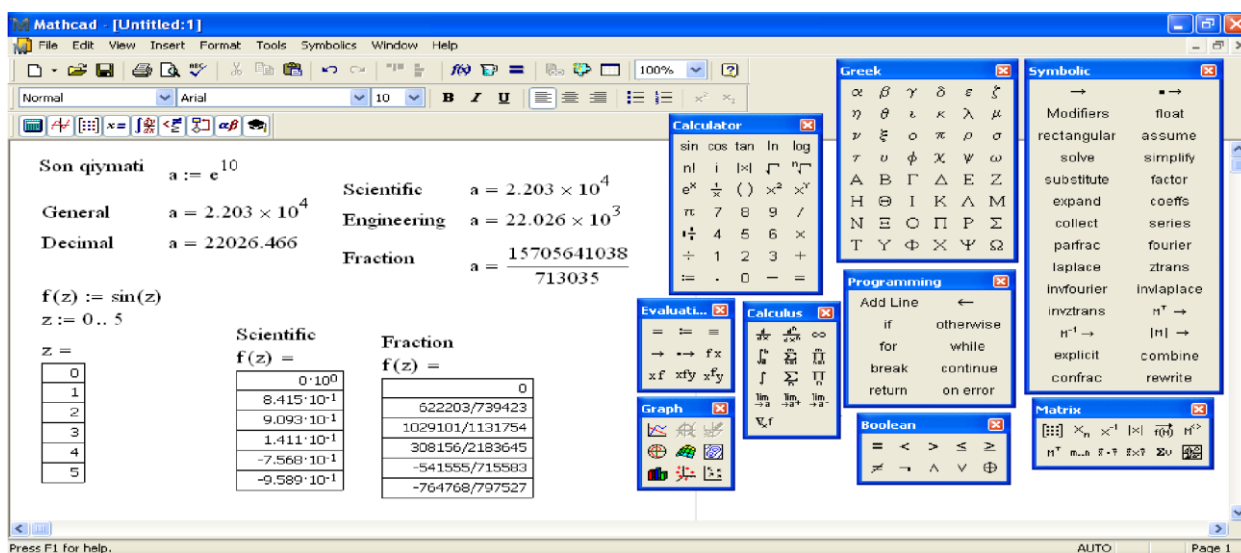
Diskret o'zgaruvchilar va sonlarni formatlash. MATHCADda diskret o'zgaruvchilar deganda sikl operatorini tushunish kerak. Bunday o'zgaruvchilar ma'lum qadam bilan o'suvchi yoki kamayuvchi sonlarni ketma-ket qabul qiladi. Masalan: $x:=0..5$. Bu shuni bildiradiki bu o'zgaruvchi qiymati qator bir necha qiymatlardir, ya'ni $x=0,1,2,3,4,5$. $x:=1,1..5$. Bunda 1 – birinchi sonni, 1,1 – ikkinchi sonni, 5 - oxirgi sonni bildiradi.

$x:=A,A+B..B$. Bunda A – birinchi, A+B – ikkinchi, B - oxirgi sonni bildiradi.

O'zgaruvchi diapazonini ko'rsatishda ikki nuqta o'rniga klaviaturadan (;) nuqta vergul kiritiladi yoki Matrix (Massiv) panelidan Range Variable (Diskret o'zgaruvchi) tugmasi bosiladi. Hisoblangan qiymatni chiqarish uchun esa o'zgaruvchi va tenglik belgisini kiritish kifoya. Natijada o'zgaruvchi qiymati ketma-ket jadvalda chiqadi. Masalan, $x:=0..5$ deb yozib, keyin $x=$ kiritish kerak. Foydalanuvchi funksiyaning uning argumentiga mos qiymatlarini hisoblab chiqarish va bu qiymatlarni jadval yoki grafik ko'rinishda tasvirlashda diskret o'zgaruvchilardan foydalanish qulaylikni keltiradi. Masalan, $f(x)=\sin(x) \cdot \cos(x)$ funksiya qiymatlarini x ning 0 dan 5 gacha bo'lgan qiymatlarida hisoblash kerak bo'lsa, u holda quyidagi kiritishni amalga oshirish kerak: $f(x)=\sin(x) \cdot \cos(x)$ $x:=0..5$ $f(x)=javob$.

Sonlarni formatlash. Odatda MATHCAD 20 belgi aniqligigacha matematik ifodalarni hisoblaydi. Hisoblash natijalarini kerakli formatga o'zgartirish uchun sichqoncha ko'rsatgichini sonli hisob chiqadigan joyga keltirib, ikki marta tez-tez bosish kerak. Natijada sonlarni formatlash natijasi Result Format oynasi paydo bo'ladi. Sonlarni formatlash quyidagilardir:

- General (Asosiy) – o'z holida qabul qilish. Son eksponensial ko'rinishda tasvirlanadi.
- Decimal (O'nlik) – o'nlik qo'zg'aluvchan nuqta ko'rinishda tasvirlanuvchi son (masalan, 12.5564).
- Scientific (Ilmiy) – son faqat darajada tasvirlanadi (masalan, $1.22 \cdot 10^5$).
- Engeneering (muhandislik) – sonning darajasi faqat 3 ga karrali qilinib tasvirlanadi (masalan, $1.22 \cdot 10_6$).
- Fraction (Kasr) – son to'g'ri yoki noto'g'ri kasr ko'rinishida tasvirlanadi.



10.4.rasm.Sonlarni formatlash va qiymatlarni har xil formada tasvirlash.

Nazorat savollari:

- 1.Qanday dasturlash tillarini bilasiz?
- 2.Mathcad dasturini yuklash ketma-ketligini aytib bering.
- 3.Mathcad dasturida nechta menyu bor?
- 4.Mathcad dasturining 6 ta xarakteristikasini sanab bering.

Mavzuga oid adabiyotlar:

- 1.R.Hamdamiyov, U.Begimqulov, N.Tayloqov. Ta'limda axborot texnologiyalari. "O'zbekiston Milliy Ensiklopediyasi" Davlat ilmiy nashriyoti. Toshkent - 2010
2. S.S. G'ulomov, B.A.Begalov. Informatika va axborot texnologiyalari. Fan, T: 2010. 686 b.
3. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
4. M.M. Aripov, T. Imomov va boshqalar Informatika. Axborot texnologiyalari. O'quv qo'llanma, 1-2-qism. -Toshkent. 2005, TDTU 334 b. va 394 b.

5.A.M.Polatov.Algoritmlar va C++ tilida dasturlash asoslari. O'quv qo'llanma. – Toshkent “Universitet” 2017. 55 b

Internet tarmog'idagi elektron manbalar:

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>
3. www.intuit.ru
4. <http://virtual-university-urasia.org>
5. <http://www.opennet.ru>
6. www.linux.org.ru
7. <https://nashol.com>

11-ma'ruza. MathCad dasturi haqida umumiy tushuncha.

Reja:

1. **MATHCAD dasturining interfeysi.**
2. **Formulalarni kiritish va tahrirlash.**
3. **Diskret o'zgaruvchilar va sonlarni formatlash.**
4. **Sonlarni formatlash.**

Tayanch so'z va iboralar: MATHCAD, diagramma, Calculator, Graph, Calculus, Untitled, Greek– Symbolik, Programming.

MATHCAD oddiy, yetarlicha qayta ishlangan va tekshirilgan matematik hisoblashlar tizimidir. MATHCAD – bu kompyuter matematikasining zamonaviy sonli usullarini qo'llashning unikal majmuasidir. U o'z ichiga yillar ichidagi matematikaning rivojlanishi natijasida yig'ilgan tajribalar, qoidalar va matematik hisoblash usullarini olgan.

MATHCAD paketi muhandislik hisob-kitob ishlarini bajarish uchun dasturiy vosita bo'lib, u professional matematiklar uchun mo'ljallangan. Uning turli versiyalari mavjud.

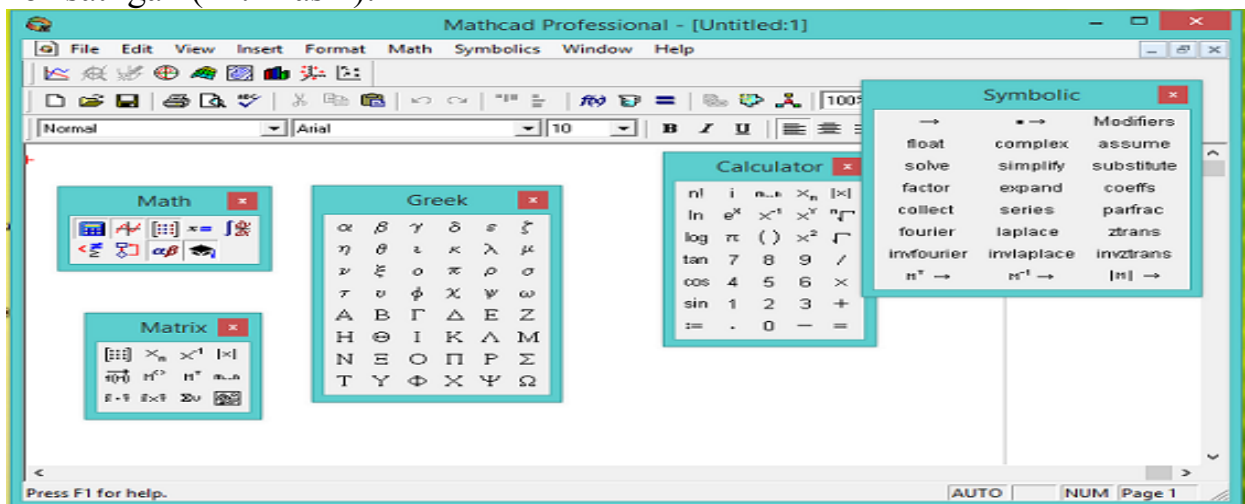
Uning yordamida o'zgaruvchi va o'zgarmas parametrli algebraik va differensial tenglamalarni yechish, funksiyalarni tahlil qilish va ularning ekstremumini izlash, topilgan yechimlarni tahlil qilish uchun jadvallar va grafiklar qurish mumkin.

MATHCAD murakkab masalalarni yechish uchun o'z dasturlash tiliga ham ega.

MATHCAD dasturining interfeysi: Windowsning barcha dasturlari interfeysiga o'xshash. MATHCAD ishga tushurilgandan so'ng uning oynasida bosh menyu va uchta panel vositasi chiqadi. MATHCAD ishga tushganda avtomatik ravishda uning ishchi hujjat fayli Untitled 1 nom bilan ochiladi va unga Workshet (Ish varag'i) deyiladi.

Standart (Standart) vositalar paneli bir necha fayllar bilan ishlash uchun buyruqlar to'plamini o'z ichiga oladi. Formatting (Formatlash) formula va matnlarni formatlash bo'yicha bir necha buyruqlarni o'z ichiga oladi. Math (Matematika) matematik vositalarini o'z ichiga olgan bo'lib, ular yordamida simvollar va operatorlarni hujjat fayli oynasiga joylashtirish uchun qo'llaniladi.

Quyidagi rasmda MATHCADning oynasi va uning matematik panel vositalari ko'rsatilgan (11.1-rasm):



11.1-rasm. MATHCAD paketi oynasi va uning matematik panel vositalari.

Calculator– asosiy matematik amallar shabloni; (Kalkulyator)

Graph– grafiklar shabloni; (Grafik)

Matrix (Massiv) – massiv va massiv amallarini bajarish shabloni; (Massiv)

Evaluation– qiymatlarni yuborish operatori va natijalarni chiqarish operatori; (Baholash)

Calculus– differensiallash, integrallash, summani hisoblash shabloni; (Hisoblash)

Boolean– mantiqiy operatorlar; (Mantiqiy operatorlar)

Programming– dastur tuzish uchun kerakli modullar yaratish operatorlari; (Dasturlashtirish)

Greek– Symbolik - belgililar ustida ishlash uchun operatorlar. (Grek alifbosi harflari)

Formulalarni kiritish va tahrirlash

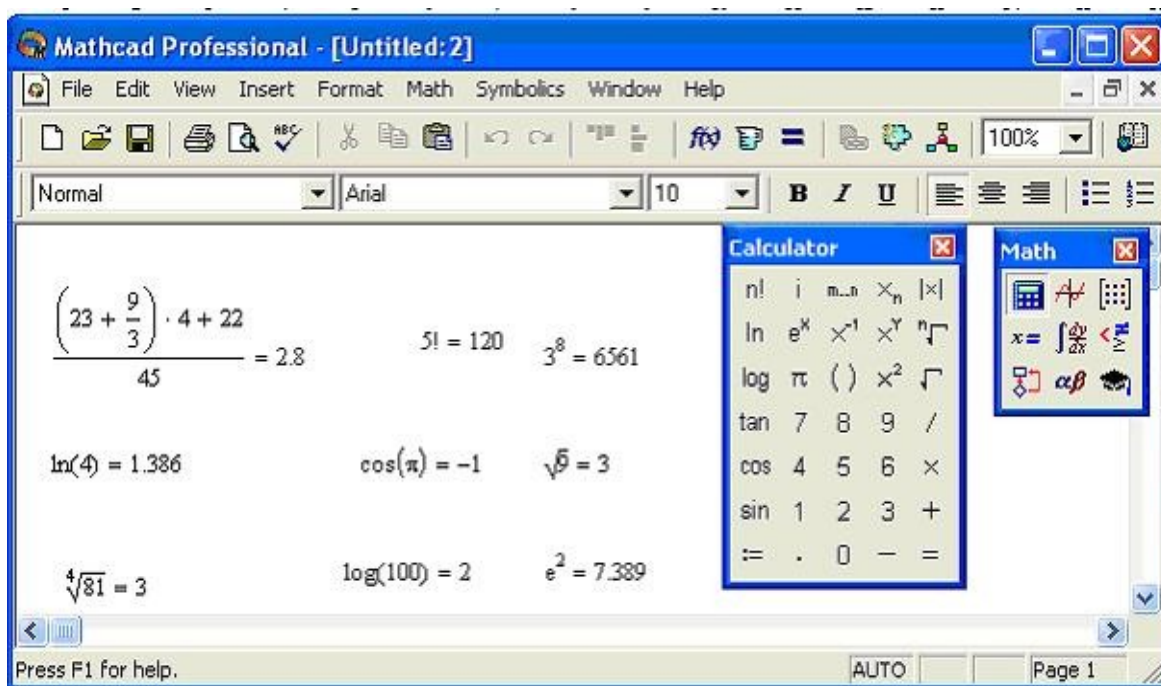
Boshlang'ich holatda ekranda kursor krestik ko'rinishda bo'ladi. Ifodani kiritishda u kiritilayotgan ifodani egallab olgan ko'k burchakli holatga o'tadi. MATHCADning har qanday operatorini kiritishni uchta usulda bajarish mumkin:

- menyu buyrug'idan foydalanib;
- klaviatura tugmalaridan foydalanib;
- matematik paneldan foydalanib.

O'zgaruvchilarga qiymat berish uchun yuborish operatori “:=” ishlatiladi. Hisoblashlarni amalga oshirish uchun oldin formuladagi o'zgaruvchi qiymatlari kiritiladi, keyin matematik ifoda yozilib tenglik “=” belgisi kiritiladi, natijada ifoda qiymati hosil bo'ladi (11.2-rasm).

Oddiy va matematik ifodalarni tahrirlashda menyu standart buyruqlaridan foydalaniladi. Tahrirlashda klaviaturadan ham foydalanish mumkin, masalan

- qirqib olish – Ctrl+X;
- nusxa olish – Ctrl+C;
- qo'yish – Ctrl+V;
- bajarishni bekor qilish – Ctrl+Z

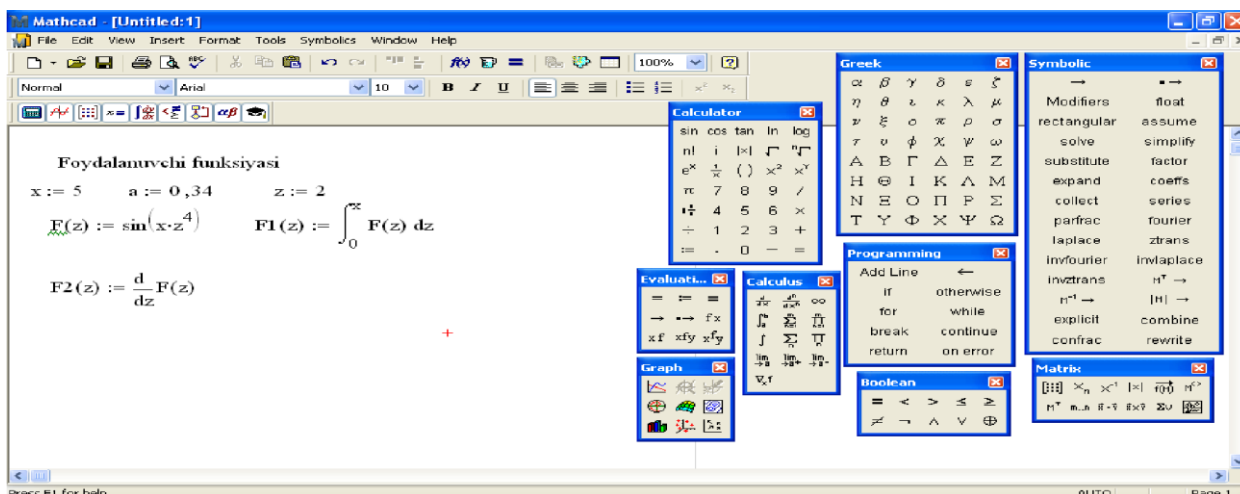


11.2-rasm. Oddiy matematik ifodalarni hisoblash.

MATHCAD dasturida 200 dan ortiq o'zida qurilgan funksiyalariga ega bo'lib, ularni matematik ifodalarda ishlatish uchun standart panel vositasidagi Insert Function (Funksiyani qo'yish) tugmasiga bog'langan muloqot oynasidan foydalaniladi.

MATHCAD 2000 hujjatiga matn kiritish uchun bosh menyudan Insert→Text Region (Qo'yish→Matn maydoni) buyrug'ini berish yoki yaxshisi klaviaturadan ikkitali kavichka ("") belgisini kiritish kerak. Bunda matn ma'lumotini kiritish uchun ekranda matn kiritish maydoni paydo bo'ladi. Matn kiritish maydoniga matematik ifodani yozish uchun matematik maydonni ham qo'yish mumkin. Buning uchun shu matn maydonida turib Insert→Math Region (Qo'yish→Matematik maydoni) buyrug'ini berish kifoya. Bu maydondagi kiritilgan matematik ifodalar ham oddiy kiritilgan matematik maydon kabi hisoblashni bajaradi.

MATHCADda foydalanuvchi funksiyasini tuzish hisoblashlarda qulaylikni va uning effektivligini oshiradi. Funksiya chap tomonda ko'rsatilib, undan keyin yuborish operatori (:=) va hisoblanadigan ifoda yoziladi. Ifodada ishlatiladigan o'zgaruvchi kattaliklari funksiya parametri qilib funksiya nomidan keyin qavs ichida yoziladi (11.3-rasm)



11.3-rasm.

Diskret o'zgaruvchilar va sonlarni formatlash. MATHCADda diskret o'zgaruvchilar deganda sikl operatorini tushunish kerak. Bunday o'zgaruvchilar ma'lum qadam bilan o'suvchi yoki kamayuvchi sonlarni ketma-ket qabul qiladi. Masalan: $x:=0..5$. Bu shuni bildiradiki bu o'zgaruvchi qiymati qator bir necha qiymatlardir, ya'ni $x=0,1,2,3,4,5$. $x:=1,1.1..5$. Bunda 1 – birinchi sonni, 1,1 – ikkinchi sonni, 5 - oxirgi sonni bildiradi.

$x:=A,A+B..B$. Bunda A – birinchi, A+B – ikkinchi, B - oxirgi sonni bildiradi.

O'zgaruvchi diapazonini ko'rsatishda ikki nuqta o'rniga klaviaturadan (;) nuqta vergul kiritiladi yoki Matrix (Massiv) panelidan Range Variable (Diskret o'zgaruvchi) tugmasi bosiladi. Hisoblangan qiymatni chiqarish uchun esa o'zgaruvchi va tenglik belgisini kiritish kifoya. Natijada o'zgaruvchi qiymati ketma-ket jadvalda chiqadi. Masalan, $x:=0..5$ deb yozib, keyin $x=$ kiritish kerak. Foydalanuvchi funksiyaning uning argumentiga mos qiymatlarini hisoblab chiqarish va bu qiymatlarni jadval yoki grafik ko'rinishda tasvirlashda diskret o'zgaruvchilardan foydalanish qulaylikni keltiradi. Masalan, $f(x)=\sin(x) \cdot \cos(x)$ funksiya qiymatlarini x ning 0 dan 5 gacha bo'lgan qiymatlarida hisoblash kerak bo'lsa, u holda quyidagi kiritishni amalga oshirish kerak: $f(x)=\sin(x) \cdot \cos(x)$ $x:=0..5$ $f(x)=javob$.

Sonlarni formatlash. Odatda MATHCAD 20 belgi aniqligigacha matematik ifodalarni hisoblaydi. Hisoblash natijalarini kerakli formatga o'zgartirish uchun sichqoncha ko'rsatgichini sonli hisob chiqadigan joyga keltirib, ikki marta tez-tez bosish kerak. Natijada sonlarni formatlash natijasi Result Format oynasi paydo bo'ladi. Sonlarni formatlash quyidagilardir:

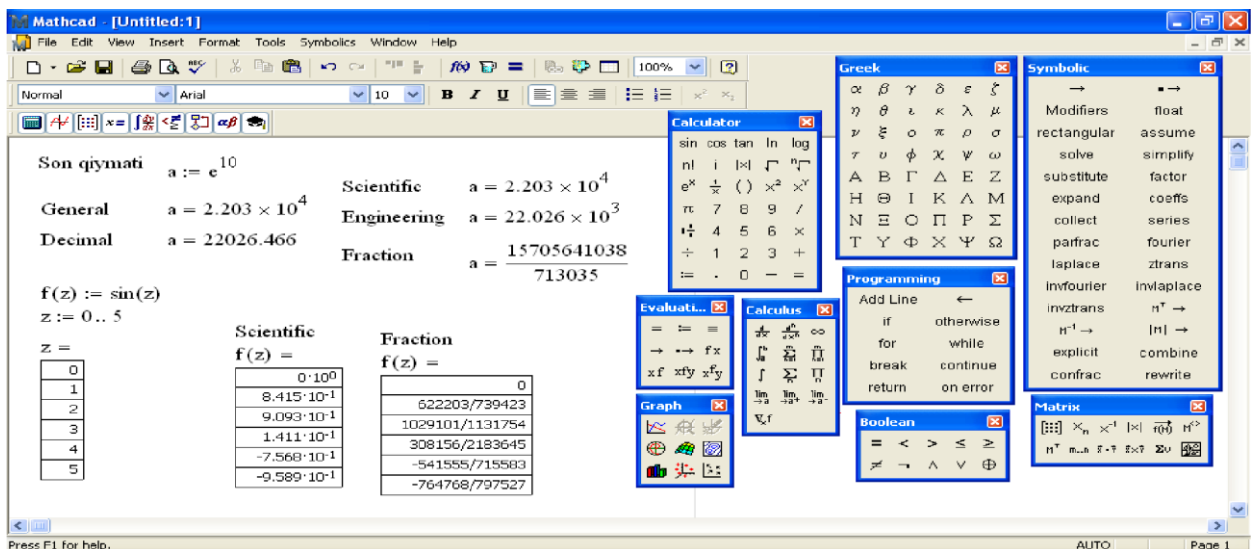
- General (Asosiy) – o'z hoida qabul qilish. Son eksponensial ko'rinishda tasvirlanadi.

- Decimal (O'nlik) – o'nlik qo'zg'aluvchan nuqta ko'rinishda tasvirlanuvchi son (masalan, 12.5564).

- Scientific (Ilmiy) – son faqat darajada tasvirlanadi (masalan, $1.22 \cdot 10^5$).

- Engeneering (muhandislik) – sonning darajasi faqat 3 ga karrali qilinib tasvirlanadi (masalan, $1.22 \cdot 10^6$).

- Fraction (Kasr) – son to'g'ri yoki noto'g'ri kasr ko'rinishida tasvirlanadi.



11.4-rasm.Sonlarni formatlash va qiymatlarni har xil formada tasvirlash.

Nazorat savollari:

- 1.Qanday dasturlash tillarini bilasiz?
- 2.Mathcad dasturini yuklash ketma-ketligini aytib bering.
- 3.Mathcad dasturida nechta menyu bor?
- 4.Mathcad dasturining 6 ta xarakteristikasini sanab bering.

Mavzuga oid adabiyotlar:

- 1.R.Hamdamov, U.Begimqulov, N.Tayloqov. Ta'limda axborot texnologiyalari. "O'zbekiston Milliy Ensiklopediyasi" Davlat ilmiy nashriyoti. Toshkent - 2010
2. S.S. G'ulomov, B.A.Begalov. Informatika va axborot texnologiyalari. Fan, T: 2010. 686 b.
3. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
4. M.M. Aripov, T. Imomov va boshqalar Informatika. Axborot texnologiyalari. O'quv qo'llanma, 1-2-qism. -Toshkent. 2005, TDTU 334 b. va 394 b.
- 5.A.M.Polatov.Algoritmlar va C++ tilida dasturlash asoslari. O'quv qo'llanma. – Toshkent "Universitet" 2017. 55 b

Internet tarmog'idagi elektron manbalar:

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>
3. www.intuit.ru
4. <http://virtual-university.eurasia.org>
5. <http://www.opennet.ru>
6. www.linux.org.ru
7. <https://nashol.com>

12-ma'ruza. Mathcad dasturidan foydalanib masalalar va sodda differensial tenglamalarni yechish.

Reja:

- 1.Oddiy va murakkab matematik masalalarni MathCad dasturi yordamida yechishni o'rganish.
- 2.O'zgaruvchilar belgilash, ulardan ishchi oynaning barcha joylarida foydalanish imkoniyati.
- 3.Foydalanuvchi interfeysi.

Tayanch so'z va iboralar: MATHCAD, diagramma, Calculator, Graph, Calculus, Untitled, Greek– Symbolik, Programming, interfeys.

Mathcad turli-tuman ilmiy va muxandislik hisoblashlarni bajaruvchi matematik redaktordir. Mathcad vositasida elementar arifmetik amallardan tortib murakkab sonli metodlar realizastiyasini amalga oshirish mumkin. Sodda interfeysi, matematik hisoblashlarning ko'rgazmaliligi, keng standart funkstiyalar va sonli metodlar kutubxonasi mavjudligi, simvolli hisoblash xamda natijalarni turli shakllarda taqdim etish imkoniyatlari Mathcad dasturini eng ommaviy matematik dasturiy ta'minot darajasiga chiqishiga sabab bo'ldi. Mathcad tarkibiga bir-biri bilan integrallashgan bir necha komponent kiradi. Bular:

- Matematik ifoda va matnlarni kiritish, taxrirlash va formatlash imkonini beruvchi matnli muxarrir;
- Standart sonli metodlardan foydalanib, kiritilgan formulalar bo'yicha hisoblashlarni bajaruvchi prostessor;
- Analitik hisoblarni bajarishga imkon beruvchi simvolli prostessor;
- Interaktiv elektron kitob ko'rinishida matematik va muxandislik ma'lumotnomasi;

Mathcad dasturi muhitida quyidagi masalalarni hal etish mumkin:

- Mathcad formula redaktori yordamida matematik formulalar kiritish. Ushbu redaktor imkoniyatlari Microsoft Word formula redaktoridan qolishmaydi;
- Kiritilgan formulalar bo'yicha matematik hisoblashlar birdaniga bajariladi;
- Turli tipdagi grafiklarni xujjatga joylashtirish;
- Turli formatdagi fayllarga ma'lumotlarni kiritish va chiqarish;
- Xujjatlarni Mathcad da bosmaga chiqarish;
- Yaratilgan xujjatlarni elektron kitob ko'rinishida birlashtirish; Mathcad dasturi komponentlari turli-tuman matematik xisoblashlar uchun qulay muxit bo'lib, bir vaqtning o'zida bu xisoblashlar natijalarini xujjatlashtirish imkonini beradi.

Dasturni yuklash buyrug'idan keyin Mathcad dasturi oynasi ekranga chiqariladi. Uning ko'rinishi ifodalangan.

Quyida Mathcad dasturi asosiy menyusi bo'limlari keltirilgan.

- Mathcad dasturi oynasi tizimi menyusini chiqaruvchi tugma.
- File- fayl va xujjatlarni yaratish, saqlash, electron pochtdan jo'natish yoki printerda chop etish bilan bog'liq buyruqlar to'plami;
- Edit- matnlarni tahrirlash uchun mo'ljallangan bo'yruqlar to'plami;

- View- mathcad ishchi oynasida xujjatlarning tashqi ko'rinishini boshqaruvchi buyruqlar to'plami;
- Insert- xujjatga turli ob'ektlarni joylashtirish uchun xizmat qiluvchi buyruqlar to'plami;
- Format- matn, formula va grafiklarni formatlovchi buyruqlar to'plami;
- Math- hisoblash jarayonini boshqaruvchi buyruqlar to'plami;
- Symbolics- simvolli hisoblashlar bo'yruqlari to'plami;
- Windows- turli xujjat oynalarini ekranda jaylashtirish buyruqlari to'plami;
- Help- yordamchi axborotlarni chiqarish buyruqlari;

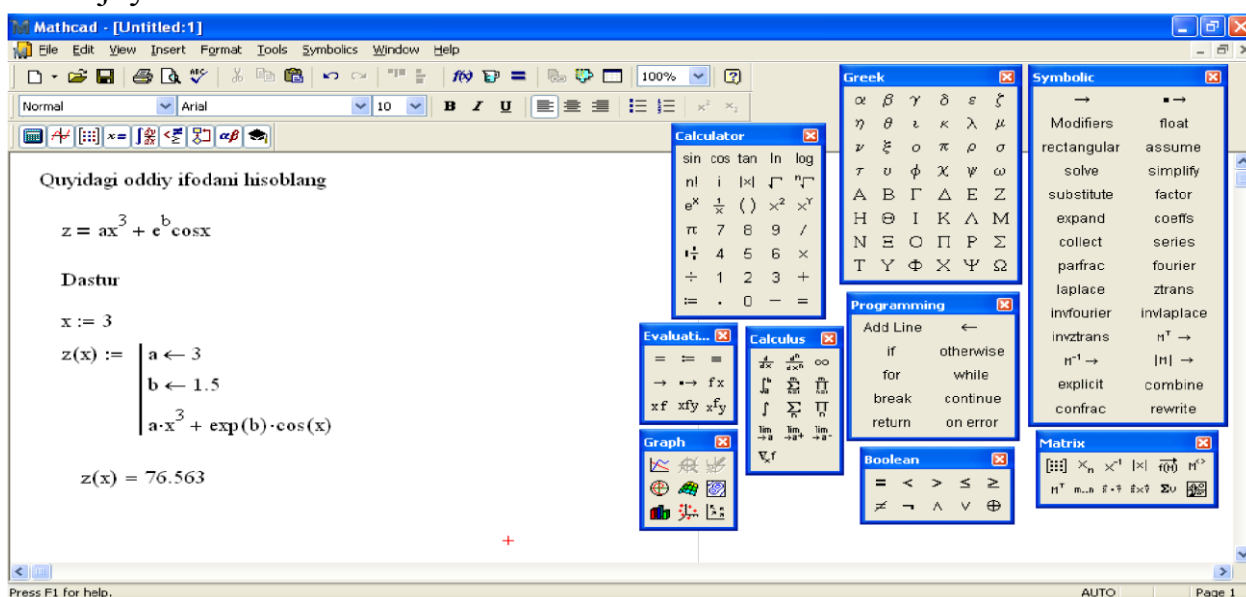
Asboblari paneli ko'p ishlatiluvchi buyruqlarga tez murojaat etish imkoniyatini beradi. Bu yerda "Standart asboblari paneli", "matematik asboblari paneli" va "formatlash uskunalari paneli" ni ajratish mumkin.

Dasturlash MATHCADda asosiy o'rin tutadi. MATHCAD ko'plab masalalarni dastursiz yechish imkonini beradi. Lekin shunday sinf masalalari borki ularni dastursiz yechib bo'lmaydi. MATHCAD har qanday murakkab dasturni kiritish imkonini beradi.

MATHCADda dasturlash juda aniq va tushunarli, unda dastur bir necha ketma-ket formulalarni ifodalaydi. Dasturlashning asosiy operatorlari Programming (Dasturlash) panelida joylashgan.

Dastur qatorini kiritish. Dasturni tuzish uchun uning qatorlarini kiritish kerak bo'ladi. Bu quyidagi keltirilgan protsedurada bajariladi:

1. Dastur ifodasi nomini kiritish.
2. Yuborish operatorini (:=) kiritish.
3. Dasturlash panelidan Add Program Line (Dastur qatorini qo'shish) tugmasini bosish.
4. Paydo bo'lgan kiritish joyiga kerakli operatorlarni kiritish, ortiqcha kiritish joyini olib tashlash.



12.1-rasm. Oddiy chiziqli dasturlar tuzish.

Kerakli kiritish qatorini ochish uchun ko'k burchakli kursorni qator oxiriga keltirib, bo'shlik tugmasini bosgan holda Add Program Line tugmasini bosish kerak. Agar kiritish qatorini qator oldidan ochish kerak bo'lsa ko'k burchakli kursorni qator boshiga keltirib, bo'shlik tugmasini bosgan holda Add Program Line tugmasini bosish kerak bo'ladi (12.1-rasm).

Ayrim hollarda, masalan ikki ichma ich joylashgan sikllar orasigi qator qo'shishda bu usul qo'l kelmay qoladi. Bu holda boshqa usulni qo'llashga to'g'ri keladi. Bu usul quyidagicha bajariladi:

1. Sikl ichi qora rangga ajratiladi.
2. Standart vositalar panelidan kesib olish (Cut) tugmasi bosiladi.
3. Add Progrm Line (dasturga qator qo'shish) dasturlash paneli tugmasi bosiladi.
4. Qator kiritish joyiga kursor qo'yilib, standart vositalar panelidan qo'yish (Paste) tugmasi bosiladi.
5. Paydo bo'lgan kiritish joyi to'ldiriladi.

Bu usul barcha hollarda ham qator kiritishda qulaylikni beradi.

Dasturda qiymatlarni lokal yuborish. Dasturda o'zgarmaslar va o'zgaruvchilarga qiymatlari berish ($\leftarrow$) yuborish operatori yordamida amalga oshiriladi.

Bu operator dasturlash panel vositasida (Local Definition) lokal aniqlash tugmasiga birlashtirilgan. Dastur tuzish davomida ko'p hollarda bu belgini klaviaturadan { belgisini bosish bilan ham bajarish mumkin.

Lokal o'zgaruvchi qiymatini dastur tashqarisida ishlatish mumkin emas. Agar tashqarida ishlatish juda kerak bo'lsa, uning uchun dasturning eng oxirgi operatoridan keyin kursorni bo'sh joyga qo'yib, keyin o'zgaruvchini yozish kerak bo'ladi. Agar o'zgaruvchining unga mos bitta qiymatini chiqarish kerak bo'lsa, shu o'zgaruvchining nomini yozish kerak. Agar vektor yoki massivni chiqarish kerak bo'lsa uning nomini kiritish kerak.

Nazorat savollari:

- 1.Qanday dasturlash tillarini bilasiz?
- 2.Mathcad dasturini yuklash ketma-ketligini aytib bering.
- 3.Mathcad dasturida nechta menyu bor?"
- 4.Mathcad dasturining 6 ta xarakteristikasini sanab bering.

Mavzuga oid adabiyotlar:

- 1.R.Hamdamiyov, U.Begimqulov, N.Tayloqov. Ta'limda axborot texnologiyalari. "O'zbekiston Milliy Ensiklopediyasi" Davlat ilmiy nashriyoti. Toshkent - 2010
2. S.S. G'ulomov, B.A.Begalov. Informatika va axborot texnologiyalari. Fan, T: 2010. 686 b.
3. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
4. M.M. Aripov, T. Imomov va boshqalar Informatika. Axborot texnologiyalari. O'quv qo'llanma, 1-2-qism. -Toshkent. 2005, TDTU 334 b. va

394 b.

5.A.M.Polatov.Algoritmlar va C++ tilida dasturlash asoslari. O'quv qo'llanma. – Toshkent “Universitet” 2017. 55 b

Internet tarmog'idagi elektron manbalar:

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>
3. www.intuit.ru
4. [http://virlual-universily eurasia.org](http://virlual-universily Eurasia.org)
5. <http://www.opennet.ru>
6. www.linux.org.ru
7. <https://nashol.com>
8. https://www.youtube.com/video urok_Project_

13-ma'ruza. Mathcad dasturidan foydalanib masalalar va sodda differensial tenglamalarni yechish.

Reja:

- 1.Oddiy va murakkab matematik masalalarni MathCad dasturi yordamida yechishni o'rganish.
- 2.O'zgaruvchilar belgilash, ulardan ishchi oynaning barcha joylarida foydalanish imkoniyati.
- 3.Foydalanuvchi interfeysi.

Tayanch so'z va iboralar: MATHCAD, diagramma, Calculator, Graph, Calculus, Untitled, Greek– Symbolik, Programming, interfeys.

Mathcad turli-tuman ilmiy va muxandislik hisoblashlarni bajaruvchi matematik redaktordir. Mathcad vositasida elementar arifmetik amallardan tortib murakkab sonli metodlar realizastiyasini amalga oshirish mumkin. Sodda interfeysi, matematik hisoblashlarning ko'rgazmaliligi, keng standart funkstiyalar va sonli metodlar kutubxonasi mavjudligi, simvolli hisoblash xamda natijalarni turli shakllarda taqdim etish imkoniyatlari Mathcad dasturini eng ommaviy matematik dasturiy ta'minot darajasiga chiqishiga sabab bo'ldi. Mathcad tarkibiga bir-biri bilan integrallashgan bir necha komponent kiradi. Bular:

- Matematik ifoda va matnlarni kiritish, taxrirlash va formatlash imkonini beruvchi matnli muxarrir;
- Standart sonli metodlardan foydalanib, kiritilgan formulalar bo'yicha hisoblashlarni bajaruvchi prostessor;
- Analitik hisoblarni bajarishga imkon beruvchi simvolli prostessor;
- Interaktiv elektron kitob ko'rinishida matematik va muxandislik ma'lumotnomasi;

Mathcad dasturi muhitida quyidagi masalalarni hal etish mumkin:

- Mathcad formula redaktori yordamida matematik formulalar kiritish.

Ushbu redaktor imkoniyatlari Microsoft Word formula redaktoridan qolishmaydi;

- Kiritilgan formulalar bo'yicha matematik hisoblashlar birdaniga bajariladi;
- Turli tipdagi grafiklarni xujjatga joylashtirish;
- Turli formatdagi fayllarga ma'lumotlarni kiritish va chiqarish;
- Xujjatlarni Mathcad da bosmaga chiqarish;
- Yaratilgan xujjatlarni elektron kitob ko'rinishida birlashtirish; Mathcad dasturi komponentlari turli-tuman matematik hisoblashlar uchun qulay muxit bo'lib, bir vaqtning o'zida bu hisoblashlar natijalarini xujjatlashtirish imkonini beradi.

Dasturni yuklash buyrug'idan keyin Mathcad dasturi oynasi ekranga chiqariladi. Uning ko'rinishi ifodalangan.

Quyida Mathcad dasturi asosiy menyusi bo'limlari keltirilgan.

- Mathcad dasturi oynasi tizimi menyusini chiqaruvchi tugma.
- File- fayl va xujjatlarni yaratish, saqlash, electron pochtdan jo'natish yoki printerda chop etish bilan bog'liq buyruqlar to'plami;
- Edit- matnlarni tahrirlash uchun mo'ljallangan bo'yuqlar to'plami;
- View- mathcad ishchi oynasida xujjatlarning tashqi ko'rinishini boshqaruvchi buyruqlar to'plami;
- Insert- xujjatga turli ob'ektlarni joylashtirish uchun xizmat qiluvchi buyruqlar to'plami;
- Format- matn, formula va grafiklarni formatlovchi buyruqlar to'plami;
- Math- hisoblash jarayonini boshqaruvchi buyruqlar to'plami;
- Symbolics- simvolli hisoblashlar bo'yuqlari to'plami;
- Windows- turli xujjat oynalarini ekranda joylashtirish buyruqlari to'plami;
- Help- yordamchi axborotlarni chiqarish buyruqlari;

Asboblar paneli ko'p ishlatiluvchi buyruqlarga tez murojaat etish imkoniyatini beradi. Bu yerda "Standart asboblar paneli", "matematik asboblar paneli" va "formatlash uskunalar paneli" ni ajratish mumkin.

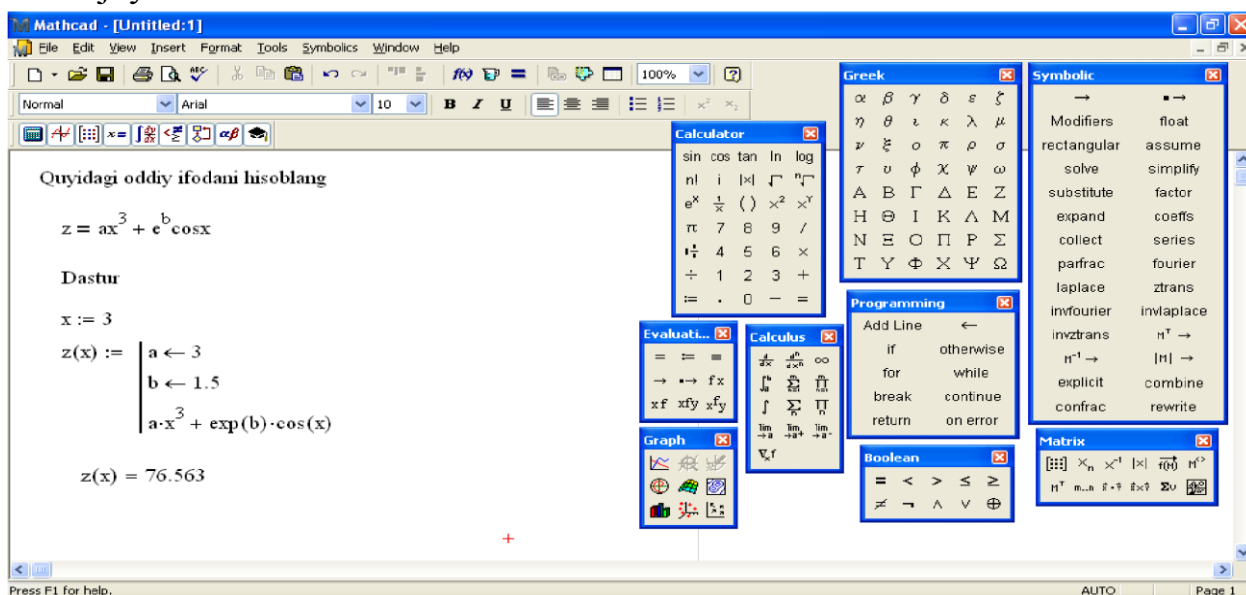
Dasturlash MATHCADda asosiy o'rin tutadi. MATHCAD ko'plab masalalarni dastursiz yechish imkoninii beradi. Lekin shunday sinf masalalari borki ularni dastursiz yechib bo'lmaydi. MATHCAD har qanday murakkab dasturni kiritish imkonini beradi.

MATHCADda dasturlash juda aniq va tushunarli, unda dastur bir necha ketma-ket formulalarni ifodalaydi. Dasturlashning asosiy operatorlari Programming (Dasturlash) panelida joylashgan.

Dastur qatorini kiritish. Dasturni tuzish uchun uning qatorlarini kiritish kerak bo'ladi. Bu quyidagi keltirilgan protsedurada bajariladi:

5. Dastur ifodasi nomini kiritish.
6. Yuborish operatorini (:=) kiritish.
7. Dasturlash panelidan Add Program Line (Dastur qatorini qo'shish) tugmasini bosish.

8. Paydo bo'lgan kiritish joyiga kerakli operatorlarni kiritish, ortiqcha kiritish joyini olib tashlash.



13.1-rasm. Oddiy chiziqli dasturlar tuzish.

Kerakli kiritish qatorini ochish uchun ko'k burchakli kursorni qator oxiriga keltirib, bo'shlik tugmasini bosgan holda Add Program Line tugmasini bosish kerak. Agar kiritish qatorini qator oldidan ochish kerak bo'lsa ko'k burchakli kursorni qator boshiga keltirib, bo'shlik tugmasini bosgan holda Add Program Line tugmasini bosish kerak bo'ladi (13.1-rasm).

Ayrim hollarda, masalan ikki ichma ich joylashgan sikllar orasigi qator qo'shishda bu usul qo'l kelmay qoladi. Bu holda boshqa usulni qo'llashga to'g'ri keladi. Bu usul quyidagicha bajariladi:

1. Sikl ichi qora rangga ajratiladi.
2. Standart vositalar panelidan kesib olish (Cut) tugmasi bosiladi.
3. Add Progm Line (dasturga qator qo'shish) dasturlash paneli tugmasi bosiladi.
4. Qator kiritish joyiga kursor qo'yilib, standart vositalar panelidan qo'yish (Paste) tugmasi bosiladi.
5. Paydo bo'lgan kiritish joyi to'ldiriladi.

Bu usul barcha hollarda ham qator kiritishda qulaylikni beradi.

Dasturda qiymatlarni lokal yuborish. Dasturda o'zgarmaslar va o'zgaruvchilarga qiymatlari berish ($\leftarrow$) yuborish operatori yordamida amalga oshiriladi.

Bu operator dasturlash panel vositasida (Local Definition) lokal aniqlash tugmasiga birlashtirilgan. Dastur tuzish davomida ko'p hollarda bu belgini klaviaturadan { belgisini bosish bilan ham bajarish mumkin.

Lokal o'zgaruvchi qiymatini dastur tashqarisida ishlatish mumkin emas. Agar tashqarida ishlatish juda kerak bo'lsa, uning uchun dasturning eng oxirgi operatoridan keyin kursorni bo'sh joyga qo'yib, keyin o'zgaruvchini yozish kerak bo'ladi. Agar o'zgaruvchining unga mos bitta qiymatini chiqarish kerak bo'lsa, shu o'zgaruvchining nomini yozish kerak. Agar vektor yoki massivni chiqarish kerak bo'lsa uning nomini kiritish kerak.

Nazorat savollari:

- 1.Qanday dasturlash tillarini bilasiz?
- 2.Mathcad dasturini yuklash ketma-ketligini aytib bering.
- 3.Mathcad dasturida nechta menyu bor”
- 4.Mathcad dasturining 6 ta xarakteristikasini sanab bering.

Mavzuga oid adabiyotlar:

- 1.R.Hamdamov, U.Begimqulov, N.Tayloqov. Ta’limda axborot texnologiyalari. “O’zbekiston Milliy Ensiklopediyasi” Davlat ilmiy nashriyoti. Toshkent - 2010
2. S.S. G`ulomov, B.A.Begalov. Informatika va axborot texnologiyalari. Fan, T: 2010. 686 b.
3. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o‘quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
4. M.M. Aripov, T. Imomov va boshqalar Informatika. Axborot texnologiyalari. O‘quv qo‘llanma, 1-2-qism. -Toshkent. 2005, TDTU 334 b. va 394 b.
- 5.A.M.Polatov.Algoritmlar va C++ tilida dasturlash asoslari. O‘quv qo‘llanma. – Toshkent “Universitet” 2017. 55 b

Internet tarmog’idagi elektron manbalar:

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>
3. www.intuit.ru
4. <http://virlual-universilyeurasia.org>
5. <http://www.opennet.ru>

14-ma’ruza.Tasvir va grafik yaratish.

Reja:

1.Ikki o'lchamli grafik qurish.

2.Uch o'lchamli grafik qurish.

Tayanch so’z va iboralar: MATHCAD, ikki o’lchamli grafik,interfeys, argument, shablon, Label, simvol.

Ikki o'lchamli grafik qurish. Ikki o'lchamli funksiya grafigini qurish uchun quyidagi protseduralarni bajarish kerak.

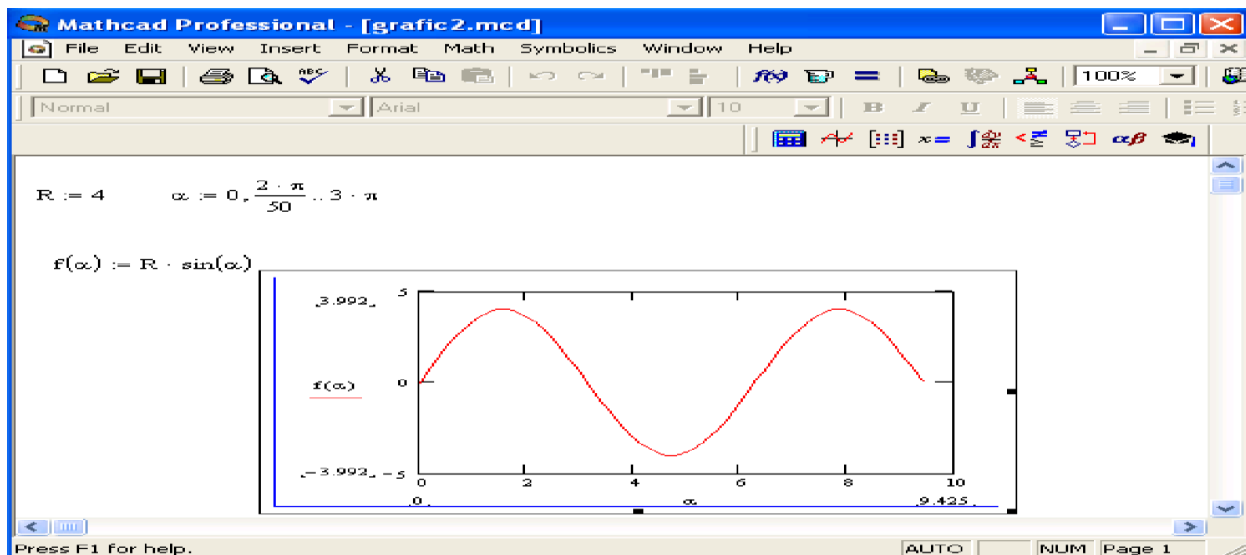
1.Qaysi joyga grafik qurish kerak bo'lsa, shu joyga krestli kursor qo'yiladi.

2.Matematik panelining Graph (Grafik) panelidan x-y Plot (Ikki o'lchovli grafik) tugmasi bosiladi.

3.Hosil bo'lgan ikki o'lchamli grafik shabloniga abssiss o'qi argumenti nomi, ordinata o'qiga funksiya nomi kiritiladi.

4.Argumentning berilgan o'zgarish diapazonida grafikni qurish uchun grafik shabloni tashqarisi sichqonchada bosiladi. Agar argumentning diapazon qiymati berilmasa, u holda avtomatik holda argument diapazon qiymati 10 dan 10 gacha bo'ladi va shu diapazonda grafik quriladi (Masalan, rasm 1).

Agar bir necha funksiyalar grafigini qurish kerak bo'lsa va ular argumentlari har xil bo'lsa, u holda grafikda funksiyalar va argumentlar nomlari ketma-ket vergul qo'yilib kiritiladi. Bunda birinchi grafik birinchi argument bo'yicha birinchi funksiya grafi gini va ikkinchisi esa mos ravishda ikkinchi argument bo'yicha ikkinchi funksiya grafigini tasvirlaydi va hakoza.



14.1-rasm. Funksiya grafigini qurish.

Quyida grafik formati muloqot oynasi qo'yilmalarini beramiz:

1. X-Y Axes - koordinata o'qini formatlash. Koordinata o'qiga setka, sonli qiymatlarni grafikga belgilarni qo'yish va quyidagilarni o'rnatish mumkin:

- LogScale - logarifmik masshtabda o'qga sonli qiymatlarni tasvirlash;
- Grid Lines - chiziqqa setkalar qo'yish;
- Numbered - koordinata o'qi bo'yicha sonlarni qo'yish;
- Auto Scale - son qiymatlar chegarasini o'qda avtomatik tanlash;
- Show Markers - grafikka belgi kiritish;
- avtomatik tanlash. Autogrid - chiziq setkasi sonini

2. Trace - funksiya grafiklarini formatlash. Har bir funksiya grafigini alohida o'zgartish mumkin:

- chiziq ko'rinishi (Solid - uzliksiz, Dot - punktir, Dash - shtrixli, Dadot - shtrixli punktir);
- chiziq rangi (Color);
- grafik tipi (Type) (Lines - chiziq, Points - nuqtali, Bar yoki SolidBar - ustunli, Step - pog'onali grafik va boshqa);
- chiziq qalinligi (Weight);
- simvol (Symbol) - grafikda hisoblangan qiymatlar uchun (aylana, krestik, to'g'ri burchak, romb).

3. Label - grafik maydoni sarlovhasi. Title (Sarlovha) maydoniga sarlovha matni kiritiladi.

4. Defaults - bu qo'yilma yordamida grafik ko'rinishga qaytish mumkin.

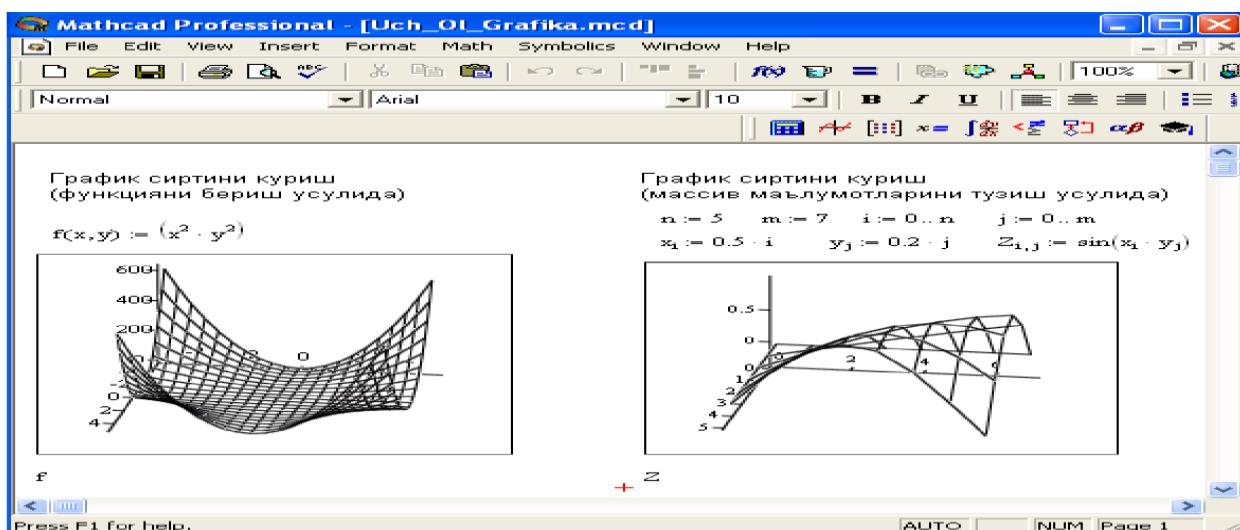
Uch o'lchamli grafik qurish. Uch o'lchamli grafik qurish uchun quyidagi protseduralarni bajarish kerak.

1. Ikki o'zgaruvchili funksiya nomini keyin ($:=$) yuborish operatori va funksiya ifodasini kiritish.

2. Grafik qurish kerak bo'lgan joyga kursor qo'yiladi.

3. Matematik panelining Graph (Grafik) panelidan Surface Plot (uch o'lchamli grafik) tugmasi bosiladi. SHu joyda uch o'lchamli grafik shabloni paydo bo'ladi.

4. SHablon maydonidan tashqarisida sichqoncha bosiladi va grafik quriladi, masalan, 14.2-rasm chap tomon.



14.2-rasm. Ikki o'zgaruvchili funksiya grafigini qurish.

Ikki o'zgaruvchili funksiya bo'yicha grafik sirtini qurishni tez qilish maqsadida boshqa usul ham mavjud va u ayrim hollarda funksiya sirtini tuzishda funksiya massiv sonli qiymatlarini ishlatadi. Bunday grafikni qurish uchun quyidagi protseduralarni bajarish kerak.

1. Diskret o'zgaruvchilar yordamida ikki funksiyaning o'zgaruvchisi uchun ham qiymatlarini kiritish.

2. Massiv kiritish. Uning elementlari funksiya qiymatlari bo'lib, ular berilgan funksiya argumentlari qiymatlaridan tashkil etiladi.

3. Kursor qaysi joyga grafik qurish kerak bo'lsa shu joyga qo'yiladi.

4. Grafik shabloniga funksiya nomini kiritish.

5. SHablon maydonidan tashqarisida sichqoncha bosiladi va grafik quriladi.

Grafik formatini qayta o'zgartirish va unga ranglar berish uchun grafik maydonini ikki marta tez-tez sichqonchani ko'rsatib bosish va ochilgan muloqot oynasidan kerakli o'zgarishlarni qilish kerak. Bu o'zgartirishlar muloqot oynasi 14.3-rasmda berilgan.

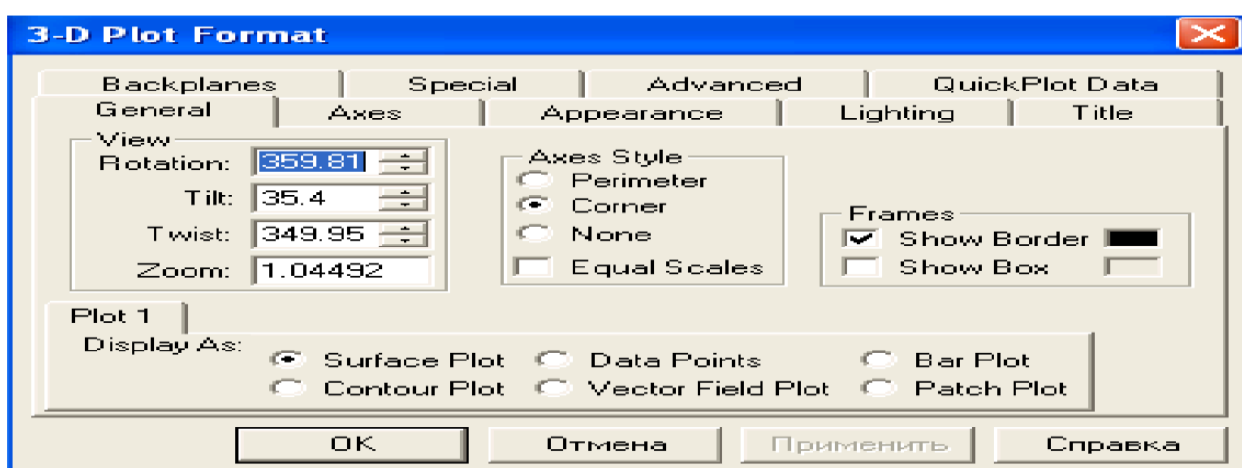
Bunda:

- Surface Plot - grafik sirti;
- Contour Plot - grafik chizig'i darajasi;
- Data Points - grafikda faqat hisob nuqtalarini tasvirlash;
- Vector Field Plot - vektor maydoni grafigi;
- Bar Plot - uch o'lchovli grafik gistogrammasi;
- Patch plot - hisob qiymatlari maydoni.

Bulardan tashqari yana bir qancha boshqarish elementlari mavjud. Ular grafikni formatlashda keng imkoniyatni beradi. Masalan, grafik masshtabini o'zgartirish, grafikni aylantirish, grafikga animatsiya berish va boshqa. 14.3-rasmda uch o'lchamli grafikni formatlash oynasi berilgan.

Grafikni boshqarishning boshqa usullari quyidagilar:

- *Grafikni aylantirish* uni ko'rsatib sichqoncha o'ng tugmasini bosish bilan amalga oshiriladi.
- *Grafikni masshtablashtirish* Ctrl tugmasini bosib sichqoncha orqali bajariladi.
- *Grafikga animatsiya* berish Shift tugmasini bosish bilan sichqoncha orqali amalga oshiriladi.



14.3-rasm. Uch o'lchamli grafikni formatlash oynasi.

Nazorat savollari:

1. Ikki o'lchamli funksiya grafigini qurish uchun qanday ishlarni bajarish kerak?
2. Grafikga animatsiya berish uchun qaysi tugmadan foydalaniladi.
3. Mathcadda bosmaga chiqarish uchun qaysi buyruq beriladi?

Mavzuga oid adabiyotlar:

1. R. Hamdamov, U. Begimqulov, N. Tayloqov. Ta'limda axborot texnologiyalari. "O'zbekiston Milliy Ensiklopediyasi" Davlat ilmiy nashriyoti. Toshkent - 2010
2. S.S. G'ulomov, B.A. Begalov. Informatika va axborot texnologiyalari. Fan, T: 2010. 686 b.
3. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
4. M.M. Aripov, T. Imomov va boshqalar Informatika. Axborot texnologiyalari. O'quv qo'llanma, 1-2-qism. -Toshkent. 2005, TDTU 334 b. va 394 b.
5. A.M. Polatov. Algoritmlar va C++ tilida dasturlash asoslari. O'quv qo'llanma. – Toshkent "Universitet" 2017. 55 b

Internet tarmog'idagi elektron manbalar:

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>

3. www.intuit.ru
4. <http://virtual-university.eurasia.org>
5. <http://www.opennet.ru>
6. www.linux.org.ru
7. <https://nashol.com>
8. https://www.youtube.com/video/urok_Project_

15-ma'ruza. Tasvir va grafik yaratish.

Reja:

1. Ikki o'lchamli grafik qurish.

2. Uch o'lchamli grafik qurish.

Tayanch so'z va iboralar: MATHCAD, ikki o'lchamli grafik, interfeys, argument, shablon, Label, simvol.

Ikki o'lchamli grafik qurish. Ikki o'lchamli funksiya grafigini qurish uchun quyidagi protseduralarni bajarish kerak.

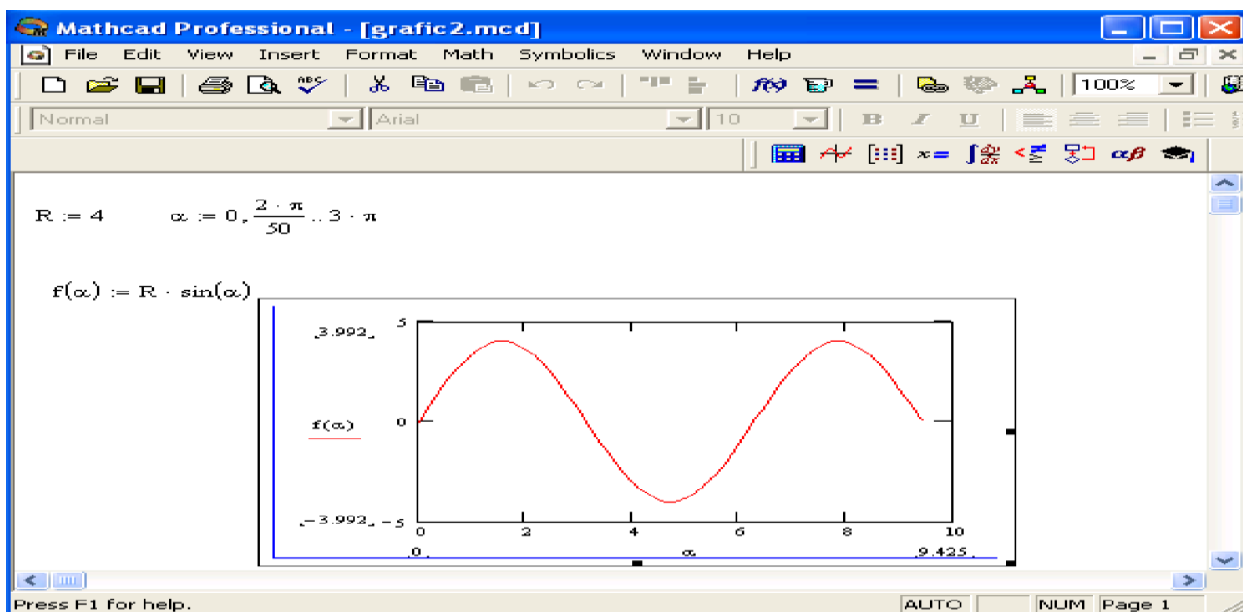
1. Qaysi joyga grafik qurish kerak bo'lsa, shu joyga krestli kursor qo'yiladi.

2. Matematik panelining Graph (Grafik) panelidan x-y Plot (Ikki o'lchovli grafik) tugmasi bosiladi.

3. Hosil bo'lgan ikki o'lchamli grafik shabloniga abssiss o'qi argumenti nomi, ordinata o'qiga funksiya nomi kiritiladi.

4. Argumentning berilgan o'zgarish diapazonida grafikni qurish uchun grafik shabloni tashqarisi sichqonchada bosiladi. Agar argumentning diapazon qiymati berilmasa, u holda avtomatik holda argument diapazon qiymati 10 dan 10 gacha bo'ladi va shu diapazonda grafik quriladi (Masalan, rasm 1).

Agar bir necha funksiyalar grafigini qurish kerak bo'lsa va ular argumentlari har xil bo'lsa, u holda grafikda funksiyalar va argumentlar nomlari ketma-ket vergul qo'yilib kiritiladi. Bunda birinchi grafik birinchi argument bo'yicha birinchi funksiya grafi gini va ikkinchisi esa mos ravishda ikkinchi argument bo'yicha ikkinchi funksiya grafigini tasvirlaydi va hakoza.



15.1-rasm. Funksiya grafigini qurish.

Quyida grafik formati muloqot oynasi qo'yilmalarini beramiz:

1. X-Y Axes - koordinata o'qini formatlash. Koordinata o'qiga setka, sonli qiymatlarni grafikga belgilarni qo'yish va quyidagilarni o'rnatish mumkin:

- LogScale - logarifmik masshtabda o'qga sonli qiymatlarni tasvirlash;
- Grid Lines - chiziqqa setkalar qo'yish;
- Numbered - koordinata o'qi bo'yicha sonlarni qo'yish;
- Auto Scale - son qiymatlar chegarasini o'qda avtomatik tanlash;
- Show Markers - grafikka belgi kiritish;
- avtomatik tanlash. Autogrid - chiziq setkasi sonini

2. Trace - funksiya grafiklarini formatlash. Har bir funksiya grafigini alohida o'zgartish mumkin:

- chiziq ko'rinishi (Solid - uzliksiz, Dot - punktir, Dash - shtrixli, Dadot - shtrixli punktir);
- chiziq rangi (Color);
- grafik tipi (Type) (Lines - chiziq, Points - nuqtali, Bar yoki SolidBar - ustunli, Step - pog'onali grafik va boshqa);
- chiziq qalinligi (Weight);
- simvol (Symbol) - grafikda hisoblangan qiymatlar uchun (aylana, krestik, to'g'ri burchak, romb).

3. Label - grafik maydoni sarlovhasi. Title (Sarlovha) maydoniga sarlovha matni kiritiladi.

4. Defaults - bu qo'yilma yordamida grafik ko'rinishga qaytish mumkin.

Uch o'lchamli grafik qurish. Uch o'lchamli grafik qurish uchun quyidagi protseduralarni bajarish kerak.

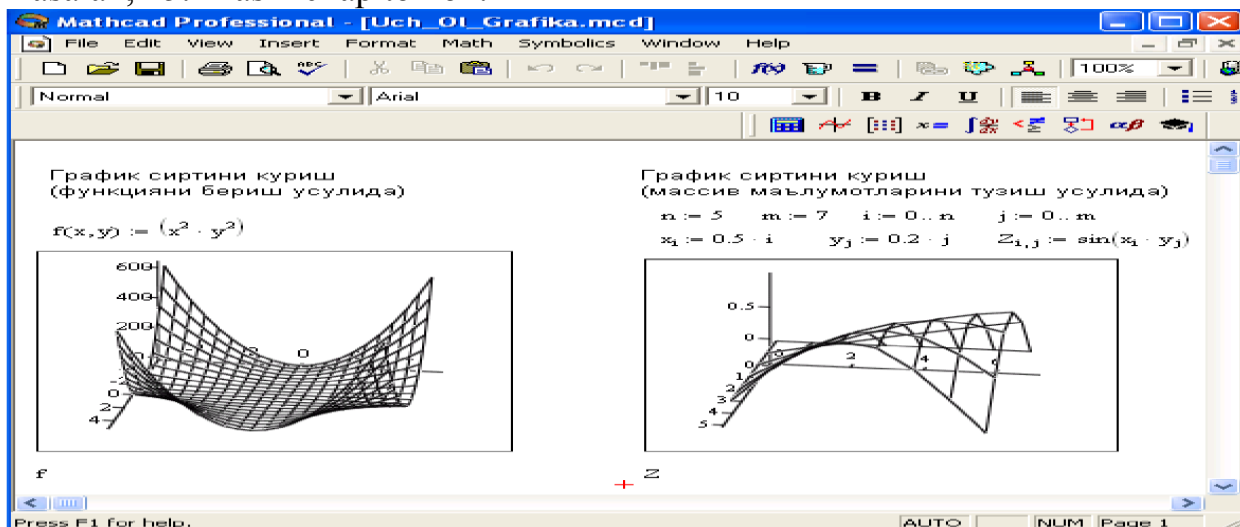
1. Ikki o'zgaruvchili funksiya nomini keyin ($:=$) yuborish operatori va funksiya ifodasini kiritish.

2. Grafik qurish kerak bo'lgan joyga kursor qo'yiladi.

3. Matematik panelining Graph (Grafik) panelidan Surface Plot (uch o'lchamli grafik) tugmasi bosiladi. SHu joyda uch o'lchamli grafik shablona paydo

bo'ladi.

4. SHablon maydonidan tashqarisida sichqoncha bosiladi va grafik quriladi, masalan, 15.2-rasm chap tomon.



15.2-rasm. Ikki o'zgaruvchili funktsiya grafigini qurish.

Ikki o'zgaruvchili funktsiya bo'yicha grafik sirtini qurishni tez qilish maqsadida boshqa usul ham mavjud va u ayrim hollarda funktsiya sirtini tuzishda funktsiya massiv sonli qiymatlarini ishlatadi. Bunday grafikni qurish uchun quyidagi protseduralarni bajarish kerak.

1. Diskret o'zgaruvchilar yordamida ikki funktsiyaning o'zgaruvchisi uchun ham qiymatlarini kiritish.

2. Massiv kiritish. Uning elementlari funktsiya qiymatlari bo'lib, ular berilgan funktsiya argumentlari qiymatlaridan tashkil etiladi.

3. Kursor qaysi joyga grafik qurish kerak bo'lsa shu joyga qo'yiladi.

4. Grafik shabloniga funktsiya nomini kiritish.

5. SHablon maydonidan tashqarisida sichqoncha bosiladi va grafik quriladi.

Grafik formatini qayta o'zgartirish va unga ranglar berish uchun grafik maydonini ikki marta tez-tez sichqonchani ko'rsatib bosish va ochilgan muloqot oynasidan kerakli o'zgarishlarni qilish kerak. Bu o'zgartirishlar muloqot oynasi 15.3-rasmda berilgan.

Bunda:

- Surface Plot - grafik sirti;
- Contour Plot - grafik chizig'i darajasi;
- Data Points - grafikda faqat hisob nuqtalarini tasvirlash;
- Vector Field Plot - vektor maydoni grafigi;
- Bar Plot - uch o'lchovli grafik gistogrammasi;
- Patch plot - hisob qiymatlari maydoni.

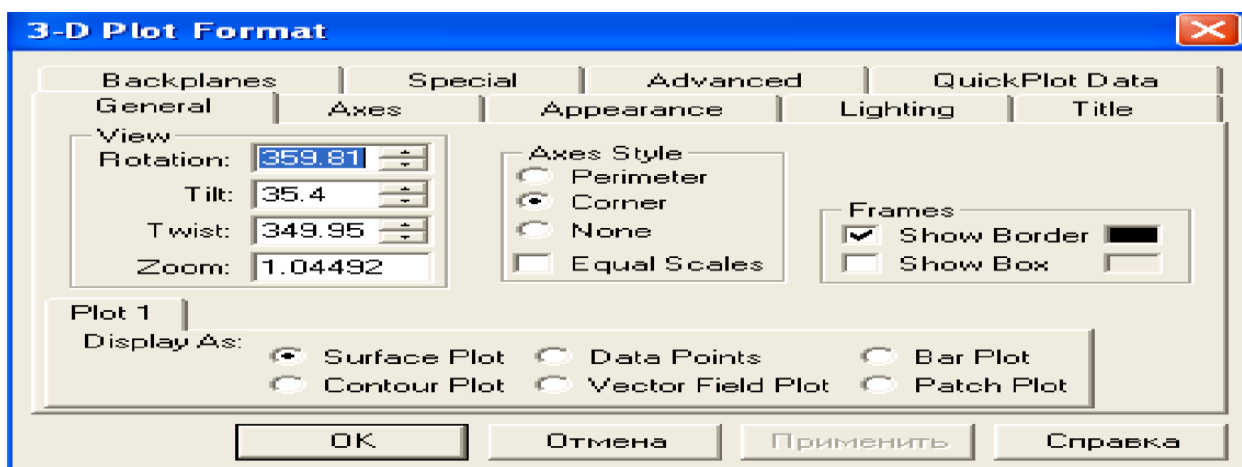
Bulardan tashqari yana bir qancha boshqarish elementlari mavjud. Ular grafikni formatlashda keng imkoniyatni beradi. Masalan, grafik masshtabini o'zgartirish, grafikni aylantirish, grafikga animatsiya berish va boshqa. 15.3-rasmda uch o'lchamli grafikni formatlash oynasi berilgan.

Grafikni boshqarishning boshqa usullari quyidagilar:

- Grafikni aylantirish uni ko'rsatib sichqoncha o'ng tugmasini bosish bilan amalga

oshiriladi.

- *Grafikni masshtablashtirish* Ctrl tugmasini bosib sichqoncha orqali bajariladi.
- *Grafikga animatsiya berish* Shift tugmasini bosish bilan sichqoncha orqali amalga oshiriladi.



15.3-rasm. Uch o'lchamli grafikni formatlash oynasi.

Nazorat savollari:

1. Ikki o'lchamli funksiya grafigini qurish uchun qanday ishlarni bajarish kerak?
2. Grafikga animatsiya berish uchun qaysi tugmadan foydalaniladi.
3. Mathcadda bosmaga chiqarish uchun qaysi buyruq beriladi?

Mavzuga oid adabiyotlar:

1. R. Hamdamov, U. Begimqulov, N. Tayloqov. Ta'limda axborot texnologiyalari. "O'zbekiston Milliy Ensiklopediyasi" Davlat ilmiy nashriyoti. Toshkent - 2010
2. S.S. G'ulomov, B.A. Begalov. Informatika va axborot texnologiyalari. Fan, T: 2010. 686 b.
3. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
4. M.M. Aripov, T. Imomov va boshqalar Informatika. Axborot texnologiyalari. O'quv qo'llanma, 1-2-qism. -Toshkent. 2005, TDTU 334 b. va 394 b.
5. A.M. Polatov. Algoritmlar va C++ tilida dasturlash asoslari. O'quv qo'llanma. –Toshkent "Universitet" 2017. 55 b

Internet tarmog'idagi elektron manbalar:

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>
3. www.intuit.ru
4. <http://virtual-university-urasia.org>
5. <http://www.opennet.ru>
6. www.linux.org.ru
7. <https://nashol.com>

8. https://www.youtube.com/video urok_Project_

16-ma'ruza. Mathcadda dasturlash.

Mavzu bo'yicha savollar to'plami:

1. Qanday dasturlash tillarini bilasiz?
2. Mathcad dasturini yuklash ketma-ketligini aytib bering.
3. Mathcad dasturida nechta menyu bor?
4. Mathcad dasturining 6 ta xarakteristikasini sanab bering.
5. Ikki o'lchamli funksiya grafigini qurish uchun qanday ishlarni bajarish kerak?
6. Grafikga animatsiya berish uchun qaysi tugmadan foydalaniladi.
7. Mathcadda bosmaga chiqarish uchun qaysi buyruq beriladi?

Mavzu bo'yicha glossariy:

LogScale - logarifmik masshtabda o'qga sonli qiymatlarni tasvirlash;
Trace - funksiya grafiklarini formatlash.
Label - grafik maydoni sarlavhasi.
Title (Sarlavha) maydoniga sarlovha matni kiritiladi.
Defaults - bu qo'yilma yordamida grafik ko'rinishga qaytish mumkin.
Calculator – asosiy matematik amallar shabloni; (Kalkulyator)
Graph – grafiklar shabloni; (Grafik)
Matrix (Massiv) – massiv va massiv amallarini bajarish shabloni; (Massiv)
Evaluation – qiymatlarni yuborish operatori va natijalarni chiqarish operatori; (Baholash)
Greek– Symbolik - belgililar ustida ishlash uchun operatorlar. (Grek alifbosi harflari)

Mavzuga oid adabiyotlar:

1. I.A.Bedarov, O.N.Belousova, N.N.Fedorova. ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ИНЖЕНЕРНЫХ ЗАДАЧ В ПАКЕТЕ MathCAD. Novosibirsk. 2005.
2. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
3. Iqtisodiy Informatika, S.S.G'ulomov va boshqalar. Sharq, 1999
4. Internet va undan foydalanish asoslari. A.Maraximov, S.Raxmonqulova, Virtec, 2001
5. Q.M. Karimov, I.D. Razzoqov. MATHCAD va Matlab muhitida ishlash. O'quv-uslubiy qo'llanma. Qarshi. "Nasaf" nashriyoti. 2014 y

Internet tarmog'idagi elektron manbalar:

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>
3. www.intuit.ru
4. <http://virtual-university-urasia.org>
5. <http://www.opennet.ru>
6. www.linux.org.ru
7. <https://nashol.com>

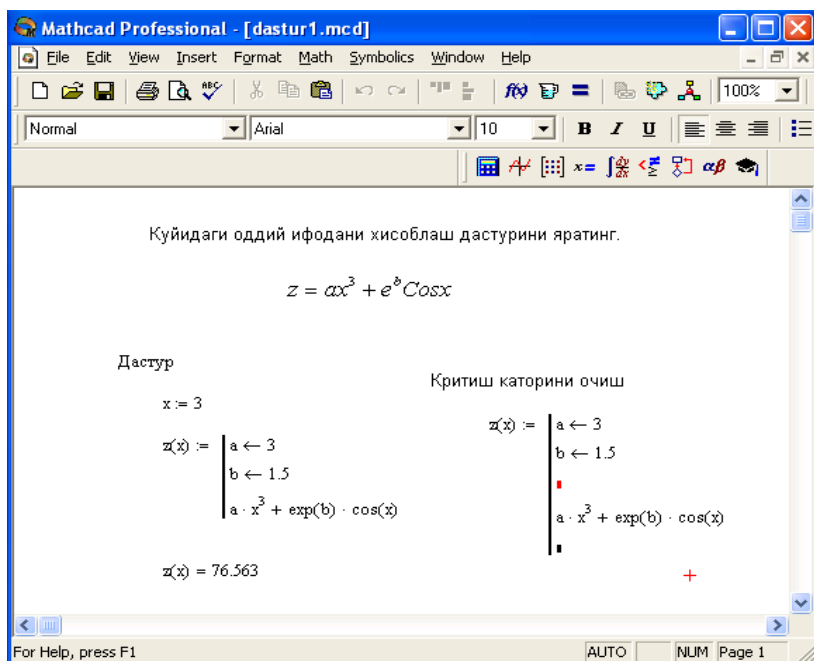
16.1.MathCadda dasturlash.

Dasturlash Mathcadda asosiy o`rin tutadi. Mathcad ko`plab masalalarni dastursiz echish imkoninii beradi. Lekin shunday sinf masalalari borki ularni dastursiz echib bo`lmaydi. Mathcad har qanday murakkab dasturni kiritish imkonini beradi. Mathcadda dasturlash juda aniq va tushunarli, unda dastur bir necha ketma-ket formulalarni ifodalaydi. Dasturlashning asosiy operatorlari Programming (Dasturlash) panelida joylashgan.

Dastur qatorini kiritish. Dasturni tuzish uchun uning qatorlarini kiritish kerak bo`ladi. Bu quyidagi keltirilgan protsedurada bajariladi:

1. Dastur ifodasi nomini kiritish.
2. Yuborish operatorini (:=) kiritish.
3. Dasturlash panelidan Add Program Line (Dastur qatorini qo`shish) tugmasini bosish.
4. Paydo bo`lgan kiritish joyiga kerakli operatorlarni kiritish, ortiqcha kiritish joyini olib tashlash.

Kerakli kiritish qatorini ochish uchun ko`k burchakli kursorni qator oxiriga keltirib, bo`shlik tugmasini bosgan holda Add Program Line tugmasini bosish kerak. Agar kiritish qatorini qator oldidan ochish kerak bo`lsa ko`k burchakli kursorni qator boshiga keltirib, bo`shlik tugmasini bosgan holda Add Program Line tugmasini bosish kerak bo`ladi.



16.1-rasm. Oddiy chiziqli dasturlar tuzish.

Ayrim hollarda, masalan ikki ichma-ich joylashgan sikllar orasiga qator qo`shishda bu usul qo`l kelmay qoladi. Bu holda boshqa usulni qo`llashga to`g`ri keladi. Bu usul quyidagicha bajariladi:

1. Sikl ichi qora ranga ajratiladi.
2. Standart vositalar panelidan kesib olish (Cut) tugmasi bosiladi.
3. Add Progrm Line (dasturga qator qo`shish) dasturlash paneli tugmasi bosiladi.

4.Qator kiritish joyiga kursor qo'yilib, standart vositalar panelidan qo'yish (Paste) tugmasi bosiladi.

5.Paydo bo'lgan kiritish joyi to'ldiriladi.

Bu usul barcha hollarda ham qator kiritishda qulaylikni beradi.

16.2.Dasturda qiymatlarni lokal yuborish.

Dasturda o'zgarmaslar va o'zgaruvchilarga qiymat berish (←) yuborish operatori yordamida amalga oshiriladi. Bu operator dasturlash panel vositasida (Local Definition) lokal aniqlash tugmasiga birlashtirilgan. Dastur tuzish davomida ko'p hollarda bu belgini klaviaturadan { belgisini bosish bilan ham bajarish mumkin.

Lokal o'zgaruvchi qiymatini dastur tashqarisida ishlatish mumkin emas. Agar tashqarida ishlatish juda kerak bo'lsa, uning uchun dasturning eng oxirgi operatoridan keyin kursorni bo'sh joyga qo'yib, keyin o'zgaruvchini yozish kerak bo'ladi.

Agar o'zgaruvchining unga mos bitta qiymatini chiqarish kerak bo'lsa, shu o'zgaruvchining nomini yozish kerak. Agar vektor yoki massivni chiqarish kerak bo'lsa uning nomini kiritish kerak.

if shartli operatori. if shartli operatori ikki bosqichda ta'sir etadi. Birinchi if operatoridan o'ngda yozilgan shart tekshiriladi. Agar u rost bo'lsa, undan chapdagi ifoda bajariladi, aks holda dasturning keyingi qatoriga o'tiladi.

Dasturda if shartli operatorini qo'yish uchun quyida keltirilgan protseduralarni bajaring.

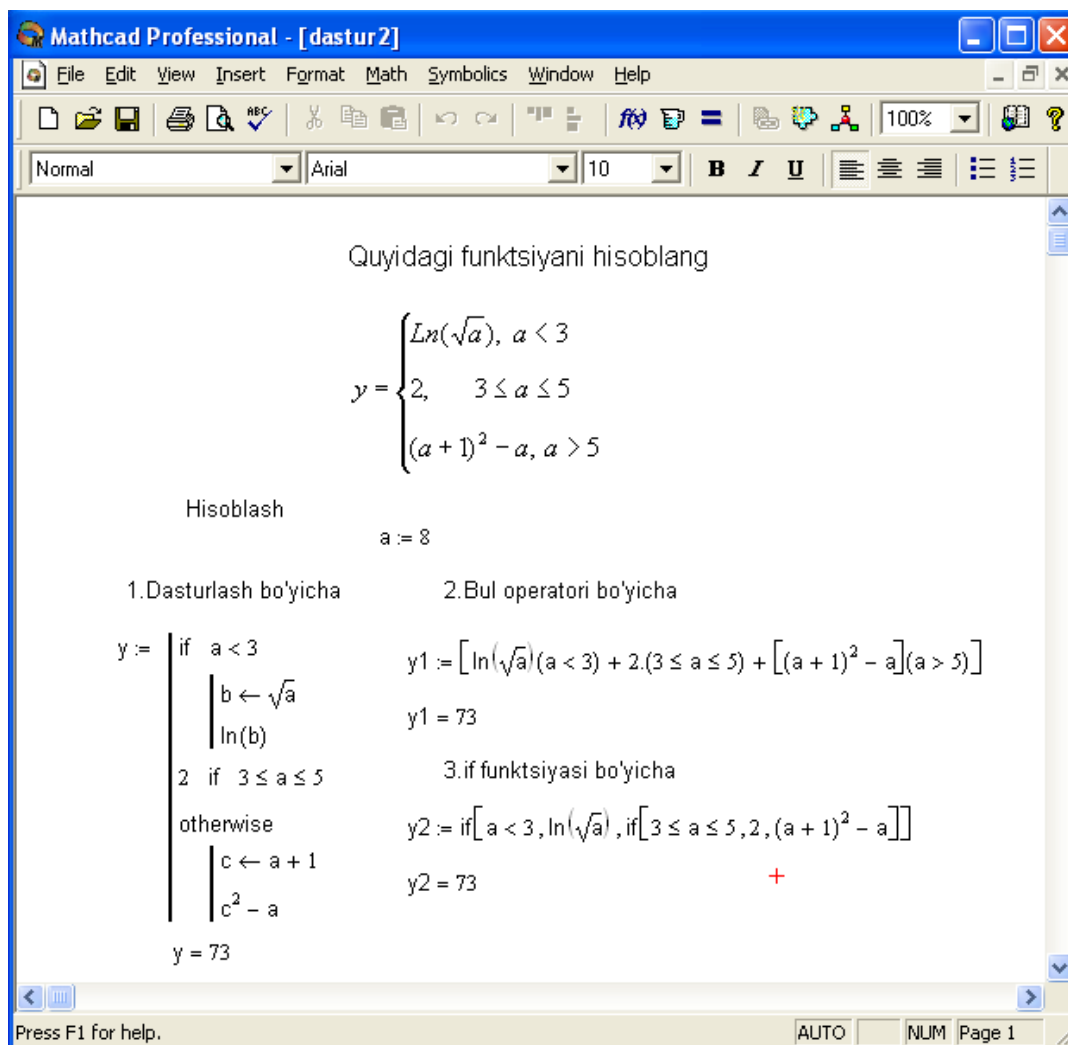
1.Tuziladigan dasturda shartli operator kiritiladigan joyga kursor qo'yiladi.

2.Dasturlash panelidan if operatori tugmasi bosiladi. Dasturda ikkita kiritishga ega operator shablani paydo bo'ladi.

3.O'ng kiritish joyiga shart kiritiladi. Bunda mantiqiy operatorlardan foydalanish mumkin. Buning uchun (Boolean) mantiqiy operatorlar panelidan foydalanish birmuncha qulayliklarni beradi.

4.If operatori chap tamoniga shart rost bo'lganda bajariladigan ifoda kiritiladi.

Agar shartning bajarilishida bir necha ifodalar bajariladigan bo'lsa, u holda bir necha kiritish joylariga ega bo'lish kerak. Buning uchun kursorni if operatorining chap tamondagi kiritish joyiga qo'yib, keyin dasturlash panelidagi Add Program Line (Dastur qatoriga qo'shish) tugmachasini necha qator kiritish kerak bo'lsa shuncha bosish kerak bo'ladi. Bunda shunga e'tibor berish kerakki, shartli operator ko'rinishi o'zgaradi. Yangi vertikal chiziq kiritish joyi bilan chap tamonda emas, pastda va if operatoridan o'ngda paydo bo'ladi. Agar shart yolg'on bo'lsa, o'tish dasturning keyingi qatoriga bo'ladi.



16.2-rasm. Shartli funktsiyani uch usulda hisoblash.

Mathcadda shartni yozishning uchta usuli bor:

- dasturlashning if shartli operatori yordamida;
- bul operatorlari yordamida;
- if funksiyasi yordamida.

rasmda shartni yozishning uchta usuli ko'rsatilgan.

16.3.MathCadda sikl opereratorlari.

Mathcadda ikkita sikl operatori mavjud: FOR va WHILE.

- Agar siklda takrorlanish soni oldindan ma'lum bo'lsa, u holda FOR operatori ishlatiladi.
- Agar sikl ma'lum shartning bajarilishi ichida takrorlanishi lozim bo'lsa, u holda WHILE operatori ishlatiladi.

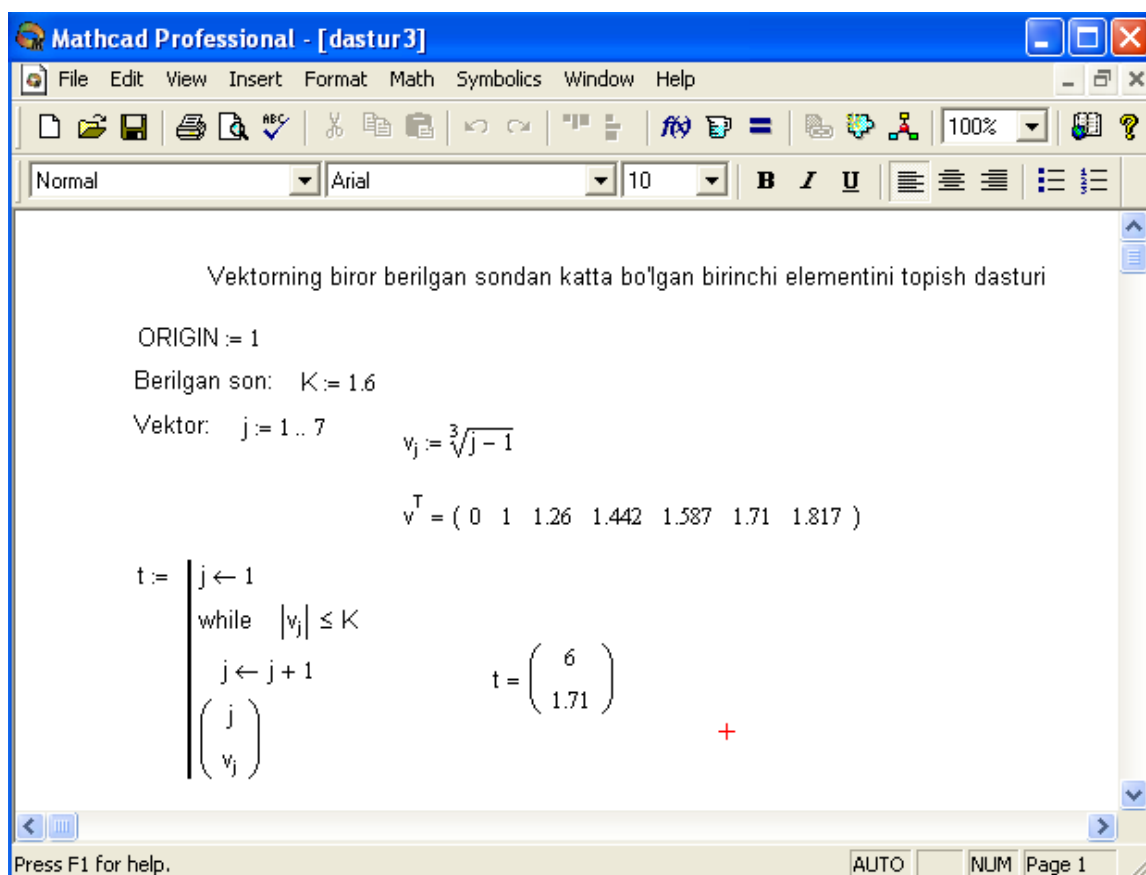
WHILE оператори.

While sikl operatori takrorlanishlar soni oldindan aniq bo'lmagan hollarda takrorlanishni biror bir shartning rost bo'lishida bajaradi. Berilgan shart oldin tekshirilib, keyin shartning bajarilishiga qarab uning tarkibidagi operatorlar bajariladi.

While sikl operatorini yozish uchun quyidagi ketma ketliklarni bajarish lozim:

1. Kursorni dastur kiritish kerak bo'lgan bo'sh joyga qo'yiladi.

2. Dasturlash panelidan While Loop (Tsikl While) tugmasi bosiladi.
3. While operatorining o'ng tamonidan shart (mantiqiy ifoda) kiritiladi.



16.3-rasm. Dasturlashda While sikl operatorini qo'llash.

4. While operatori pastidan sikl hisoblashi lozim bo'lgan ifodalar kiritiladi. Agar siklda bir necha ifodalarni hisoblash kerak bo'lsa, oldin kursorni kiritish joyiga qo'yib, keyin Add Program Line (Dasturga qator kiritish) yoki "J" (yopuvchi o'rta qavs) tugmasini sikl nechta qatorni o'z tarkibiga kiritisa shuncha marta bosish kerak bo'ladi. Keyin kiritish joylarini kerakli ifodalar bilan to'ldirib, ortiq kiritish joyi olib tashlanadi. Quyidagi rasmda misol tariqasida berilgan qiymatdan biron vektorning birinchi katta qiymatini aniqlash keltirilgan.

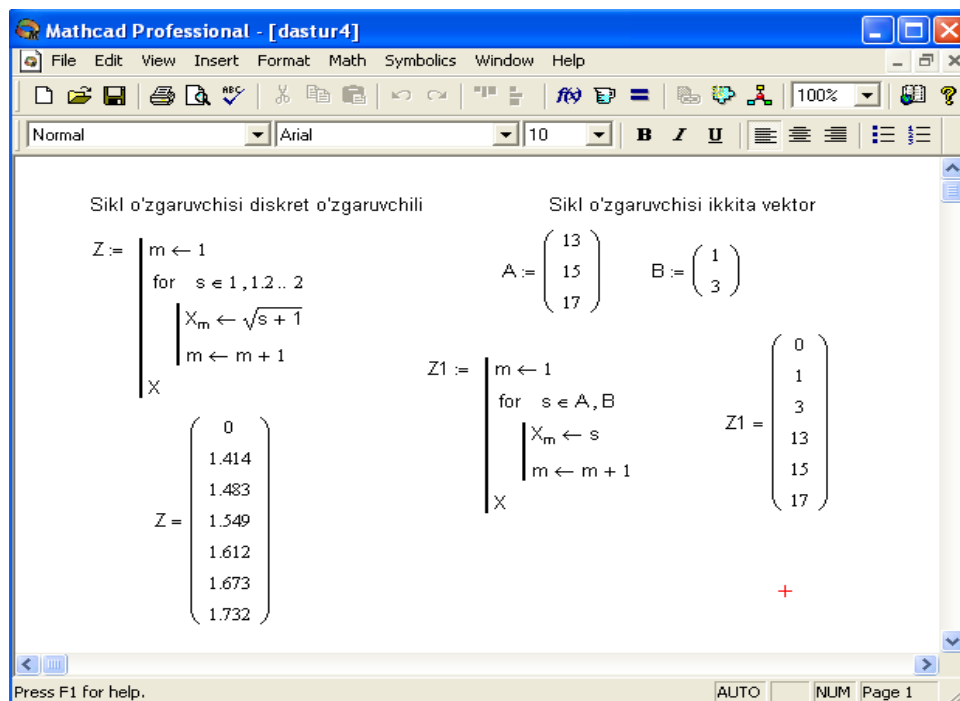
FOR operatori.

For sikl operatorini takrorlanishlar soni oldindan aniq bo'lganda ishlatish maqsadga muvofiqdir. For operatorining takrorlanishini, undan oldin berilgan o'zgaruvchi aniqlaydi.

For sikl operatorini yozish uchun quyidagi ketma ketliklarni bajarish lozim:

1. Kursorni dastur kiritish kerak bo'lgan bo'sh joyga qo'yiladi.
2. Dasturlash panelidan For Loop (Tsikl For) tugmasi bosiladi.
3. For operatorining o'ng tamonidan o'zgaruvchi nomi kiritilib, ungan keyin o'zgaruvchining o'zgarish diapazoni beriladi. Sikl o'zgaruvchisi sonlar qatori yoki vektor bo'lishi mumkin. Masalan rasmda o'zgaruvchi qiymatlari verul bilan ajratilgan vektor qilib berilgan.
4. For operatori pastidan sikl hisoblashi lozim bo'lgan ifodalar kiritiladi. Agar

siklda bir necha ifodalarni hisoblash kerak bo'lsa, oldin kursorni kiritish joyiga qo'yib, keyin Add Program Line (Dasturga qator kiritish) yoki "]" (yopuvchi o'rta qavs) tugmasini sikl nechta qatorni o'z tarkibiga kiritsa shuncha marta bosish kerak bo'ladi. Keyin kiritish joylarini kerakli ifodalar bilan to'ldirib, ortiq kiritish joyi olib tashlanadi. Quyidagi rasmda keltirilgan misolda berilgan qiymatdan biron vektorning birinchi katta qiymatini aniqlash berilgan.



16.4-rasm. Dasturlashda For sikl operatorini qo'llash.

17-ma'ruza. Mathcadda dasturlash.

Mavzu bo'yicha savollar to'plami:

- 1.Qanday dasturlash tillarini bilasiz?
- 2.Mathcad dasturini yuklash ketma-ketligini aytib bering.
- 3.Mathcad dasturida nechta menyu bor?
- 4.Mathcad dasturining 6 ta xarakteristikasini sanab bering.
- 5.Ikki o'lchamli funksiya grafigini qurish uchun qanday ishlarni bajarish kerak?
- 6.Grafikga animatsiya berish uchun qaysi tugmadan foydalaniladi.
- 7.Mathcadda bosmaga chiqarish uchun qaysi buyruq beriladi?

Mavzu bo'yicha glossariy:

LogScale - logarifmik masshtabda o'qga sonli qiymatlarni tasvirlash;
Trace - funksiya grafiklarini formatlash.
Label - grafik maydoni sarlavhasi.
Title (Sarlavha) maydoniga sarlovha matni kiritiladi.
Defaults - bu qo'yilma yordamida grafik ko'rinishga qaytish mumkin.
Calculator— asosiy matematik amallar shabloni; (Kalkulyator)
Graph— grafiklar shabloni; (Grafik)

Matrix (Massiv) – massiv va massiv amallarini bajarish shabloni; (Massiv)

Evaluation– qiymatlarni yuborish operatori va natijalarni chiqarish operatori; (Baholash)

Greek– Symbolik - belgililar ustida ishlash uchun operatorlar. (Grek alifbosi harflari)

Mavzuga oid adabiyotlar:

1. I.A.Bedarov, O.N.Belousova, N.N.Fedorova. ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ИНЖЕНЕРНЫХ ЗАДАЧ В ПАКЕТЕ MathCAD. Novosibirsk. 2005.

2. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.

3. Iqtisodiy Informatika, S.S.G'ulomov va boshqalar.Sharq, 1999

4. Internet va undan foydalanish asoslari. A.Maraximov, S.Raxmonqulova, Virtec, 2001

5.Q.M. Karimov, I.D. Razzoqov. MATHCAD va Matlab muhitida ishlash.

O'quv-uslubiy qo'llanma. Qarshi. "Nasaf" nashriyoti. 2014 y

Internet tarmog'idagi elektron manbalar:

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)

2. <http://moodle.edu.uz>

3. www.intuit.ru

4. <http://virlual-universily Eurasia.org>

5. <http://www.opennet.ru>

6. www.linux.org.ru

7. <https://nashol.com>

17.1.MathCadda dasturlash.

Dasturlash Mathcadda asosiy o`rin tutadi. Mathcad ko`plab masalalarni dastursiz echish imkoninii beradi. Lekin shunday sinf masalalari borki ularni dastursiz echib bo`lmaydi. Mathcad har qanday murakkab dasturni kiritish imkonini beradi. Mathcadda dasturlash juda aniq va tushunarli, unda dastur bir necha ketma-ket formulalarni ifodalaydi. Dasturlashning asosiy operatorlari Programming (Dasturlash) panelida joylashgan.

Dastur qatorini kiritish. Dasturni tuzish uchun uning qatorlarini kiritish kerak bo`ladi. Bu quyidagi keltirilgan protsedurada bajariladi:

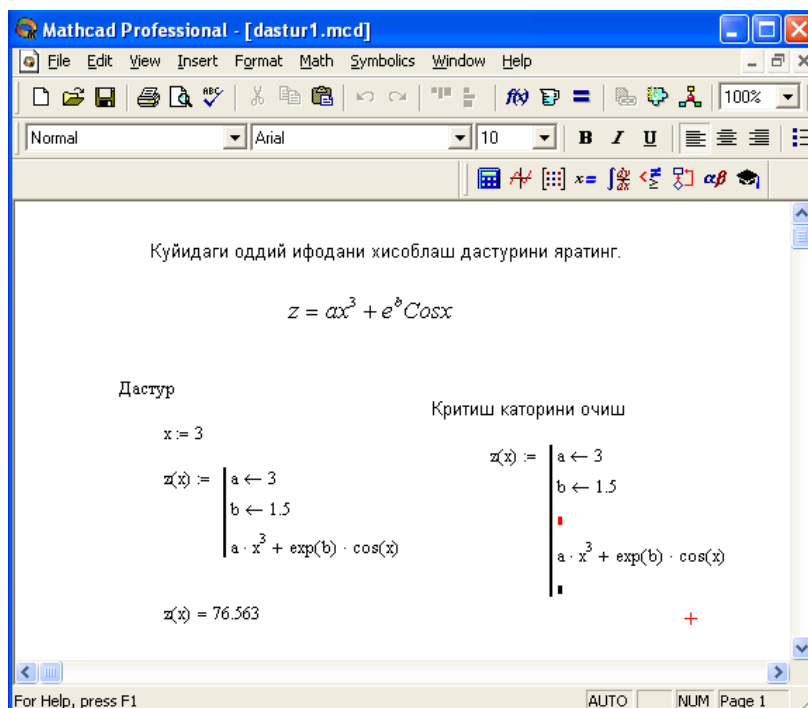
5. Dastur ifodasi nomini kiritish.

6. Yuborish operatorini (:=) kiritish.

7. Dasturlash panelidan Add Program Line (Dastur qatorini qo`shish) tugmasini bosish.

8. Paydo bo`lgan kiritish joyiga kerakli operatorlarni kiritish, ortiqcha kiritish joyini olib tashlash.

Kerakli kiritish qatorini ochish uchun ko`k burchakli kursorni qator oxiriga keltirib, bo`shlik tugmasini bosgan holda Add Program Line tugmasini bosish kerak. Agar kiritish qatorini qator oldidan ochish kerak bo`lsa ko`k burchakli kursorni qator boshiga keltirib, bo`shlik tugmasini bosgan holda Add Program Line tugmasini bosish kerak bo`ladi.



17.1-rasm. Oddiy chiziqli dasturlar tuzish.

Ayrim hollarda, masalan ikki ichma-ich joylashgan sikllar orasiga qator qo`shishda bu usul qo`l kelmay qoladi. Bu holda boshqa usulni qo`llashga to`g`ri keladi. Bu usul quyidagicha bajariladi:

1. Sikl ichi qora ranga ajratiladi.
2. Standart vositalar panelidan kesib olish (Cut) tugmasi bosiladi.
3. Add Progm Line (dasturga qator qo`shish) dasturlash paneli tugmasi bosiladi.
4. Qator kiritish joyiga kursor qo`yilib, standart vositalar panelidan qo`yish (Paste) tugmasi bosiladi.

5. Paydo bo`lgan kiritish joyi to`ldiriladi.

Bu usul barcha hollarda ham qator kiritishda qulaylikni beradi.

17.2. Dasturda qiymatlarni lokal yuborish.

Dasturda o`zgarmlar va o`zgaruvchilarga qiymat berish ($\leftarrow$) yuborish operatori yordamida amalga oshiriladi. Bu operator dasturlash panel vositasida (Local Definition) lokal aniqlash tugmasiga birlashtirilgan. Dastur tuzish davomida ko`p hollarda bu belgini klaviaturadan { belgisini bosish bilan ham bajarish mumkin.

Lokal o`zgaruvchi qiymatini dastur tashqarisida ishlatish mumkin emas. Agar tashqarida ishlatish juda kerak bo`lsa, uning uchun dasturning eng oxirgi operatoridan keyin kursorni bo`sh joyga qo`yib, keyin o`zgaruvchini yozish kerak bo`ladi.

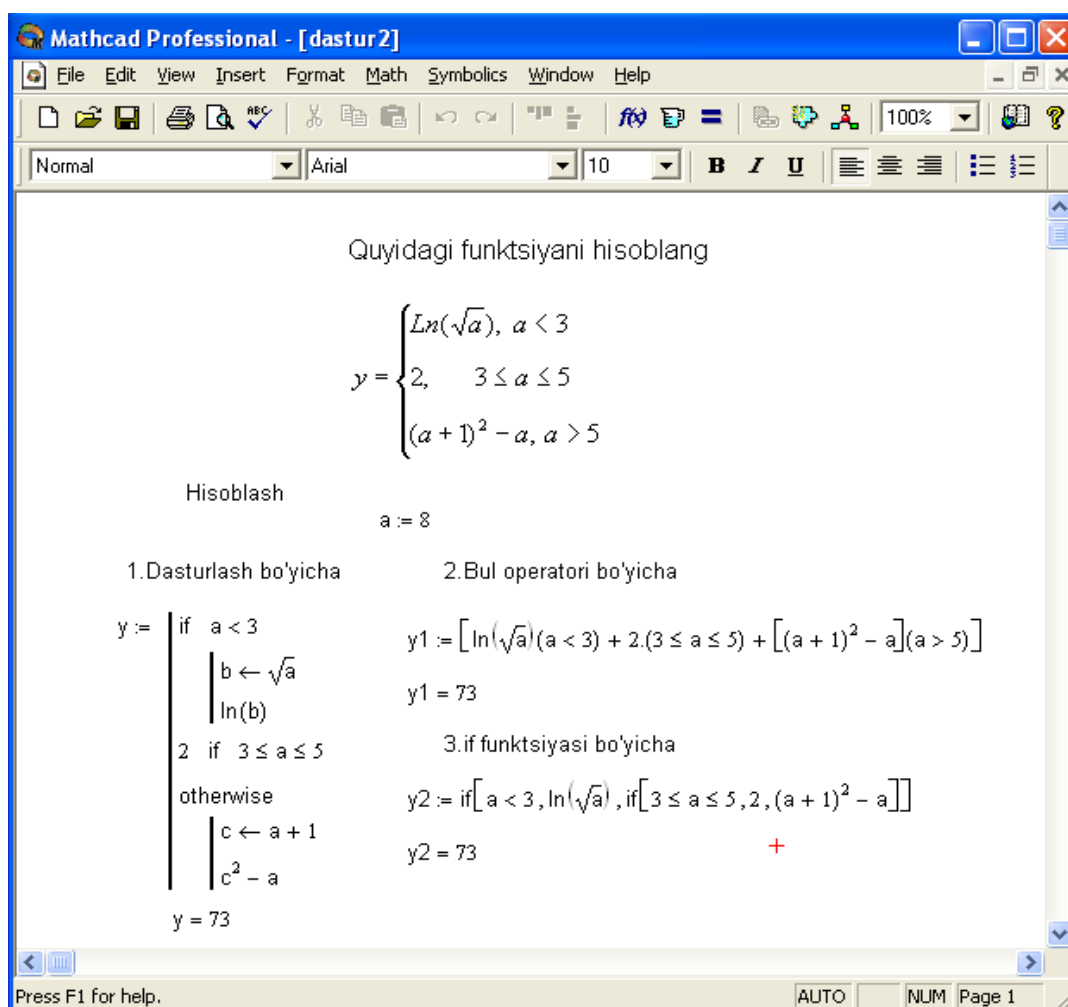
Agar o`zgaruvchining unga mos bitta qiymatini chiqarish kerak bo`lsa, shu o`zgaruvchining nomini yozish kerak. Agar vektor yoki massivni chiqarish kerak bo`lsa uning nomini kiritish kerak.

if shartli operatori. if shartli operatori ikki bosqichda ta'sir etadi. Birinchi if operatoridan o`ngda yozilgan shart tekshiriladi. Agar u rost bo`lsa, undan chapdagi ifoda bajariladi, aks holda dasturning keyingi qatoriga o`tiladi.

Dasturda if shartli operatorini qo'yish uchun quyida keltirilgan protseduralarni bajaring.

1. Tuziladigan dasturda shartli operator kiritiladigan joyga kursor qo'yiladi.
2. Dasturlash panelidan if operatori tugmasi bosiladi. Dasturda ikkita kiritishga ega operator shablani paydo bo'ladi.
3. O'ng kiritish joyiga shart kiritiladi. Bunda mantiqiy operatorlardan foydalanish mumkin. Buning uchun (Boolean) mantiqiy operatorlar panelidan foydalanish birmuncha qulayliklarni beradi.
4. If operatori chap tamoniga shart rost bo'lganda bajariladigan ifoda kiritiladi.

Agar shartning bajarilishida bir necha ifodalar bajariladigan bo'lsa, u holda bir necha kiritish joylariga ega bo'lish kerak. Buning uchun kursorni if operatorining chap tamondagi kiritish joyiga qo'yib, keyin dasturlash panelidagi Add Program Line (Dastur qatoriga qo'shish) tugmachasini necha qator kiritish kerak bo'lsa shuncha bosish kerak bo'ladi. Bunda shunga e'tibor berish kerakki, shartli operator ko'rinishi o'zgaradi. Yangi vertikal chiziq kiritish joyi bilan chap tamonda emas, pastda va if operatoridan o'ngda paydo bo'ladi. Agar shart yolg'on bo'lsa, o'tish dasturning keyingi qatoriga bo'ladi.



17.2-rasm. Shartli funktsiyani uch usulda hisoblash.

Mathcadda shartni yozishning uchta usuli bor:

- dasturlashning if shartli operatori yordamida;
- bul operatorlari yordamida;
- if funksiyasi yordamida.

rasmda shartni yozishning uchta usuli ko`rsatilgan.

17.3.MathCadda sikl opreratorlari.

Mathcadda ikkita sikl operatori mavjud: FOR va WHILE.

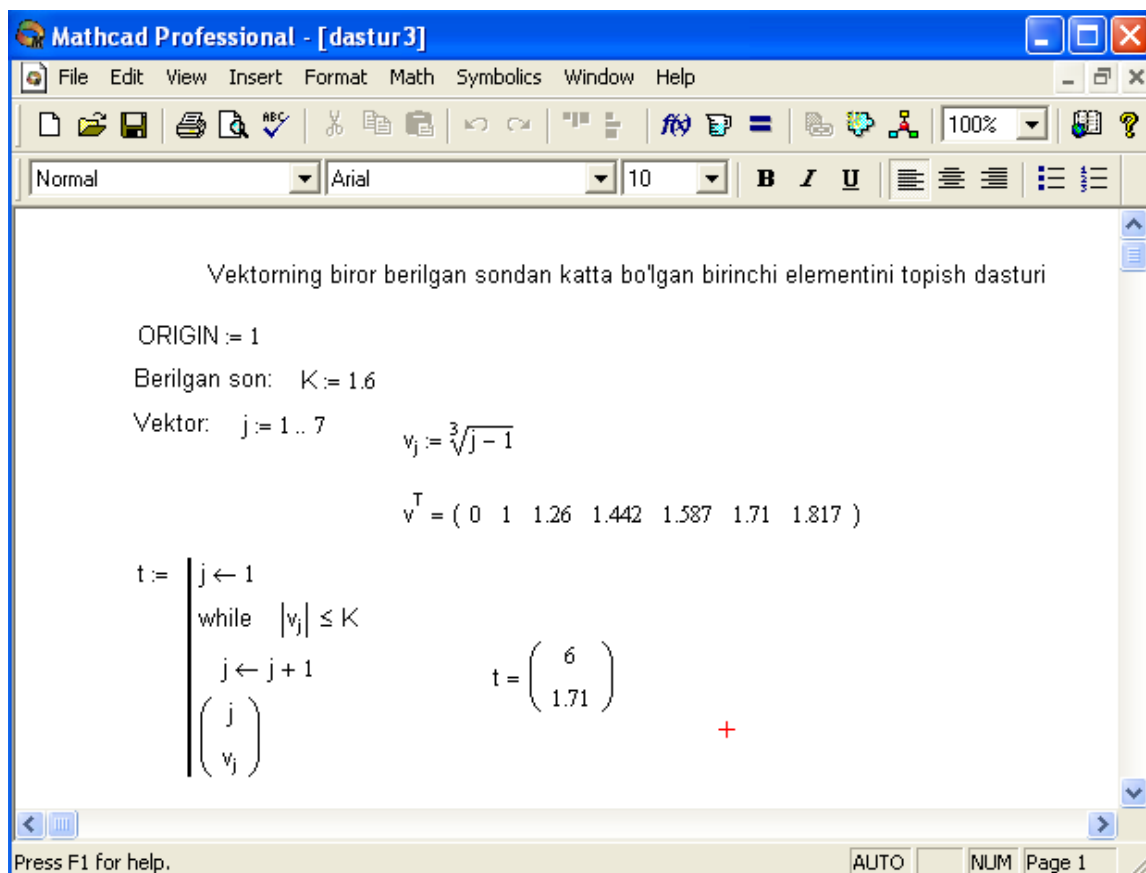
- Agar siklda takrorlanish soni oldindan ma'lum bo`lsa, u holda FOR operatori ishlatiladi.
- Agar sikl ma'lum shartning bajarilishi ichida takrorlanishi lozim bo`lsa, u holda WHILE operatori ishlatiladi.

WHILE оператори.

While sikl operatori takrorlanishlar soni oldindan aniq bo`lmagan hollarda takrorlanishni biror bir shartning rost bo`lishida bajaradi. Berilgan shart oldin tekshirilib, keyin shartning bajarilishiga qarab uning tarkibidagi operatorlar bajariladi.

While sikl operatorini yozish uchun quyidagi ketma ketliklarni bajarish lozim:

- 1.Kursorni dastur kiritish kerak bo`lgan bo`sh joyga qo`yiladi.
- 2.Dasturlash panelidan While Loop (Tsikl While) tugmasi bosiladi.
- 3.While operatorining o`ng tamonidan shart (mantiqiy ifoda) kiritiladi.



17.3-rasm. Dasturlashda While sikl operatorini qo`llash.

- 4.While operatori pastidan sikl hisoblashi lozim bo`lgan ifodalar kiritiladi. Agar

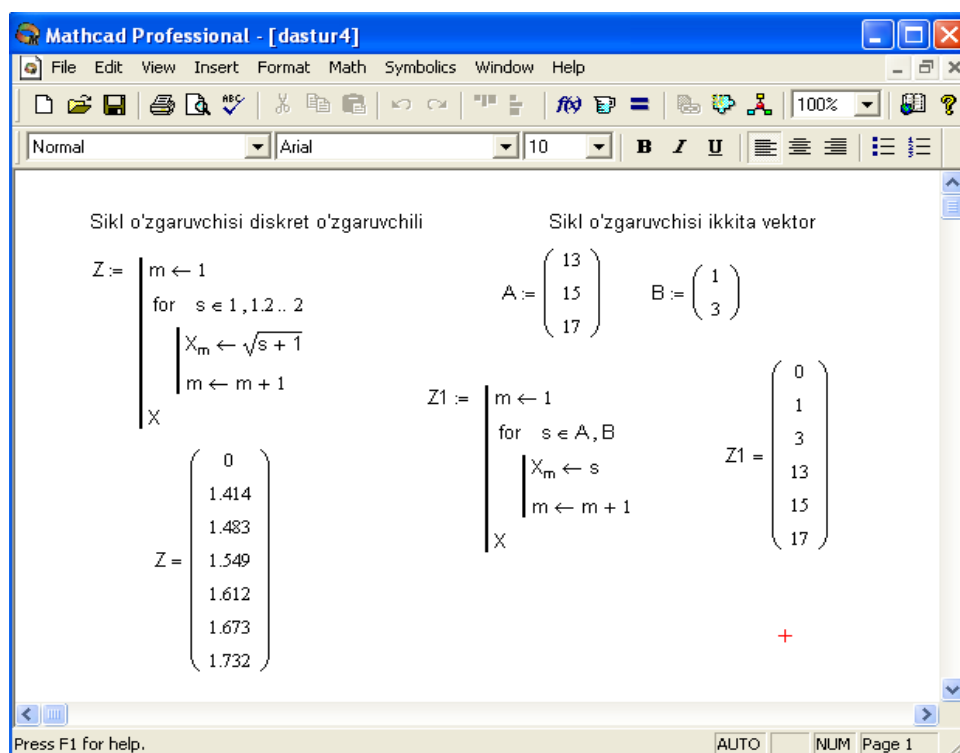
siklda bir necha ifodalarni hisoblash kerak bo'lsa, oldin kursorni kiritish joyiga qo'yib, keyin Add Program Line (Dasturga qator kiritish) yoki "]" (yopuvchi o'rta qavs) tugmasini sikl nechta qatorni o'z tarkibiga kiritsa shuncha marta bosish kerak bo'ladi. Keyin kiritish joylarini kerakli ifodalar bilan to'ldirib, ortiq kiritish joyi olib tashlanadi. Quyidagi rasmda misol tariqasida berilgan qiymatdan biron vektorning birinchi katta qiymatini aniqlash keltirilgan.

FOR operatori.

For sikl operatorini takrorlanishlar soni oldindan aniq bo'lganda ishlatish maqsadga muvofiqdir. For operatorining takrorlanishini, undan oldin berilgan o'zgaruvchi aniqlaydi.

For sikl operatorini yozish uchun quyidagi ketma ketliklarni bajarish lozim:

1. Kursorni dastur kiritish kerak bo'lgan bo'sh joyga qo'yiladi.
2. Dasturlash panelidan For Loop (Tsikl For) tugmasi bosiladi.
3. For operatorining o'ng tamonidan o'zgaruvchi nomi kiritilib, ungan keyin o'zgaruvchining o'zgarish diapazoni beriladi. Sikl o'zgaruvchisi sonlar qatori yoki vektor bo'lishi mumkin. Masalan rasmda o'zgaruvchi qiymatlari verul bilan ajratilgan vektor qilib berilgan.
4. For operatori pastidan sikl hisoblashi lozim bo'lgan ifodalar kiritiladi. Agar siklda bir necha ifodalarni hisoblash kerak bo'lsa, oldin kursorni kiritish joyiga qo'yib, keyin Add Program Line (Dasturga qator kiritish) yoki "]" (yopuvchi o'rta qavs) tugmasini sikl nechta qatorni o'z tarkibiga kiritsa shuncha marta bosish kerak bo'ladi. Keyin kiritish joylarini kerakli ifodalar bilan to'ldirib, ortiq kiritish joyi olib tashlanadi. Quyidagi rasmda keltirilgan misolda berilgan qiymatdan biron vektorning birinchi katta qiymatini aniqlash berilgan.



17.4-rasm. Dasturlashda For sikl operatorini qo'llash.

18-ma'ruza. Mathcadning boshqa dasturiy ta'minotlar bilan o'zaro bog'liqligi.

Mavzu bo'yicha savollar to'plami:

1. Rastrli tasvirlar qanday tasvirlar?
2. READFILE funksiyasidan qanday foydalaniladi?
3. Mathcad dasturida nechta menyu bor?
4. Mathcad dasturining 6 ta xarakteristikasini sanab bering.
5. Ikki o'lchamli funksiya grafigini qurish uchun qanday ishlarni bajarish kerak?
6. Grafikga animatsiya berish uchun qaysi tugmadan foydalaniladi.
7. Mathcadda bosmaga chiqarish uchun qaysi buyruq beriladi?

Mavzu bo'yicha glossariy:

LogScale - logarifmik masshtabda o'qga sonli qiymatlarni tasvirlash;
Trace - funksiya grafiklarini formatlash.
Label - grafik maydoni sarlavhasi.
Title (Sarlavha) maydoniga sarlovha matni kiritiladi.
Defaults - bu qo'yilma yordamida grafik ko'rinishga qaytish mumkin.
Calculator – asosiy matematik amallar shabloni; (Kalkulyator)
Graph – grafiklar shabloni; (Grafik)
Matrix (Massiv) – massiv va massiv amallarini bajarish shabloni; (Massiv)
Evaluation – qiymatlarni yuborish operatori va natijalarni chiqarish operatori; (Baholash)
Greek– Symbolik - belgililar ustida ishlash uchun operatorlar. (Grek alifbosi harflari)

18.1.MathCad va Microsoft Word.

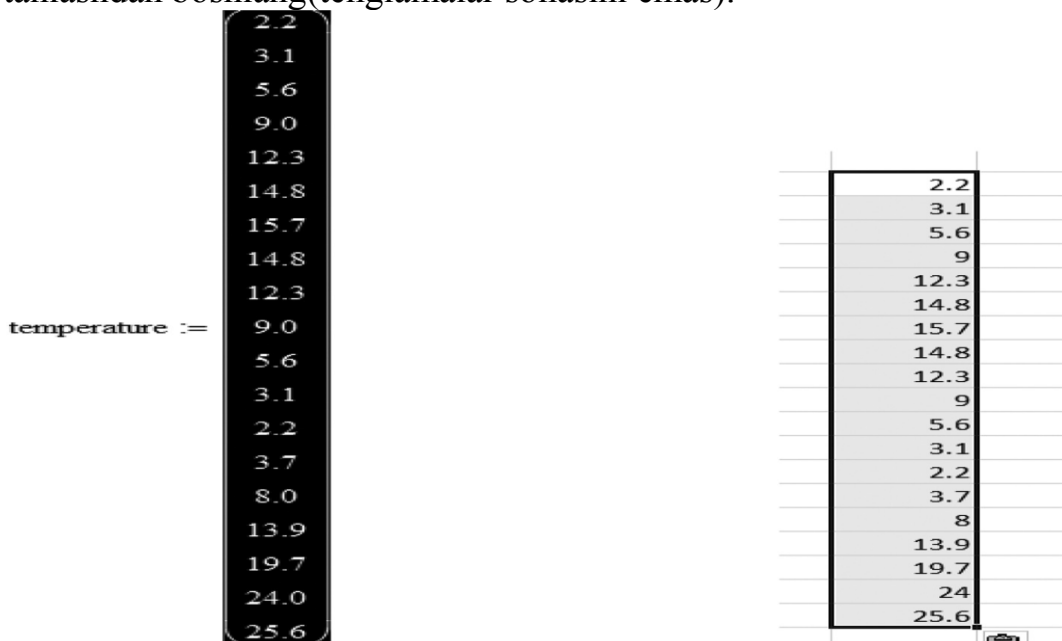
MathCad dan biror ma'lumotni Microsoft Word ga ko'chirish uchun matematik va izohli sohadan tashqarida sichqonchani bosib turib kerakli joyga sudrang. Keyin nusxa olingan soxani buferga joylashtirish uchun Edit/Copy tanglang yoki (Ctrl+C).

Microsoft Word da hujjatni boshqaring va buning uchun Paste special > Pasting (Spetsialnaya vstavka > Vstavit) kuchli matn sohasi yoki (Ctrl+V). Bu Mathcad dan Word ga soha nusxasini olishning eng sodda usuli.

Windows uchu qo'shimcha variant Metafile hisoblanib, yirik operatorlar va MathCad grafiklarini saqlay oladi va Kengaytirilgan Metafile esa masshtablashtirilgan, masshtablashtirilayotgan grafik tasvirlarni yaratadi, lekin noto'g'ri akslantiradi. Shuningdek Word da MathCaddagi hujjat obyektini yaratish mumkin, bu MathCad sohasini yaratishning imkoniyatlarini oshiradi. Foydalanuvchi Wordga rastr ko'rinishda obyekt va tenglamalarni o'rnatgandan so'ng ularni o'zgartirish imkoniyatiga ega bo'ladi. Rastrli tasvir bu tasvir fayllari bo'lib, odatda tasvirlarni taxrlash uchun juda katta fayllardan iborat. Buni boshqa variantlar ish bermaganda ishlatish mumkin.

18.2.MathCad va Microsoft Excel.

Berilganlarni Exceldan Mathcadga o'tkazishning bir necha turlari mavjud. Ma'lumotlarni Exceldan o'tkazishdan avval xar doim Excelda faylni saqlang, chunki bu joriy o'tkazilayotgan ma'lumotlarni kafolati bo'ladi. Usullar orasidagi farq: yoki bir marotabalik o'tkazish yoki ma'lumotlar avtomatik holda tiklanadi. Shunga qaramay berilganlarni shu usulda o'tkazish avtomatik holda tiklanmaydi. Mathcaddan Excelga o'tkazish uchun massiv yoki matritsni tanlashdan boshlang (tenglamalar sohasini emas).

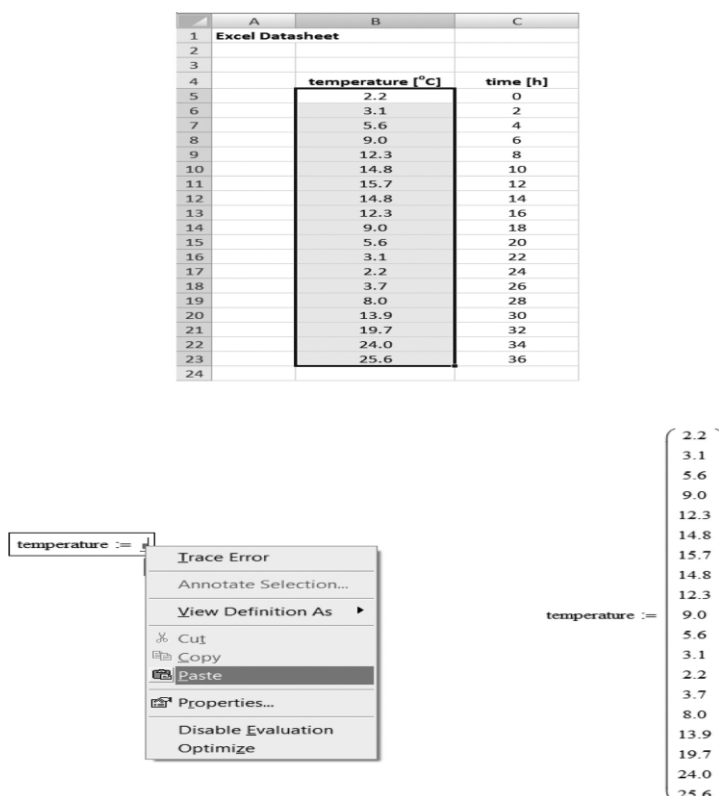


18.1-rasm. Kopirovat/vstavlyat rezultati iz Mathcad i Excel.

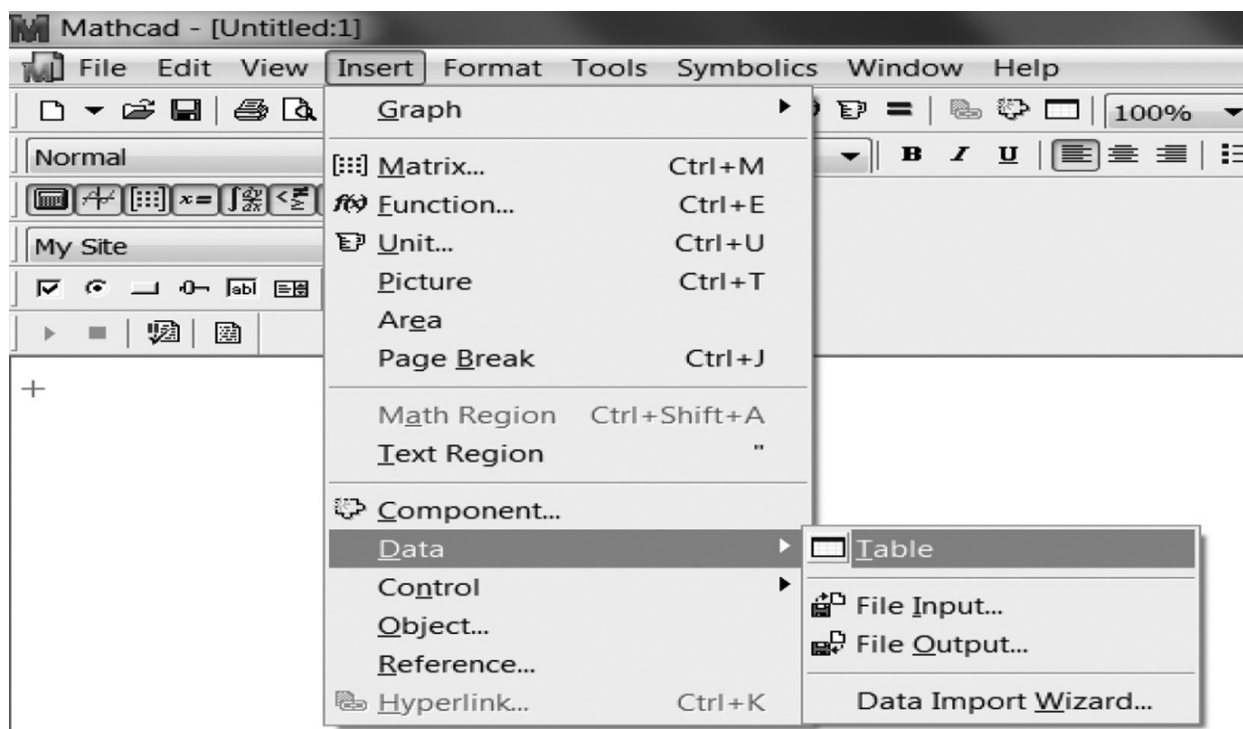
Agar bir marotabalik o'tkazish kerak bo'lsa avtorlar sodda nusxa olish-o'rnatish usulini taklif etadi. Shundan so'ng (Ctrl + C) orqali nusxa oling va (Ctrl + V) orqali kerakli Excel yacheykasiga o'rning. Xuddi shuningdek Exceldan MathCadga o'tkazish uchun MathCadda o'zgaruchi yarating va Exceldagi ma'lumotlarni o'zgaruvchining to'ldirguvchisiga (markeriga) o'rning. Boshqa varianti Data Menu (Vstavka dannix) menyusidan foydalanish. Bu yerda uchta variant keltirilgan: berilganlar jadvali, kiruvchi fayllar va berilganlarni import ustasi. Berilganlar jadvali va bir marotabali ma'lumotlarni Excel dan MathCadga o'tkazish. Jadvaldagi berilganlar avtomatik ravishda tiklanmaydi va shuning uchun qo'l orqali ma'lumotlar kiritilishi kerak. Agar jadval o'rnatilayotgan bo'lsa, u holda o'zgaruvchining nomi uchun joy ajratilgan. Odatda 2×2 ko'rinishdagi jadvalni sudrash orqali o'tkazilganda (kursorni chetida qora kvadrat belgisi bo'ladi) jadvalda o'zgarishlar sodir bo'lishi mumkin. Jadval yacheykalarga ma'lumot kiritish orqali yoki Excel dagi nusxa olish komandasi (yoki Ctrl+C) orqali nusxa olinadi va Vstavka menyusi (yoki Ctrl+V) orqali MathCadga o'rnatish mumkin.

Excel dan berilganlarni impot qilishning boshqa varianti. Birinchidan yacheykani tanlash uchun jadvalning ixtiyoriy yacheykasinin bosing, sichqonchaning o'ng tamonini bosing, ko'rsatilgan kontekst menyu ochiladi. Fayllarning kiritish komponenti fayllarnin o'qish imkoniyatiga ega. Yana, xuddi

shuningdek, kiritilgan bir marotaba ma'lumotlarni xech qachon avtomatik xolda tiklanmaydi. Fayllarni kiritish komponentini o'rnatish uchun, **Insert/Data/File Input** (Vstavka / Peredacha dannix / vxodnogo fayla) ketma- ketligini bajaring.



18.2-rasm. Exceldan MathCad muxitiga nusxa olish/qo'yish.



18.3-rasm. Berilganlar jadvalini yaratish.

I :=		
	0	1
0	0	
1		

D :=		
	0	1
0	0	
1		
2		
3		

18.4-rasm. Jadvalni berilganlar bilan to'ldirish.

Berilganlarning import ustasi kiritilayotgan berilganlardan farqi shundaki u foydalanuvchiga Excel fayllarini ko'rishga va kerakli qiymatni tanlashga imkoniyat beradi. Agar Excelda qiymat tiklansa, MathCad muxitida qayta hisoblangandan so'ng tiklanadi.

READFILE fuksiyasining paramerlari ham berilganlarning import ustasi kabi imkoniyat beradi , lekin muloqat oynasiz.

Mathcad muxiti varaqani qayta ko'rib chiqqach har doim READFILE funksiyasi bajariladi, shuning uchun u Excel fayllarini har vaqtda tiklaydi.

READFILE (filePath, "Excel") yoki READEXCEL (filePath)

Bu parametr Excel faylidagi xamma berilganlarni kema-ket xisoblaydi. Ayrim xollarda berilganlarni tozalashni so'raydi.

READFILE (filePath, "Excel", startRow, startCol) yoki

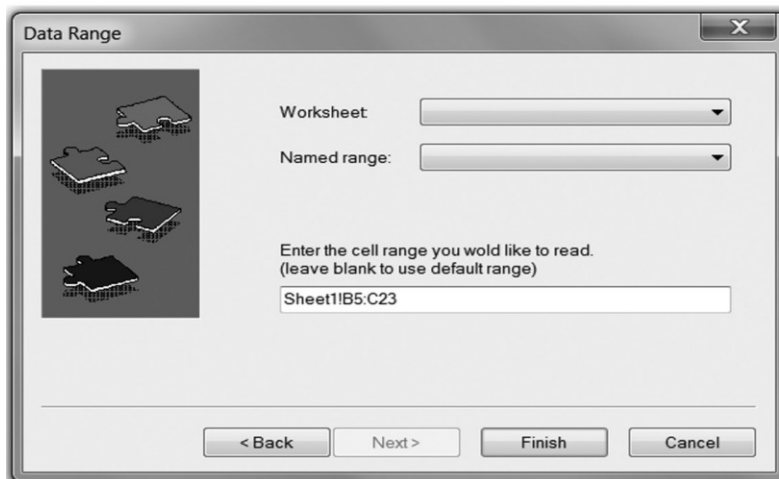
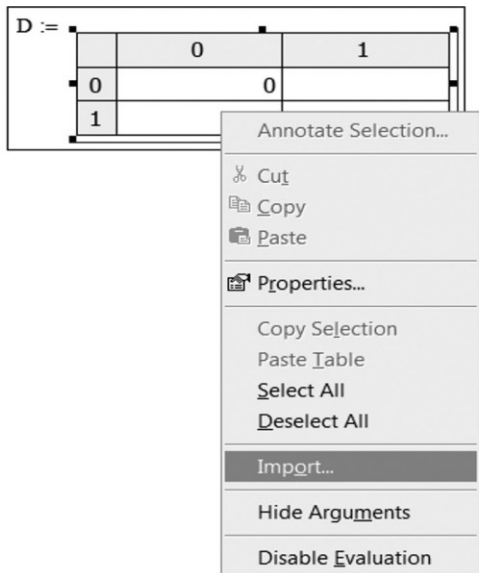
READEXCEL (filePath, "B2:B4")

MathCad startRow va start Collardan boshlab ma'lumotlani uzliksiz o'qiydi.

IB:READEXCEL(excelfile_path,"B5:C10")

$$IB := \text{READEXCEL}(\text{excelfile_path}, "B5:C10")$$

$$IB = \begin{pmatrix} 2.234 & 0 \\ 3.134 & 2 \\ 5.593 & 4 \\ 8.952 & 6 \\ 12.311 & 8 \\ 14.771 & 10 \end{pmatrix}$$



D :=

	0	1
0	2.234	0
1	3.134	2
2	5.593	4
3	8.952	6
4	12.311	8
5	14.771	10
6	15.671	12
7	14.771	14
8	12.311	16
9	8.952	...

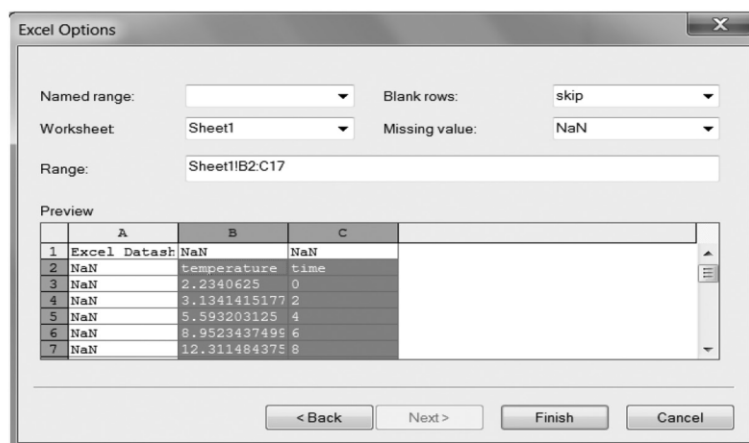
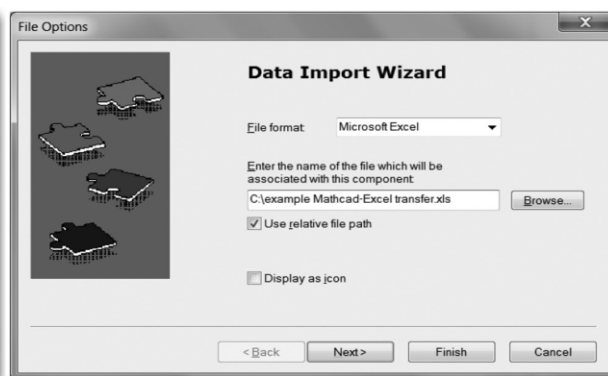
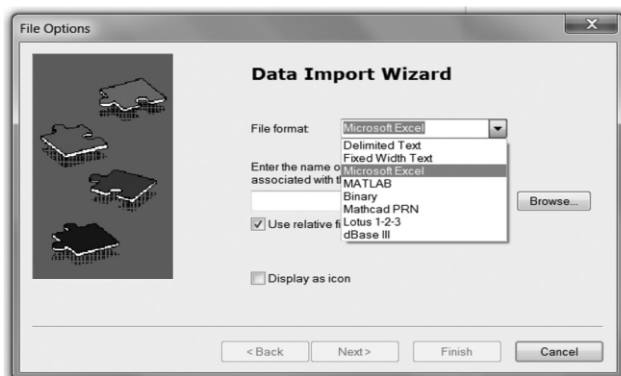
18.5-rasm. Import Excel dan MathCad muxitiga.

D :=



C:\Us...\example Mathcad-Excel transfer.xlsx

Import Excel dan MathCad muxitiga.



	0	1
0	"temperature"	"time"
1	2.234	0
2	3.134	2
3	5.593	4
4	8.952	6
5	12.311	8
6	14.771	10
7	15.671	12
8	14.771	14
9	12.311	16
10	8.952	18
11	5.593	20
12	3.134	22
13	2.154	...

18.6-rasm.Import Exceldan *import ustasi orqali* MathCad muxitiga.

```

filePath := "example Mathcad-Excel transfer.xlsx"
excelfile_path := concat(CWD, filePath)

D := READFILE(excelfile_path, "Excel")

```

D =

	1	2	3
1	"Excel Datasheet"	NaN	NaN
2	NaN	NaN	NaN
3	NaN	NaN	NaN
4	NaN	"temperature"	"time"
5	NaN	2.234	0
6	NaN	3.134	2
7	NaN	5.593	4
8	NaN	8.952	6
9	NaN	12.311	8
10	NaN	14.771	10
11	NaN	15.671	12
12	NaN	14.771	14
13	NaN	12.311	16
14	NaN	8.952	...

18.6-rasm. READFILE funksiyasidan qanday foydalanishga misol.
 READFILE (filePath, "Excel", vRow, vCol)

vRow va vCol vektorlar satirlarni ishga tushirish va to'xtatishni ko'rsatadi.

$$\begin{aligned}
 &vRows := \begin{pmatrix} 5 \\ 10 \end{pmatrix} \quad vCols := \begin{pmatrix} 2 \\ 3 \end{pmatrix} \\
 &AF := READFILE(excelfile_path, "Excel", vRows, vCols) \\
 &AF = \begin{pmatrix} 2.234 & 0 \\ 3.134 & 2 \\ 5.593 & 4 \\ 8.952 & 6 \\ 12.311 & 8 \\ 14.771 & 10 \end{pmatrix}
 \end{aligned}$$

Shuningd
 imkoniyatiga eg
 ishlatmoqchi bo

$$vL = \begin{pmatrix} 14.771 & 10 \\ 12.311 & 8 \\ 8.952 & 6 \end{pmatrix}$$

r kengaytmasin tanlash
 arini MathCad muxitida

19-ma'ruza. Mathcadning boshqa dasturiy ta'minotlar bilan o'zaro bog'liqligi.

Mavzu bo'yicha savollar to'plami:

1. Rastrli tasvirlar qanday tasvirlar?
2. READFILE funksiyasidan qanday foydalaniladi?
3. Mathcad dasturida nechta menyu bor?
4. Mathcad dasturining 6 ta xarakteristikasini sanab bering.
5. Ikki o'lchamli funksiya grafigini qurish uchun qanday ishlarni bajarish kerak?
6. Grafikga animatsiya berish uchun qaysi tugmadan foydalaniladi.
7. Mathcadda bosmaga chiqarish uchun qaysi buyruq beriladi?

Mavzu bo'yicha glossariy:

LogScale - logarifmik masshtabda o'qga sonli qiymatlarni tasvirlash;
Trace - funksiya grafiklarini formatlash.
Label - grafik maydoni sarlavhasi.
Title (Sarlavha) maydoniga sarlovha matni kiritiladi.
Defaults - bu qo'yilma yordamida grafik ko'rinishga qaytish mumkin.
Calculator – asosiy matematik amallar shabloni; (Kalkulyator)
Graph – grafiklar shabloni; (Grafik)
Matrix (Massiv) – massiv va massiv amallarini bajarish shabloni; (Massiv)
Evaluation – qiymatlarni yuborish operatori va natijalarni chiqarish operatori; (Baholash)
Greek– Symbolik - belgililar ustida ishlash uchun operatorlar. (Grek alifbosi harflari)

19.1.MathCad va Microsoft Word.

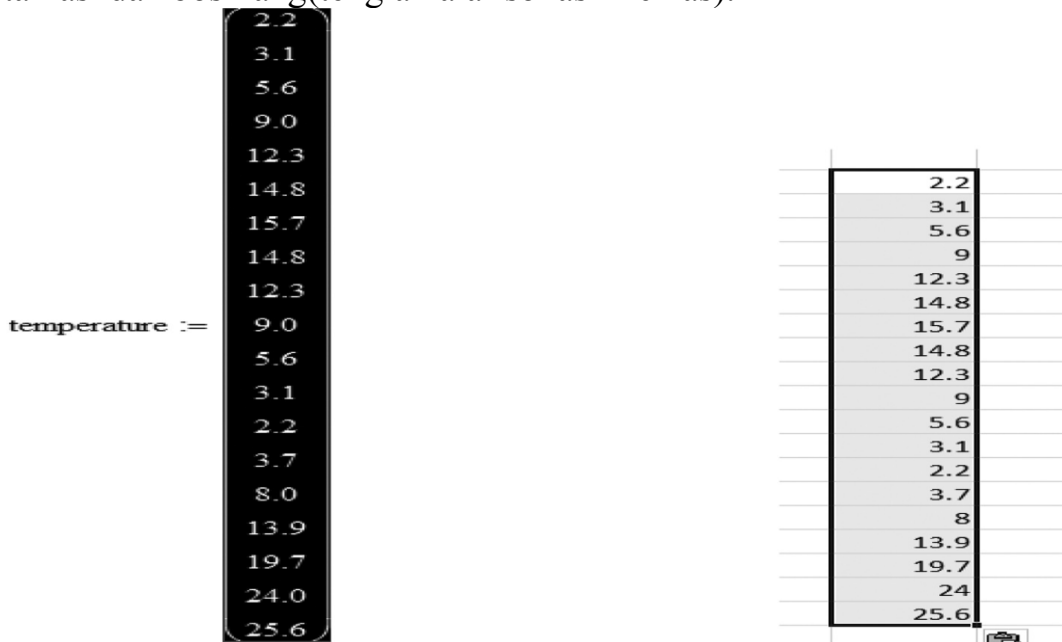
MathCad dan biror ma'lumotni Microsoft Word ga ko'chirish uchun matematik va izohli sohadan tashqarida sichqonchani bosib turib kerakli joyga sudrang. Keyin nusxa olingan soxani buferga joylashtirish uchun Edit/Copy tanglang yoki (Ctrl+C).

Microsoft Word da hujjatni boshqaring va buning uchun Paste special > Pasting (Spetsialnaya vstavka > Vstavit) kuchli matn sohasi yoki (Ctrl+V). Bu Mathcad dan Word ga soha nusxasini olishning eng sodda usuli.

Windows uchu qo'shimcha variant Metafile hisoblanib, yirik operatorlar va MathCad grafiklarini saqlay oladi va Kengaytirilgan Metafile esa masshtablashtirilgan, masshtablashtirilayotgan grafik tasvirlarni yaratadi, lekin noto'g'ri akslantiradi. Shuningdek Word da MathCaddagi hujjat obyektini yaratish mumkin, bu MathCad sohasini yaratishning imkoniyatlarini oshiradi. Foydalanuvchi Wordga rastr ko'rinishda obyekt va tenglamalarni o'rnatgandan so'ng ularni o'zgartirish imkoniyatiga ega bo'ladi. Rastrli tasvir bu tasvir fayllari bo'lib, odatda tasvirlarni taxrlash uchun juda katta fayllardan iborat. Buni boshqa variantlar ish bermaganda ishlatish mumkin.

19.2.MathCad va Microsoft Excel.

Berilganlarni Exceldan Mathcadga o'tkazishning bir necha turlari mavjud. Ma'lumotlarni Exceldan o'tkazishdan avval xar doim Excelda faylni saqlang, chunki bu joriy o'tkazilayotgan ma'lumotlarni kafolati bo'ladi. Usullar orasidagi farq: yoki bir marotabalik o'tkazish yoki ma'lumotlar avtomatik holda tiklanadi. Shunga qaramay berilganlarni shu usulda o'tkazish avtomatik holda tiklanmaydi. Mathcaddan Excelga o'tkazish uchun massiv yoki matritsni tanlashdan boshlang (tenglamalar sohasini emas).

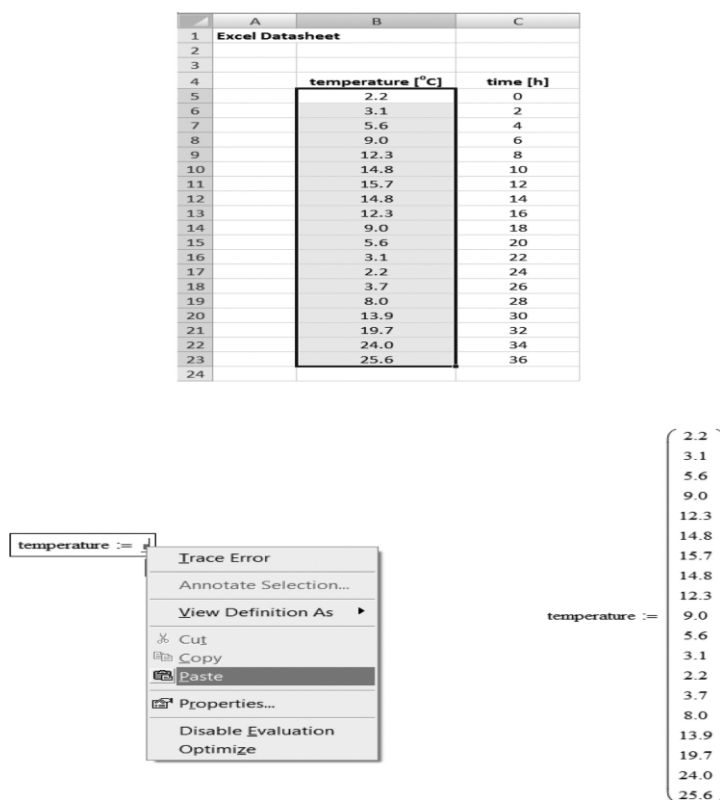


19.1-rasm. Kopirovat/vstavlyat rezultati iz Mathcad i Excel.

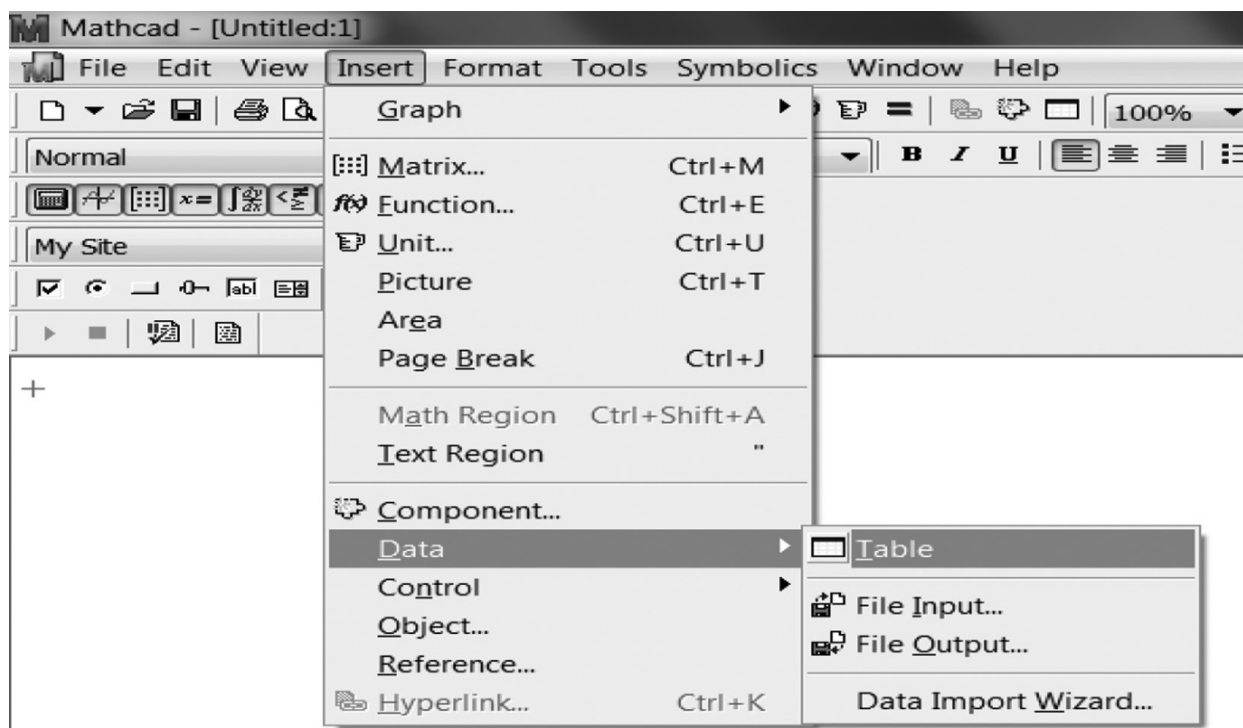
Agar bir marotabalik o'tkazish kerak bo'lsa avtorlar sodda nusxa olish-o'rnatish usulini taklif etadi. Shundan so'ng (Ctrl + C) orqali nusxa oling va (Ctrl + V) orqali kerakli Excel yacheykasiga o'rning. Xuddi shuningdek Exceldan MathCadga o'tkazish uchun MathCadda o'zgaruchi yarating va Exceldagi ma'lumotlarni o'zgaruvchining to'ldiruvchisiga (markeriga) o'rning. Boshqa varianti Data Menu (Vstavka dannix) menyusidan foydalanish. Bu yerda uchta variant keltirilgan: berilganlar jadvali, kiruvchi fayllar va berilganlarni import ustasi. Berilganlar jadvali va bir marotabali ma'lumotlarni Excel dan MathCadga o'tkazish. Jadvaldagi berilganlar avtomatik ravishda tiklanmaydi va shuning uchun qo'l orqali ma'lumotlar kiritilishi kerak. Agar jadval o'rnatilayotgan bo'lsa, u holda o'zgaruvchining nomi uchun joy ajratilgan. Odatda 2×2 ko'rinishdagi jadvalni sudrash orqali o'tkazilganda (kursorni chetida qora kvadrat belgisi bo'ladi) jadvalda o'zgarishlar sodir bo'lishi mumkin. Jadval yacheykalarga ma'lumot kiritish orqali yoki Excel dagi nusxa olish komandasi (yoki Ctrl+C) orqali nusxa olinadi va Vstavka menyusi (yoki Ctrl+V) orqali MathCadga o'rnatish mumkin.

Excel dan berilganlarni impot qilishning boshqa varianti. Birinchidan yacheykani tanlash uchun jadvalning ixtiyoriy yacheykasinin bosing, sichqonchaning o'ng tamonini bosing, ko'rsatilgan kontekst menyu ochiladi.

Fayllarning kiritish komponenti fayllarnin o‘qish imkoniyatiga ega. Yana, xuddi shuningdek, kiritilgan bir marotaba ma’lumotlarni xech qachon avtomatik xolda tiklanmaydi. Fayllarni kiritish komponentini o‘rnatish uchun, **Insert/Data/File Input** (Vstavka / Peredacha dannix / vxodnogo fayla) ketma- ketligini bajaring.



19.2-rasm. Exceldan MathCad muxitiga nusxa olish/qo‘yish.



19.3-rasm. Berilganlar jadvalini yaratish.

B :=		
	0	1
0	0	
1		

D :=		
	0	1
0	0	
1		
2		
3		

19.4-rasm. Jadvalni berilganlar bilan to'ldirish.

Berilganlarning import ustasi kiritilayotgan berilganlardan farqi shundaki u foydalanuvchiga Excel fayllarini ko'rishga va kerakli qiymatni tanlashga imkoniyat beradi. Agar Excelda qiymat tiklansa, MathCad muxitida qayta hisoblangandan so'ng tiklanadi.

READFILE fuksiyasining paramerlari ham berilganlarning import ustasi kabi imkoniyat beradi , lekin muloqat oynasiz.

Mathcad muxiti varaqani qayta ko'rib chiqqach har doim READFILE funksiyasi bajariladi, shuning uchun u Excel fayllarini har vaqtda tiklaydi.

READFILE (filePath, "Excel") yoki READEXCEL (filePath)

Bu parametr Excel faylidagi xamma berilganlarni kema-ket xisoblaydi. Ayrim xollarda berilganlarni tozalashni so'raydi.

READFILE (filePath, "Excel", startRow, startCol) yoki

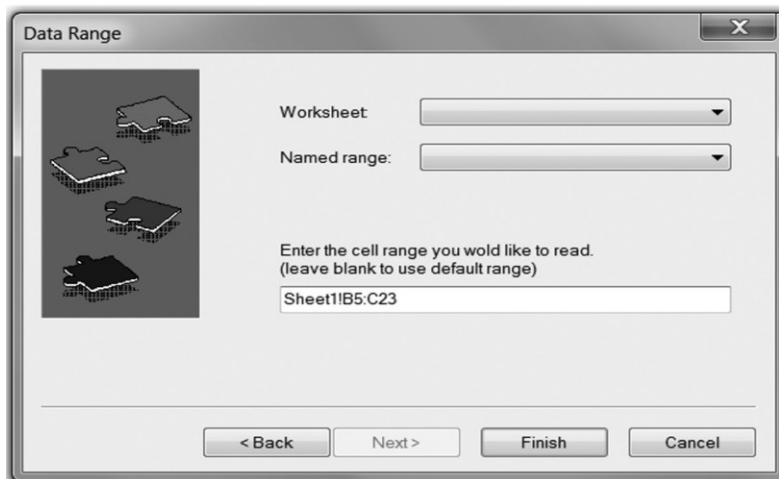
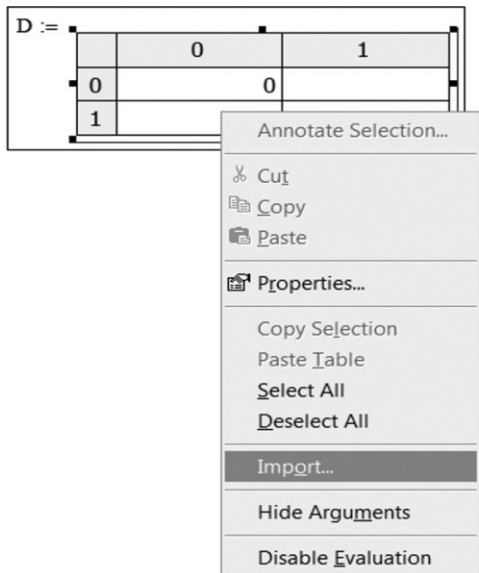
READEXCEL (filePath, "B2:B4")

MathCad startRow va start Collardan boshlab ma'lumotlani uzliksiz o'qiydi.

IB:READEXCEL(excelfile_path,"B5:C10")

IB := READEXCEL(excelfile _ path,"B5:C10")		
IB =	2.234	0
	3.134	2
	5.593	4
	8.952	6
	12.311	8
	14.771	10

14.771	10
15.311	8



D :=

	0	1
0	2.234	0
1	3.134	2
2	5.593	4
3	8.952	6
4	12.311	8
5	14.771	10
6	15.671	12
7	14.771	14
8	12.311	16
9	8.952	...

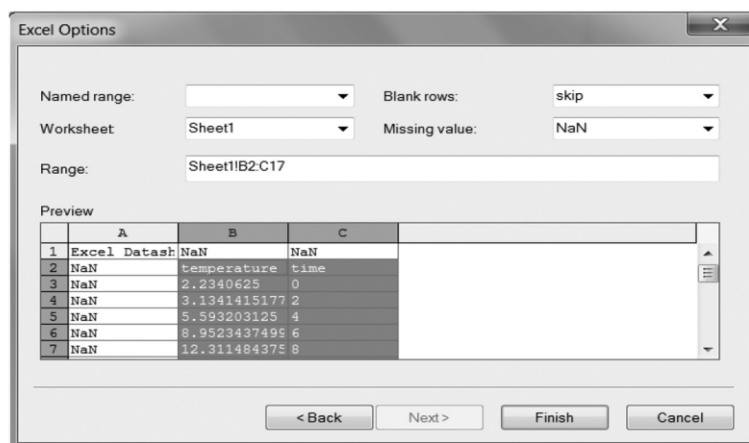
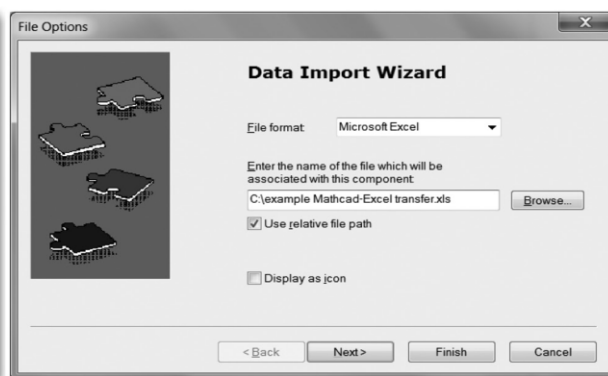
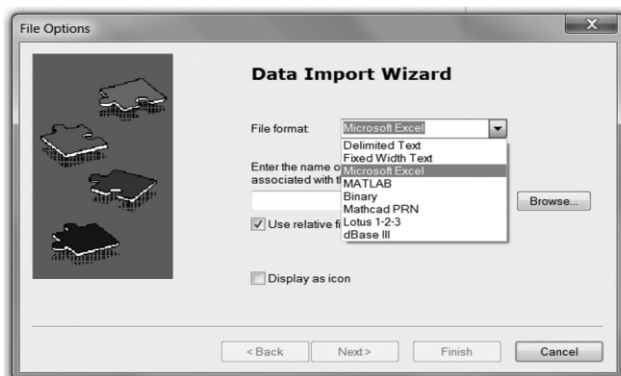
19.5-rasm. Import Excel dan MathCad muxitiga.

D :=



C:\Us...\example Mathcad-Excel transfer.xlsx

Import Excel dan MathCad muxitiga.



	0	1
0	"temperature"	"time"
1	2.234	0
2	3.134	2
3	5.593	4
4	8.952	6
5	12.311	8
6	14.771	10
7	15.671	12
8	14.771	14
9	12.311	16
10	8.952	18
11	5.593	20
12	3.134	22
13	2.154	...

19.6-rasm.Import Exceldan *import ustasi orqali* MathCad muxitiga.

```
filePath := "example Mathcad-Excel transfer.xlsx"
```

```
excelfile_path := concat(CWD, filePath)
```

```
D := READFILE(excelfile_path, "Excel")
```

D =

	1	2	3
1	"Excel Datasheet"	NaN	NaN
2	NaN	NaN	NaN
3	NaN	NaN	NaN
4	NaN	"temperature"	"time"
5	NaN	2.234	0
6	NaN	3.134	2
7	NaN	5.593	4
8	NaN	8.952	6
9	NaN	12.311	8
10	NaN	14.771	10
11	NaN	15.671	12
12	NaN	14.771	14
13	NaN	12.311	16
14	NaN	8.952	...

19.7-rasm. READFILE funksiyasidan qanday foydalanishga misol.

READFILE (filePath, "Excel", vRow, vCol)

vRow va vCol vektorlar satirlarni ishga tushirish va to'xtatishni ko'rsatadi.

$$vRows := \begin{pmatrix} 5 \\ 10 \end{pmatrix} \quad vCols := \begin{pmatrix} 2 \\ 3 \end{pmatrix}$$

$$AF := READFILE(excelfile_path, "Excel", vRows, vCols)$$

$$AF = \begin{pmatrix} 2.234 & 0 \\ 3.134 & 2 \\ 5.593 & 4 \\ 8.952 & 6 \\ 12.311 & 8 \\ 14.771 & 10 \end{pmatrix}$$

Shuningd
imkoniyatiga eg
ishlatmoqchi bo

$$vL = \begin{pmatrix} 14.771 & 10 \\ 12.311 & 8 \\ 8.952 & 6 \end{pmatrix}$$

r kengaytmasin tanlash
arini MathCad muxitida

20-ma'ruza. Matlab dasturi haqida umumiy tushuncha.

Reja:

1. Matlab dasturi ishchi oynasi, menyu bo'limlari.
2. Matlab dasturida funksiyalar masteri bilan ishlash.
3. Diagrammalar, ikki xamda uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.
4. Kimyo injinering masalalarini yechimini topishda MatLab dasturi imkoniyatlaridan yuqori darajada foydalanish.
5. Vektorli yoki matrisalar shaklda saqlanadigan ma'lumot hamda element nomlanishi va boshqalar.

Tayanch so'z va iboralar: Matlab, diagrammalar, vektorli, matrisa, funksiyalar masteri, modellash, algoritim.

MATLAB tizimi – kompyuterda turli yo'nalishdagi: matematika, fizika, mexanika, boshqaruv va muhandislik masalalarini yechish, turli xil energetik, mexanik va dinamik tizimlarni modellashtirish, loyihalash, tavsiflash va tahlil qilish masalalarining aniq, tez va samarali hal etish uchun mo'ljallangan tizim va turli xil sohali foydalanuvchilarga mo'ljallangan dasturlash tilidir .

MATLAB dasturlash tili sifatida 1970- yillarning oxirida Kliv Mouler (Cleve Mouler) tomonidan yaratilgan. MATLAB atamasi inglizcha Matrix Laboratory so'zlaridan kelib chiqqan bo'lib, matritsa laboratoriyasi degan ma'noni anglatadi. MATLABda matematik hisoblashlar, modellash algoritmlarini yaratish, ma'lumotlarni tahlil, tadqiq qilish hamda vizuallashtirish, ilmiy va injinerlik grafikasi va ilovalarni loyihalash va boshqalarda foydalanish mumkin.

MATLAB yordamida aniq masalalarni yechish boshqa skalyar dasturlash tillaridagiga (masalan, C++) nisbatan bir necha marta tez bajariladi. MATLABning boshqa dasturlash tillaridan farq qiluvchi xususiyati shuki, u o'z ishida ma'lumotlarni matritsalar shaklida tashkil etish usulidan foydalaniladi .

MATLAB tizimi ikkita katta qismlardan tashkil topgan: MATLAB yadrosi va qo'shimcha kutubxonalar (yoki "toolboxes" – "asboblar komplektlari"). MATLAB yadrosi asosiy funksiyalarni va umummo'ljallangan amallarni ta'minlaydi. Kutubxonalar esa maxsus ixtisoslashgan funksiyalarni saqlaydi va bu ixtisoslashgan funksiyalar foydalanuvchilarga qat'iy aniqlangan sohada hisoblashlarni va ma'lumotlarni qayta ishlash imkoniyatini beradi. MATLAB Toolboxlari funksiyalarning ma'lum maqsadga ko'ra yig'ilgan tizimli majmuasi bo'lib xususiy masalalarni yechish uchun xizmat qiladi.

Keyingi yillarda loyihachilar matematik tizimlarning integratsiyalashuviga va ulardan birgalikda foydalanishga katta e'tibor bermoqdalar. Murakkab matematik masalalarni bir necha tizimlar yordamida yechish eng yaxshi va mos vositalarni tanlash imkoniyatini beradi, shuningdek, olinadigan natijalarning ishonchliligini orttiradi.

Foydalaniluvchi M-fayl matnli formatga ega bo'lganligi sababli unga har qanday yangi buyruqni, operatorni yoki funktsiyani kiritishi mumkin. Keyin undan birlashtirilgan funktsiya yoki operator kabi foydalana oladi. Bunda Basic, C yoki Paskal dasturlash tillaridan farqli ravishda yangi funksiyalarni e'lon qilish shart

emas. Bu jihatdan MATLAB Logo va Fortran tillariga o'xshash, lekin MATLABda yangi ma'lumotlar fayl ko'rinishida diskda saqlanishi sababli operator va funksiyalar soni amalda chegaralanmagan. MATLAB muhitida ishlaganda foydalanuvchi avvalroq yaratilgan va diskda saqlab qo'yilgan ma'lumotlarni (o'zgaruvchilarni), hamda joriy seansda diskdagi fayllarda yaratilgan ma'lumotlar yozuvlarini ishchi oynaga yuklash imkoniyatiga egadir. Tizimning tayanch so'zlar to'plamiga maxsus belgilar arifmetik ishoralar va mantiqiy amallar, arifmetik, algebraik, trigonometrik va boshqa maxsus funksiyalar, Furening tez o'zgartirish funksiyalari va filtrlash, vektor va matritsaviy funksiyalar, kompleks sonlar bilan ishlash uchun vositalar, Dekart va qutbli koordinatalar tizimlarida grafiklar qurish uchun operatorlar, uch o'lchamli sirtlar va boshqalar kiradi. Umuman olganda, MATLAB tayyor vositalarning katta to'plamini taqdim etadi (ularning katta qismini – m-fayllar ko'rinishidagi tashqi kengaytmalar tashkil etadi).

Asosiy tushuncha “bajarish orqali o'rganish” hisoblanadi. Shu sababli o'rganishning eng yaxshi yo'li insonning o'zi harakat qilishidir. MATLAB buyruqlarining o'rganilishiga asosiy yo'l bu misollar orqali o'rganish hisoblanadi. Ushbu mavzuda MATLAB qay darajada tenglamalar va matematik hisoblashlarning boshqarilishi yoki bajarilishi mumkinligi to'g'risida aytib o'tilgan.

MATLAB original holda yozilgan bo'lib, LINPACK (chiziqli tizimlar paketlar) va EISPACK (Eign tizimlar paketi) loyihalari tomonidan rivojlantirilgan.

Texnikaviy muammolarni yechishda MATLAB dasturi boshqa odatiy kompyuter dasturlari bilan qiyoslaganda juda ko'p ustunligi mavjuddir. Bu dasturiy paket 1984-yildan beri ishlatilib, hozirda u ko'pgina ishlab chiqarish va universitetlarda standart vosita hisoblanadi. Unda Bunga qo'shimcha ravishda esa grafik qo'shimchalar mavjud bo'lib natijalar juda tez vezuallash imkonini beradi. Bunda uskunalar paneli jarayonni hisoblash, boshqarish nazariyasi modellash tirish optimallashtirish, va boshqa ko'pgina amaliy fan va muhandislik sohalari mavjud.

MATLAB ko'plab amaliy masalalarni yechish imkoniyatini beruvchi operatorlar va funksiyalarga ega. Ular yordamida ko'plab amaliy masalalarni yechish mumkin. Bunday masalalarni yechish uchun oldin murakkab dasturlarni tuzish zarur bo'lar edi. Misol uchun, matritsalar bilan amallar, hosila va integralning qiymatlarini hisoblash va boshqalar. MATLABda bunday masalalarni yechishga imkon beruvchi tayyor funksiyalarning soni kengaytma paketlarni ham qo'shib hisoblaganda mingdan ortiqni tashkil qiladi va ularning soni uzluksiz ortib bormoqda.

Asosiy xususiyatlari

Ushbu ma'ruzaning hech bo'lmaganda birinchi bosqichini o'rganish (kirish yoki chiqish) mo'ljallangan. Bunda:

MATLABni chaqirish

Eng oddiy hisoblashlar

MATLABdan eng tez chiqib ketish

MATLAB tizimi bilan keng tarqalgan matematik tizimlar (Mathcad, Maple va Mathemati) integrallashuvi mumkin. Matematik tizimlarni zamonaviy matnli

protessorlar bilan birlashtirishga intilish ham mavjud. Masalan, MATLAB yangi versiyalarining vositasi —matn protessorlarida tayyorlanayotgan hujjatning kerakli joylariga MATLAB hujjatlari va sonli, jadval yoki grafik ko`rinishdagi hisoblash natijalarini qo`yish imkoniyatini beradi. Natijada "jonli" elektron kitoblarni tayyorlash mumkin. Ularda namoyish qilinayotgan misollarni operativ tarzda o`zgartirish mumkin. Masalan, boshlang`ich shartlarni o`zgartirib, masalani yechish natijalarining o`zgarishini kuzatish mumkin. MATLAB da grafiklarni Microsoft PowerPoint -slaydlariga eksport qilishning takomillashgan vositalari ham ko`zda tutilgan.

MATLAB da tizimni kengaytirish masalalari maxsus kengaytirish paketlari - Toolbox asboblari to`plami yordamida hal qilinadi. Ularning ko`plari boshqa dasturlar bilan integratsiyalashuv uchun maxsus vositalarga ega. MATLAB tizimi bloklar ko`rinishida berilgan, dinamik tizim va qurilmalarni modellash uchun yaratilgan Simulink dasturiy tizimi bilan ham integratsiyalashgan. Vizual-yunaltirilgan dasturlash prinsiplariga asoslangan Simulink murakkab qurilmalarni yuqori aniklikda modellash imkoniyatini beradi.

MATLABni ishga tushirish

MATLAB dasturini o`rnatgandan so`ng, kompyuter ishchi stoli uni oynasidan MATLAB belgisi ustiga ikki marta bosiladi. MATLABni ishga tushirganda, boshqa bir yangi oyna ochiladi. Kompyuter ish stoli oynasida boshqa dasturlarning oynalari joylashadi. Ish stolining asosiy ish qurollari:

Command Window(buyruq oynasi)

Command History(buyruqlar tarixi)

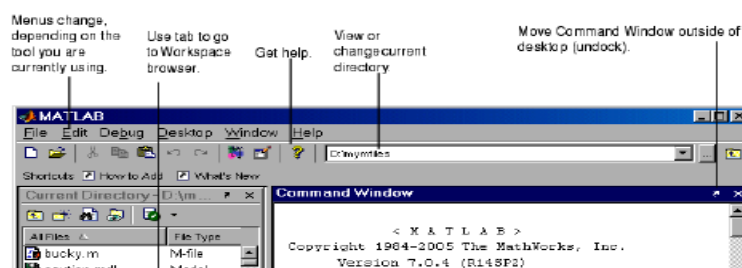
Workspace (ishchi oyna)

Current Directory (joriy papka joylashuvi)

Help Browser (yordam brauzeri)

Start button (start tugmasi)

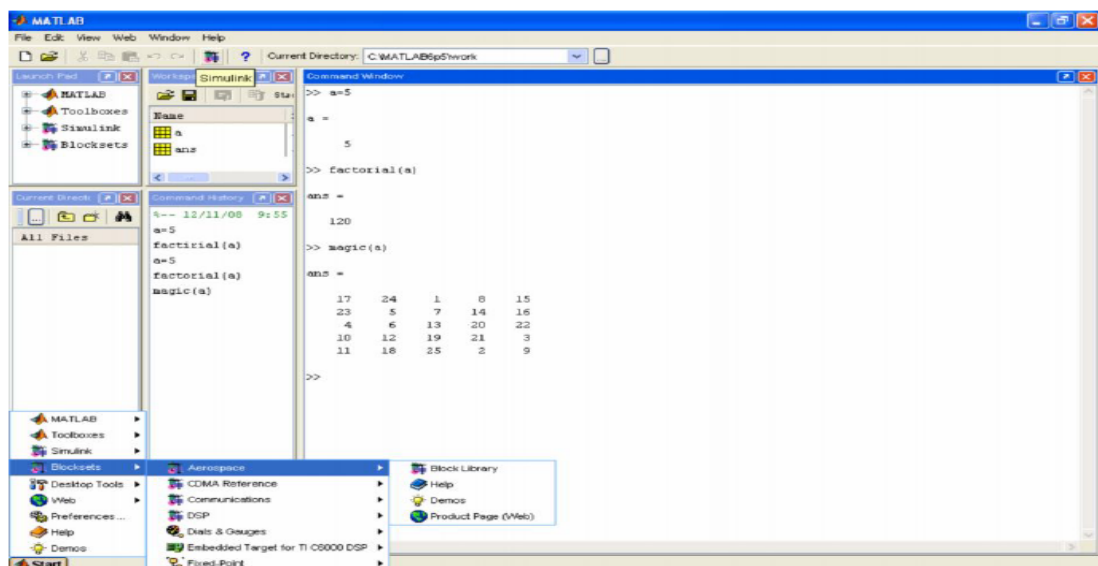
Dastlabki holatda, MATLAB ishga tushirilganda, muloqot oynasi 20.1 – rasmdagi kabi bo`ladi. MATLAB ish oynasining boshqa ko`rinishlari ham mavjud. Oynalar joylashishi va o`lchamini ixtiyoriy ravishda o`zgartirish imkoniyati mavjud. Quyida eng oddiy hisoblashlar berilgan bo`lib, MATLAB ishlayotgan kompyuter haqida siz yetarlicha bilasiz deb qaraymiz.



20.1-rasm. Matlab ishchi oynasi

MATLAB tizimining asosiy oynasi quyidagicha ko'rinishda bo'lib, quyidagi bo'limlardan iborat:

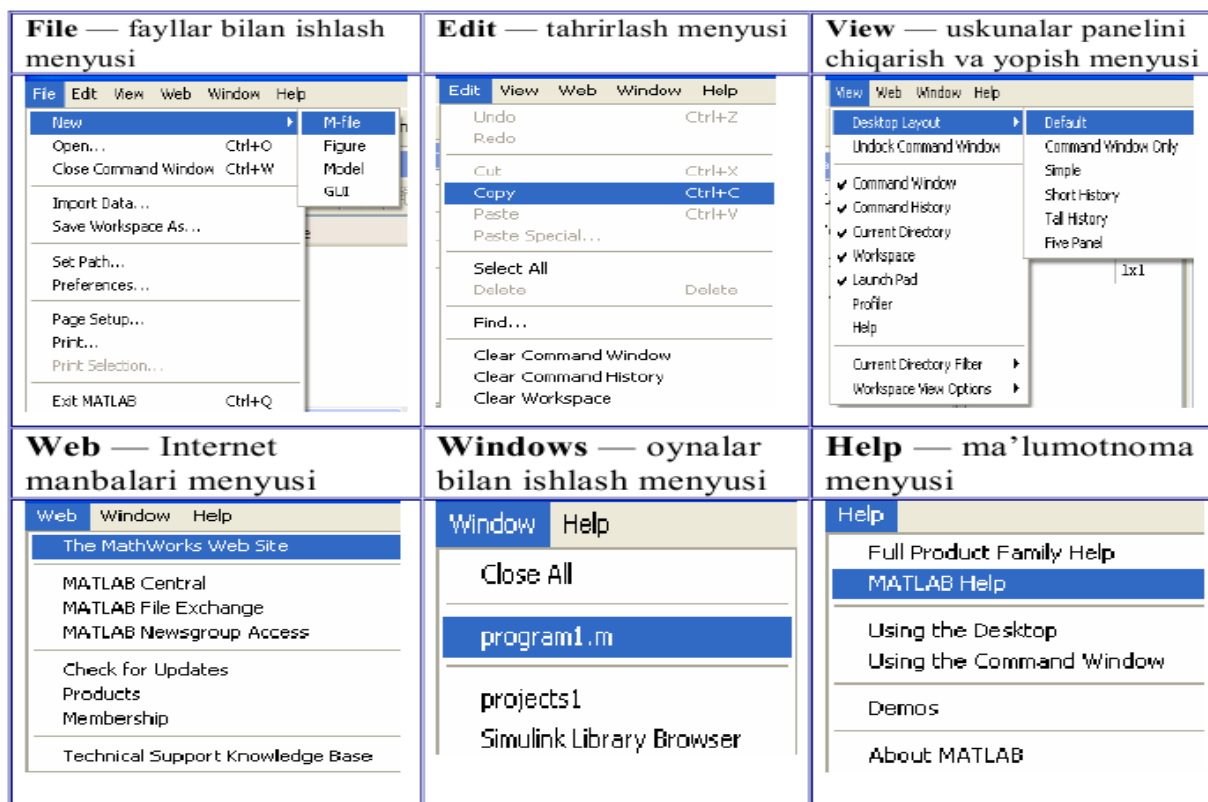
1.	Sarlavha satri;	5.	Komandalar ishchi varag'i;
2.	Asosiy menyular satri;	6.	Oxirgi yozilgan komandalar ro'yxati;
3.	Uskunalar paneli;	7.	Holat satri.
4.	Ishchi soha;		



MATLAB tizimining asosiy oynasi

20.2-rasm. Matlab tizimining asosiy oynasi

Asosiy menyular satri quyidagi menyulardan iborat.



20.3-rasm. Matlab tizimining asosiy manyular satri

Matlab ning ishchi varag'i tom ma'noda uch qismga bo'linadi:

Buyruqlarni kiritish maydoni- buyruqlar satridan tashkil topgan. Har bir buyruq satri >> simvoli (bu simvol avtomatik tarzda buyruqlar satrining boshida bo'ladi va uni yozish shart emas) bilan boshlanadi;

Natijani chiqarish maydoni- kiritilgan buyruqlarni qayta ishlangandan so'ng hosil bo'lgan ma'lumotlar (analitik ifodalar, natijalar va xabarlar) ni o'z ichiga oladi;

Matnli izohlar maydoni- ro'y bergan xatoliklar yoki bajarilgan buyruqlarga izohlar, turli xarakterdagi xabarlar.

Buyruqlar Enter tugmasini bosish (bir marta) orqali amalga oshiriladi.

Matlab ni besh qismga ajratish mumkin:

1. Matlab tili
2. Matlab muhiti
3. boshqariluvchi grafika
4. Matematik funksiyalar bibliotekasi
5. Dasturiy interfeysi

Matlab tili

MATLAB dasturlash tilida boshqa dasturlash tillari kabi lotin alifbosining A dan Z gacha barcha katta va kichik harflari, 0 dan 9 gacha arab raqamlaridan foydalaniladi. Katta va kichik harflar, xuddi C++ dasturlash tilidagidek, ham o'zgaruvchi sifatida, ham ozgarmas sifatida bir-biridan farq qiladi. Lotin alifbosi harflaridan tashqari, klaviaturadagi barcha maxsus belgilardan foydalaniladi.

Buyruqlar Enter tugmasini bosish (bir marta) orqali amalga oshiriladi. O'zgaruvchi nomi nechta va qanaqa belgi yoki belgilardan iborat bo'lishidan qat'iy nazar, lotin harflaridan boshlanib, 63 ta belgidan oshmasligi shart. Katta va kichik harflar bir-biridan farq qiladi. Agar buyruq o'zgaruvchi nomi yozilmay bajarilsa, buyruq natijasi maxsus ans(inglizcha answer-javob) o'zgaruvchisi orqali beriladi. Ishchi sohadagi o'zgaruvchilar haqidagi ma'lumotlarni who yoki whos buyruqlari orqali ko'rish mumkin.

MATLAB da barcha ma'lumotlar matritsa yoki massiv ko'rinishida("MATLAB" so'zi inglizcha "Matrix Laboratory", yani "Matritsali Laboratoriya" so'zlarining qisqartirilgan ifodasidir) tasvirlanadi. Hattoki, skalyar o'zgaruvchilarni umumiy holda 1x1 o'lchovli massiv(matritsa) deb qarash qabul qilingan. Shuning uchun ham massiv va matritsalar ustida ishlash, MATLAB da samarali ishlashda muhim ahamiyatga ega.

Massiv – bir turdagi ma'lumotlarning raqamlangan va tartiblangan to'plamidir. Massivning nomi bo'lishi shart. Massivlar o'lchovi yoki o'lchami bilan bir-biridan farq qiladi: bir o'lchovli, ikki o'lchovli, ko'p o'lchovli. Massiv elementlariga murojaat qilish indekslar orqali amalga oshiriladi. MATLAB da massiv elementlarini raqamlash bir(1)dan boshlangani uchun indeksleri birga teng yoki katta bo'ladi.

Oddiy arifmetik amallar.

MATLAB da arifmetik amallar yetarlicha kengaytirilgan, hamda matritsaviy va arifmetik amallarni o'z ichiga oladi. Quyida arifmetik va matritsaviy amallar keltirilgan:

1) o'zgarmaslar

T. R	O'zgarmaslar	O'zgarmaslarning aytilishi
1.	pi	π soni
2.	i yoki j	mavhum son
3.	inf	cheksizlik
4.	NaN	$\frac{0}{0}$ ko'rinishdagi aniqmaslik
5.	true	mantiqiy rost
6.	false	mantiqiy yolg'on

20.4-rasm. O'zgarmaslar

2) arifmetik amallar:

T. R	Arifmetik amal belgilari	Arifmetik amal belgilari aytilishi
1.	+	Qo'shish(skalyar yoki matritsaviy)
2.	-	Ayirish(skalyar yoki matritsaviy)
3.	*	Ko'paytirish(skalyar yoki matritsaviy)
4.	/	Bo'lish(skalyar)
5.	^	Darajaga ko'tarish(skalyar yoki matritsaviy)
6.	.*	Massiv mos elementlari buyicha ko'paytirish
7.	./	O'lchovlari bir xil massiv mos elementlari buyicha bo'lish
8.	.^	Massiv mos elementlari buyicha darajaga ko'tarish
9.	\	Matritsaviy chapdan ungga bo'lish
10.	.\	Massiv mos elementlari buyicha chapdan ungga bo'lish
11.	'	Qo'shma matritsani hisoblash
12.	.'	Transponerlash

20.5-rasm. Arifmetik amallar

MATLAB da matematik ifodalar ma'lum bir bajarilish tartibiga asosan bajariladi. Avval mantiqiy amallar, so'ngra arifmetik amallar: avval daraja, keyin ko'paytirish va bo'lish, undan keyin esa qo'shish va ayirish bajariladi. Agar ifodada qavslar bo'lsa, avval qavs ichidagi ifoda yuqoridagi tartibda bajariladi.

3) munosabat amallari:

T. R	Operator(sintaksis)	Amal belgilari aytilishi
1.	$=$; ($x = y$)	Teng
2.	$\sim$; ($x \sim y$)	Teng emas
3.	$<$; ($x < y$)	Kichik
4.	$>$; ($x > y$)	Katta
5.	$\leq$; ($x \leq y$)	Kichik yoki teng
6.	$\geq$; ($x \geq y$)	Katta yoki teng

4) mantiqiy amallar:

T. R	Operator(sintaksis)	Amal belgilari aytilishi
1.	& ; <i>and</i> (<i>and</i> (<i>a</i>, <i>b</i>))	va
2.	 ; <i>or</i> (<i>or</i> (<i>a</i>, <i>b</i>))	yoki
3.	~ ; <i>not</i> (<i>not</i> (<i>a</i>, <i>b</i>))	inkor
4.	<i>xor</i> (<i>xor</i> (<i>a</i>, <i>b</i>))	
5.	<i>any</i> (<i>any</i> (<i>a</i>))	
6.	<i>all</i> (<i>all</i> (<i>a</i>))	

20.6-rasm. Munosabat va mantiqiy amallar.

Buyruqni kiritish uchun, Command windowsdan (>>) ko'rsatkichiga sichqonchani olib borib belgilaymiz. Odatda, ikki xil ko'rsatkich mavjud:

>> To'liq ishlaydigan MATLAB dasturi uchun

EDU> Ta'lim uchun olingan MATLAB turi

(Eslatma: qaydni osonlashtirish uchun, >>ko'rsatkichini garchi MATLAB versiyasi ta'lim maqsadlarida bo'lishiga qaramasdan standart ko'rsatkich sifatida ishlatamiz.

21-ma'ruza. Matlab dasturi haqida umumiy tushuncha.

Reja:

1. Matlab dasturi ishchi oynasi, menyu bo'limlari.
2. Matlab dasturida funksiyalar masteri bilan ishlash.
3. Diagrammalar, ikki xamda uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.
4. Kimyo injineriing masalalarini yechimini topishda MatLab dasturi imkoniyatlaridan yuqori darajada foydalanish.
5. Vektorli yoki matrisalar shaklda saqlanadigan ma'lumot hamda element nomlanishi va boshqalar.

Tayanch so'z va iboralar: Matlab, diagrammalar, vektorli, matrisa, funksiyalar masteri, modellash, algoritim.

MATLAB tizimi – kompyuterda turli yo'nalishdagi: matematika, fizika, mexanika, boshqaruv va muhandislik masalalarini yechish, turli xil energetik, mexanik va dinamik tizimlarni modellashtirish, loyihalash, tavsiflash va tahlil qilish masalalarining aniq, tez va samarali hal etish uchun mo'ljallangan tizim va turli xil sohali foydalanuvchilarga mo'ljallangan dasturlash tilidir .

MATLAB dasturlash tili sifatida 1970- yillarning oxirida Kliv Mouler (Cleve

Mouler) tomonidan yaratilgan. MATLAB atamasi inglizcha Matrix Laboratory soʻzlaridan kelib chiqqan boʻlib, matritsa laboratoriyasi degan maʼnoni anglatadi. MATLABda matematik hisoblashlar, modellash algoritmlarini yaratish, maʼlumotlarni tahlil, tadqiq qilish hamda vizuallashtirish, ilmiy va injinerlik grafikasi va ilovalarni loyihalash va boshqalarda foydalanish mumkin.

MATLAB yordamida aniq masalalarni yechish boshqa skalyar dasturlash tillaridagiga (masalan, C++) nisbatan bir necha marta tez bajariladi. MATLABning boshqa dasturlash tillaridan farq qiluvchi xususiyati shuki, u oʻz ishida maʼlumotlarni matritsalar shaklida tashkil etish usulidan foydalaniladi.

MATLAB tizimi ikkita katta qismlardan tashkil topgan: MATLAB yadrosi va qoʻshimcha kutubxonalar (yoki “toolboxes” – “asboblar komplektlari”). MATLAB yadrosi asosiy funksiyalarni va umummoʻljallangan amallarni taʼminlaydi. Kutubxonalar esa maxsus ixtisoslashgan funksiyalarni saqlaydi va bu ixtisoslashgan funksiyalar foydalanuvchilarga qatʼiy aniqlangan sohada hisoblashlarni va maʼlumotlarni qayta ishlash imkoniyatini beradi. MATLAB Toolboxlari funksiyalarning maʼlum maqsadga koʻra yigʻilgan tizimli majmuasi boʻlib xususiy masalalarni yechish uchun xizmat qiladi.

Keyingi yillarda loyihachilar matematik tizimlarning integratsiyalashuviga va ulardan birgalikda foydalanishga katta eʼtibor bermoqdalar. Murakkab matematik masalalarni bir necha tizimlar yordamida yechish eng yaxshi va mos vositalarni tanlash imkoniyatini beradi, shuningdek, olinadigan natijalarning ishonchliligini orttiradi.

Foydalaniluvchi M-fayl matnli formatga ega boʻlganligi sababli unga har qanday yangi buyruqni, operatorni yoki funktsiyani kiritishi mumkin. Keyin undan birlashtirilgan funktsiya yoki operator kabi foydalana oladi. Bunda Basic, C yoki Paskal dasturlash tillaridan farqli ravishda yangi funksiyalarni eʼlon qilish shart emas. Bu jihatdan MATLAB Logo va Fortran tillariga oʻxshash, lekin MATLABda yangi maʼlumotlar fayl koʻrinishida diskda saqlanishi sababli operator va funksiyalar soni amalda chegaralanmagan. MATLAB muhitida ishlaganda foydalanuvchi avvalroq yaratilgan va diskda saqlab qoʻyilgan maʼlumotlarni (oʻzgaruvchilarni), hamda joriy seansda diskdagi fayllarda yaratilgan maʼlumotlar yozuvlarini ishchi oynaga yuklash imkoniyatiga egadir. Tizimning tayanch soʻzlar toʻplamiga maxsus belgilar arifmetik ishoralar va mantiqiy amallar, arifmetik, algebraik, trigonometrik va boshqa maxsus funksiyalar, Furening tez oʻzgartirish funksiyalari va filtrlash, vektor va matritsaviy funksiyalar, kompleks sonlar bilan ishlash uchun vositalar, Dekart va qutbli koordinatalar tizimlarida grafiklar qurish uchun operatorlar, uch oʻlchamli sirtlar va boshqalar kiradi. Umuman olganda, MATLAB tayyor vositalarning katta toʻplamini taqdim etadi (ularning katta qismini – m-fayllar koʻrinishidagi tashqi kengaytmalar tashkil etadi).

Asosiy tushuncha “bajarish orqali oʻrganish” hisoblanadi. Shu sababli oʻrganishning eng yaxshi yoʻli insonning oʻzi harakat qilishidir. MATLAB buyruqlarining oʻrganilishiga asosiy yoʻl bu misollar orqali oʻrganish hisoblanadi. Ushbu mavzuda MATLAB qay darajada tenglamalar va matematik hisoblashlarning boshqarilishi yoki bajarilishi mumkinligi toʻgʻrisida aytib

o`tilgan.

MATLAB original holda yozilgan bo`lib, LINPACK (chiziqli tizimlar paketlar) va EISPACK (Eign tizimlar paketi) loyihalari tomonidan rivojlantirilgan.

Texnikaviy muammolarni yechishda MATLAB dasturi boshqa odatiy kompyuter dasturlari bilan qiyoslaganda juda ko`p ustunligi mavjuddir. Bu dasturiy paket 1984-yildan beri ishlatilib, hozirda u ko`pgina ishlab chiqarish va universitetlarda standart vosita hisoblanadi. Unda Bunga qo`shimcha ravishda esa grafik qo`shimchalar mavjud bo`lib natijalar juda tez vezuallash imkonini beradi. Bunda uskunalar paneli jarayonni hisoblash, boshqarish nazariyasi modellash tirish optimallashtirish, va boshqa ko`pgina amaliy fan va muhandislik sohalari mavjud.

MATLAB ko`plab amaliy masalalarni yechish imkoniyatini beruvchi operatorlar va funksiyalarga ega. Ular yordamida ko`plab amaliy masalalarni yechish mumkin. Bunday masalalarni yechish uchun oldin murakkab dasturlarni tuzish zarur bo`lar edi. Misol uchun, matritsalar bilan amallar, hosila va integralning qiymatlarini hisoblash va boshqalar. MATLABda bunday masalalarni yechishga imkon beruvchi tayyor funksiyalarning soni kengaytma paketlarni ham qo`shib hisoblaganda mingdan ortiqni tashkil qiladi va ularning soni uzluksiz ortib bormoqda.

Asosiy xususiyatlari

Ushbu ma`ruzaning hech bo`lmaganda birinchi bosqichini o`rganish (kirish yoki chiqish) mo`ljallangan. Bunda:

MATLABni chaqirish

Eng oddiy hisoblashlar

MATLABdan eng tez chiqib ketish

MATLAB tizimi bilan keng tarqalgan matematik tizimlar (Mathcad, Maple va Mathemati) integrallashuvi mumkin. Matematik tizimlarni zamonaviy matnli protsessorlar bilan birlashtirishga intilish ham mavjud. Masalan, MATLAB yangi versiyalarining vositasi —matn protsessorlarida tayyorlanayotgan hujjatning kerakli joylariga MATLAB hujjatlari va sonli, jadval yoki grafik ko`rinishdagi hisoblash natijalarini qo`yish imkoniyatini beradi. Natijada "jonli" elektron kitoblarni tayyorlash mumkin. Ularda namoyish qilinayotgan misollarni operativ tarzda o`zgartirish mumkin. Masalan, boshlang`ich shartlarni o`zgartirib, masalani yechish natijalarining o`zgarishini kuzatish mumkin. MATLAB da grafiklarni Microsoft PowerPoint -slaydlariga eksport qilishning takomillashgan vositalari ham ko`zda tutilgan.

MATLAB da tizimni kengaytirish masalalari maxsus kengaytirish paketlari - Toolbox asboblari to`plami yordamida hal qilinadi. Ularning ko`plari boshqa dasturlar bilan integratsiyalashuv uchun maxsus vositalarga ega. MATLAB tizimi bloklar ko`rinishida berilgan, dinamik tizim va qurilmalarni modellash uchun yaratilgan Simulink dasturiy tizimi bilan ham integratsiyalashgan. Vizual-yunaltirilgan dasturlash prinsiplariga asoslangan Simulink murakkab qurilmalarni yuqori aniklikda modellash imkoniyatini beradi.

MATLABni ishga tushirish

MATLAB dasturini o`rnatgandan so`ng, kompyuter ishchi stoli uni oynasidan MATLAB belgisi ustiga ikki marta bosiladi. MATLABni ishga tushirganda,

boshqa bir yangi oyna ochiladi. Kompyuter ish stoli oynasida boshqa dasturlarning oynalari joylashadi. Ish stolining asosiy ish qurollari:

- Command Window(buyruq oynasi)
- Command History(buyruqlar tarixi)
- Workspace (ishchi oyna)
- Current Directory (joriy papka joylashuvi)
- Help Browser (yordam brauzeri)
- Start button (start tugmasi)

Dastlabki holatda, MATLAB ishga tushirilganda, muloqot oynasi 20.1 – rasmdagi kabi bo'ladi. MATLAB ish oynasining boshqa ko'rinishlari ham mavjud. Oynalar joylashishi va o'lchamini ixtiyoriy ravishda o'zgartirish imkoniyati mavjud. Quyida eng oddiy hisoblashlar berilgan bo'lib, MATLAB ishlayotgan kompyuter haqida siz yetarlicha bilasiz deb qaraymiz.

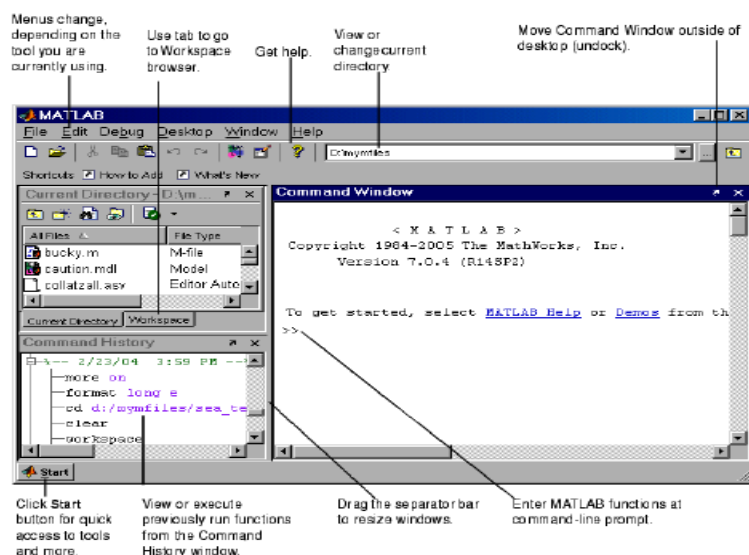
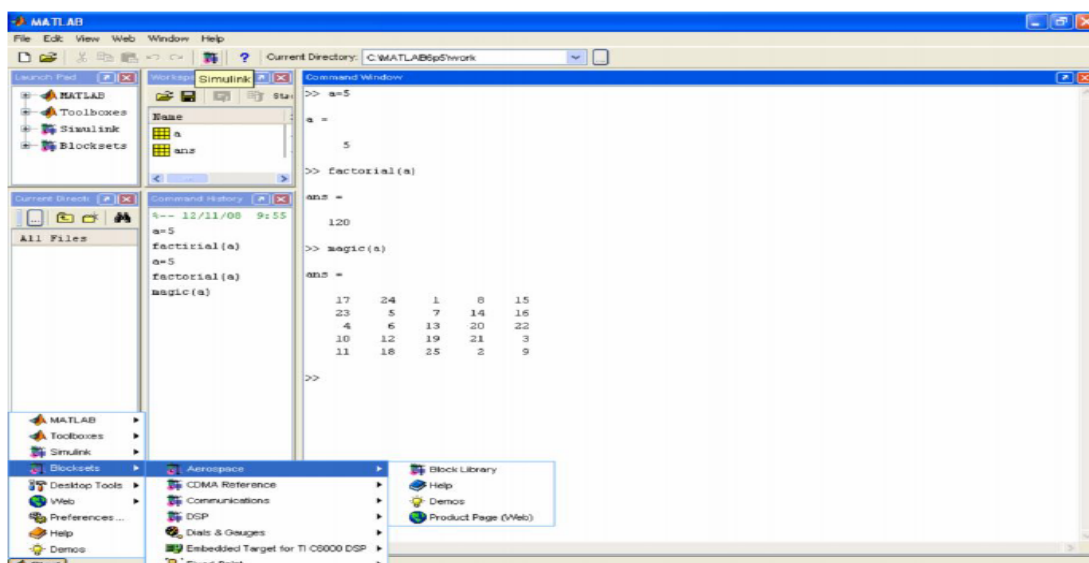


Figure 1.1: The graphical interface to the MATLAB workspace

21.1-rasm. Matlab ishchi oynasi

MATLAB tizimining asosiy oynasi quyidagicha ko'rinishda bo'lib, quyidagi bo'limlardan iborat:

1.	Sarlavha satri;	5.	Komandalar ishchi varag'i;
2.	Asosiy menyular satri;	6.	Oxirgi yozilgan komandalar ro'yxati;
3.	Uskunalar paneli;	7.	Holat satri.
4.	Ishchi soha;		



MATLAB tizimining asosiy oynasi

21.2-rasm. Matlab tizimining asosiy oynasi

Asosiy menyular satri quyidagi menyulardan iborat.

File — fayllar bilan ishlash menyusi 	Edit — tahrirlash menyusi 	View — uskunalar panelini chiqarish va yopish menyusi
Web — Internet manbalari menyusi 	Windows — oynalar bilan ishlash menyusi 	Help — ma'lumotnoma menyusi

21.3-rasm. Matlab tizimining asosiy menyular satri

Matlab ning ishchi varag'i tom ma'noda uch qismga bo'linadi:

Buyruqlarni kiritish maydoni- buyruqlar satridan tashkil topgan. Har bir

buyruq satri >> simvoli (bu simvol avtomatik tarzda buyruqlar satrining boshida bo'ladi va uni yozish shart emas) bilan boshlanadi;

Natijani chiqarish maydoni- kiritilgan buyruqlarni qayta ishlangandan so'ng hosil bo'lgan ma'lumotlar (analitik ifodalar, natijalar va xabarlar) ni o'z ichiga oladi;

Matnli izohlar maydoni- ro'y bergan xatoliklar yoki bajarilgan buyruqlarga izohlar, turli xarakterdagi xabarlar.

Buyruqlar Enter tugmasini bosish (bir marta) orqali amalga oshiriladi.

Matlab ni besh qismga ajratish mumkin:

1. Matlab tili
2. Matlab muhiti
3. boshqariluvchi grafika
4. Matematik funksiyalar bibliotekasi
5. Dasturiy interfeysi

Matlab tili

MATLAB dasturlash tilida boshqa dasturlash tillari kabi lotin alifbosining A dan Z gacha barcha katta va kichik harflari, 0 dan 9 gacha arab raqamlaridan foydalaniladi. Katta va kichik harflar, xuddi C++ dasturlash tilidagidek, ham o'zgaruvchi sifatida, ham ozgarmas sifatida bir-biridan farq qiladi. Lotin alifbosi harflaridan tashqari, klaviaturadagi barcha maxsus belgilardan foydalaniladi.

Buyruqlar Enter tugmasini bosish (bir marta) orqali amalga oshiriladi. O'zgaruvchi nomi nechta va qanaqa belgi yoki belgilardan iborat bo'lishidan qat'iy nazar, lotin harflaridan boshlanib, 63 ta belgidan oshmasligi shart. Katta va kichik harflar bir-biridan farq qiladi. Agar buyruq o'zgaruvchi nomi yozilmay bajarilsa, buyruq natijasi maxsus ans(inglizcha answer-javob) o'zgaruvchisi orqali beriladi. Ishchi sohadagi o'zgaruvchilar haqidagi ma'lumotlarni who yoki whos buyruqlari orqali ko'rish mumkin.

MATLAB da barcha ma'lumotlar matritsa yoki massiv ko'rinishida("MATLAB" so'zi inglizcha "Matrix Laboratory", yani "Matritsali Laboratoriya" so'zlarining qisqartirilgan ifodasidir) tasvirlanadi. Hattoki, skalyar o'zgaruvchilarni umumiy holda 1x1 o'lchovli massiv(matritsa) deb qarash qabul qilingan. Shuning uchun ham massiv va matritsalar ustida ishlash, MATLAB da samarali ishlashda muhim ahamiyatga ega.

Massiv – bir turdagi ma'lumotlarning raqamlangan va tartiblangan to'plamidir. Massivning nomi bo'lishi shart. Massivlar o'lchovi yoki o'lchami bilan bir-biridan farq qiladi: bir o'lchovli, ikki o'lchovli, ko'p o'lchovli. Massiv elementlariga murojaat qilish indekslar orqali amalga oshiriladi. MATLAB da massiv elementlarini raqamlash bir(1)dan boshlangani uchun indeksleri birga teng yoki katta bo'ladi.

Oddiy arifmetik amallar.

MATLAB da arifmetik amallar yetarlicha kengaytirilgan, hamda matritsaviy va arifmetik amallarni o'z ichiga oladi. Quyida arifmetik va matritsaviy amallar keltirilgan:

1) o'zgarmaslar

T. R	O'zgarmaslar	O'zgarmaslarning aytilishi
1.	pi	π soni
2.	i yoki j	mavhum son
3.	inf	cheksizlik
4.	NaN	$\frac{0}{0}$ ko'rinishdagi aniqmaslik
5.	true	mantiqiy rost
6.	false	mantiqiy yolg'on

21.4-rasm. O'zgarmaslar

2) arifmetik amallar:

T. R	Arifmetik amal belgilari	Arifmetik amal belgilari aytilishi
1.	+	Qo'shish(skalyar yoki matritsaviy)
2.	-	Ayirish(skalyar yoki matritsaviy)
3.	*	Ko'paytirish(skalyar yoki matritsaviy)
4.	/	Bo'lish(skalyar)
5.	^	Darajaga ko'tarish(skalyar yoki matritsaviy)
6.	.*	Massiv mos elementlari buyicha ko'paytirish
7.	./	O'lchovlari bir xil massiv mos elementlari buyicha bo'lish
8.	.^	Massiv mos elementlari buyicha darajaga ko'tarish
9.	\	Matritsaviy chapdan ungga bo'lish
10.	.\	Massiv mos elementlari buyicha chapdan ungga bo'lish
11.	'	Qo'shma matritsani hisoblash
12.	.'	Transponerlash

21.5-rasm. Arifmetik amallar

MATLAB da matematik ifodalar ma'lum bir bajarilish tartibiga asosan bajariladi. Avval mantiqiy amallar, so'ngra arifmetik amallar: avval daraja, keyin ko'paytirish va bo'lish, undan keyin esa qo'shish va ayirish bajariladi. Agar ifodada qavslar bo'lsa, avval qavs ichidagi ifoda yuqoridagi tartibda bajariladi.

3) munosabat amallari:

T. R	Operator(sintaksis)	Amal belgilari aytilishi
1.	$= ; (x = y)$	Teng
2.	$\sim ; (x \sim y)$	Teng emas
3.	$< ; (x < y)$	Kichik
4.	$> ; (x > y)$	Katta
5.	$\leq ; (x \leq y)$	Kichik yoki teng
6.	$\geq ; (x \geq y)$	Katta yoki teng

4) mantiqiy amallar:

T. R	Operator(sintaksis)	Amal belgilari aytilishi
1.	$\&; \text{and} (\text{and} (a, b))$	va
2.	$; \text{or} (\text{or} (a, b))$	yoki
3.	$\sim ; \text{not} (\text{not} (a, b))$	inkor
4.	$\text{xor} (\text{xor} (a, b))$	
5.	$\text{any} (\text{any} (a))$	
6.	$\text{all} (\text{all} (a))$	

21.6-rasm. Munosabat va mantiqiy amallar.

Buyruqni kiritish uchun, Command windowsdan ($\gg$) ko'rsatkichiga sichqonchani olib borib belgilaymiz. Odatda, ikki xil ko'rsatkich mavjud:

$\gg$ To'liq ishlaydigan MATLAB dasturi uchun

EDU> Ta'lim uchun olingan MATLAB turi

(Eslatma: qaydni osonlashtirish uchun, $\gg$ ko'rsatkichini garchi MATLAB versiyasi ta'lim maqsadlarida bo'lishiga qaramasdan standart ko'rsatkich sifatida ishlatamiz.

22-ma'ruza. Matlab dasturida funksiya va grafik

Reja:

1. Matlab dasturida funksiya hamda operator tushunchisi, turlari va ulardan foydalanish.
2. MATLAB dasturida ikki va uch o'lchovli grafiklar qurish.
3. Chizilgan grafiklar va grafik oynalarni loyihalash va boshqarishda grafik oyna menyu va uskunalar paneli elementlari hamda buyruqlar orqali amalga oshirish.

Tayanch so'z va iboralar: Matlab, diagrammalar, vektorli, matrisa, funksiyalar masteri, modellash, algoritm.

MATLAB muhiti grafikasi elementlari bilan ishlash, 2 va 3 o'lchovli funksiyalar grafiklarini chizish ustida amallar bajarib amaliy ko'nikmalarni mustahkamlash mumkin.

Ikki o'lchovli grafika.

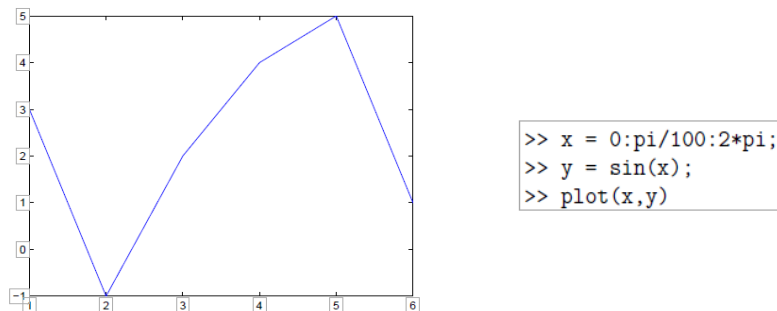
- MATLAB da ikki o'lchovli grafiklarni chizishda asosan quyidagi buyruqlardan foydalaniladi:

- loglog, polar, stairs, area, pcolor, line, pie, plot, semilogx, comet, bar, fill, colormap, ribbon, pie3, strips, semilogy, stem, barh, patch, rectangle, scatter, errorbar, imagesc va h.k.;

- Chizilgan grafiklar va grafik oynalarni loyihalash va boshqarishda grafik oyna menyu va uskunalar paneli elementlari hamda quyidagi buyruqlar orqali amalga oshiriladi:

- grid, axis, hold, figure, shg, clf, subplot va h.k.;

plot buyrug'ini ko'rib chiqamiz



22.1-rasm. Plot buyrug'i

Odatda bir o'zgaruvchili funksiya grafiklarini chizishda plot buyrug'i va uning turli xil ko'rinishlaridan foydalaniladi.

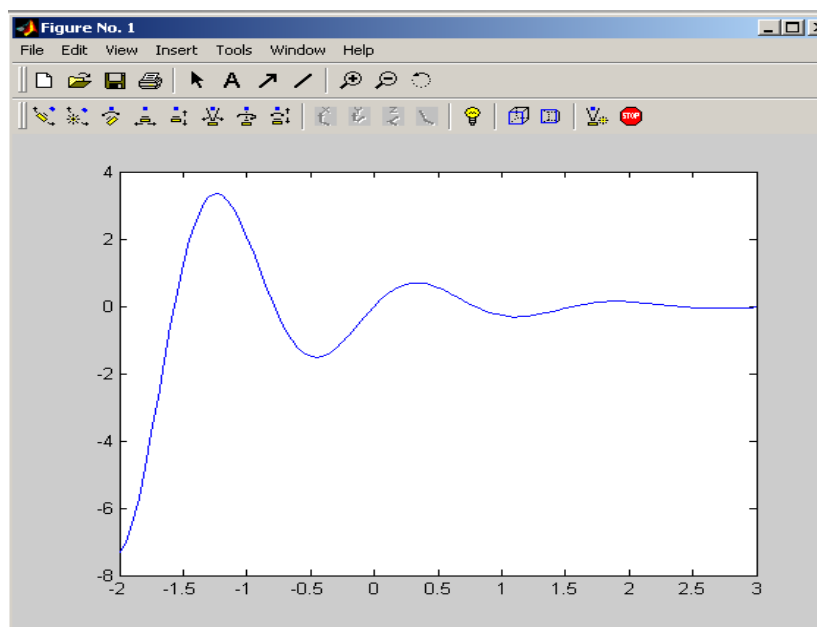
```
>> x=[a:h:b];  
>> y=f(x);  
>> plot(x,y)
```

Bajarilishi:

Bunda dastlab berilgan h qadam bilan bo'laklarga bo'lingan oraliq kiritiladi, keyin funksiya kiritiladi, undan keyin plot buyrug'i kiritiladi.

1-mashq.

- ◆ $y = e^{-x}\sin(4x)$ funksiya grafigini $[-2; 3]$ oraliqda chizamiz:
- ◆ `>> x=[-2:0.05:3];`
- ◆ `>> y=exp(-x).*sin(4*x);`
- ◆ `>> plot(x,y)`

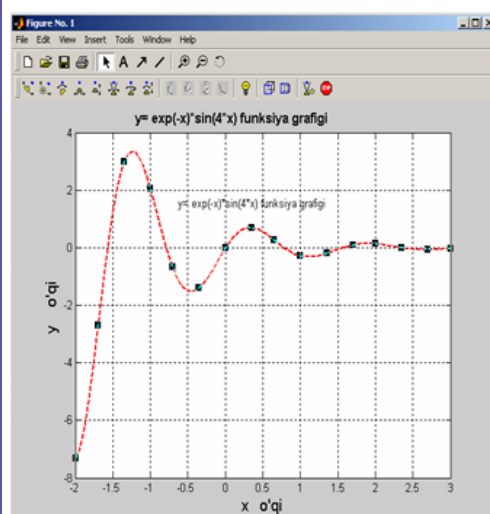


22.2-rasm.Funksiya grafigi.

CHIZMANI TAHRIRLASH

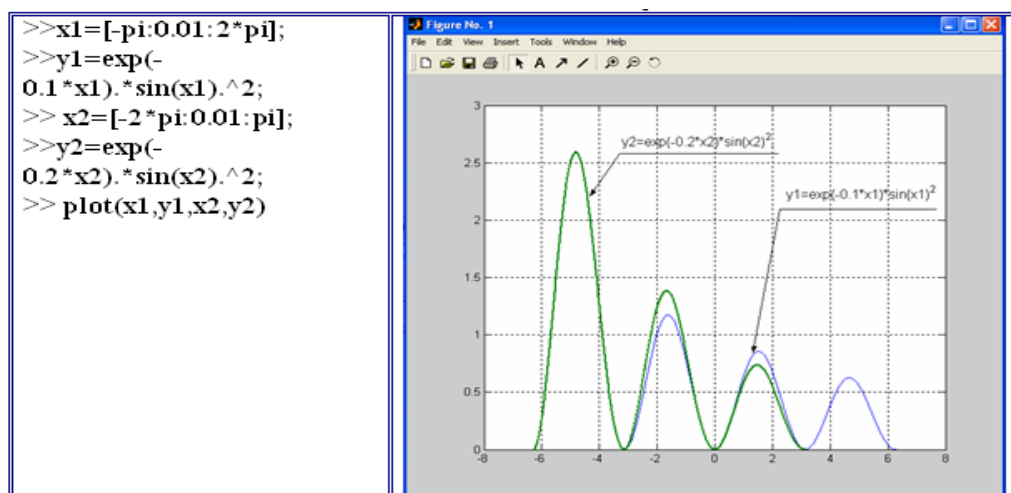
Chiziq rangi, tipi va stilini quyidagi jadvalda ko'rsatilgan belgilardan yoki grafik maydon imkoniyatlaridan foydalanib o'zgartirish mumkin:

Chiziq rangi		Chiziq nuqtalari tipi	
y	Sariq	▪	Nuqtali chiziq
m	Pushtirang	°	Aylanali
c	Havorang	x	Krest(xoch)
r	Qizil	+	Qo'shish belgisi
g	Yashil	*	Yulduzcha
b	Ko'k	s	Kvadrat
w	Oq	d	Romb
k	Qora	v	burchakli
	va hokazo	^	burchakli
Chiziq tipi		<	burchakli
—	Uzluksiz, to'liq	>	burchakli
:	Punktli	p	Besh yulduzli
.-	Shtrix-punktli	h	Olti yulduzli
--	Shtrixli		va hokazo



22.3-rasm.Funksiya grafigi.

2-mashq. Bitta grafik oynada bir necha grafik

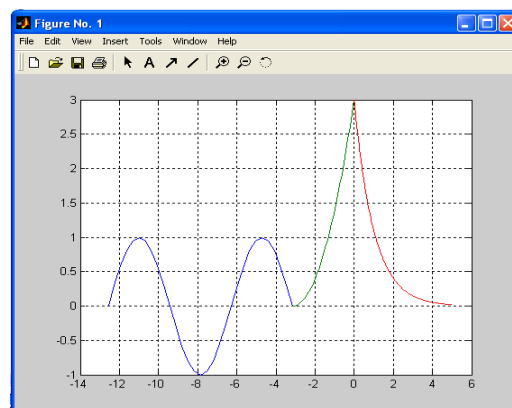


22.4-rasm.Funksiya grafigi.

3-mashq.

$$y(x) = \begin{cases} \sin x & -4\pi \leq x \leq -\pi \\ 3(x/\pi + 1)^2 & -\pi < x \leq 0 \\ 3e^{-x} & 0 < x \leq 5 \end{cases}$$

- ◆ >> x1=[-4*pi:pi/10:-pi];
- ◆ >> y1=sin(x1);
- ◆ >> x2=[-pi:pi/30:0];
- ◆ >> y2=3*(x2/pi+1).^2;
- ◆ >> x3=[0:0.02:5];
- ◆ >> y3=3*exp(-x3);
- ◆ >> plot(x1,y1,x2,y2,x3,y3)



22.5-rasm.Funksiya grafigi.

Plot buyrug`I uchun atributlar

SYMBOL	COLOR	SYMBOL	LINE STYLE	SYMBOL	MARKER
k	Black	—	Solid	+	Plus sign
r	Red	--	Dashed	o	Circle
b	Blue	:	Dotted	*	Asterisk
g	Green	-.	Dash-dot	.	Point
c	Cyan	none	No line	×	Cross
m	Magenta			s	Square
y	Yellow			d	Diamond

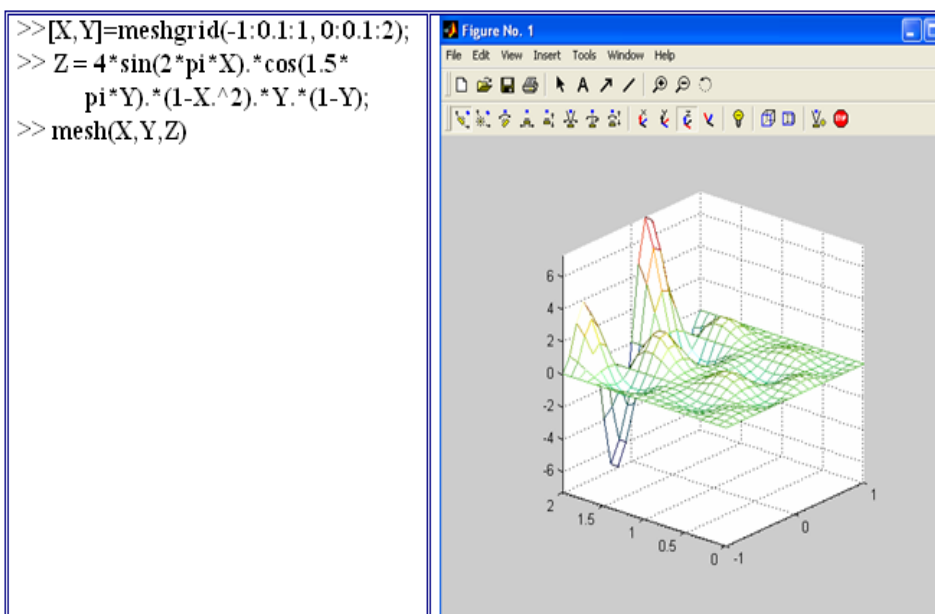
Uch o`lchovli grafika.

MATLAB da uch o'lchovli grafiklarni chizishda asosan quyidagi buyruqlardan foydalaniladi:

♦ bar3, plot3, mesh, surf, sphere, cylinder, bar3h, contour, meshgrid, fill3, ellipsoid, logo va h.k.;

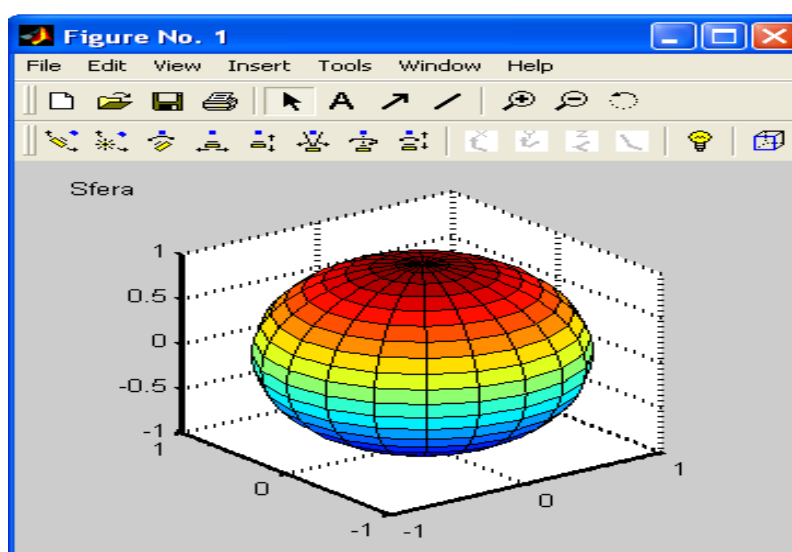
Mesh buyrug'i

Odatda ko'p o'zgaruvchili funksiya grafiklarini chizishda mesh buyrug'i va uning turli xil ko'rinishlaridan foydalaniladi. Misol:

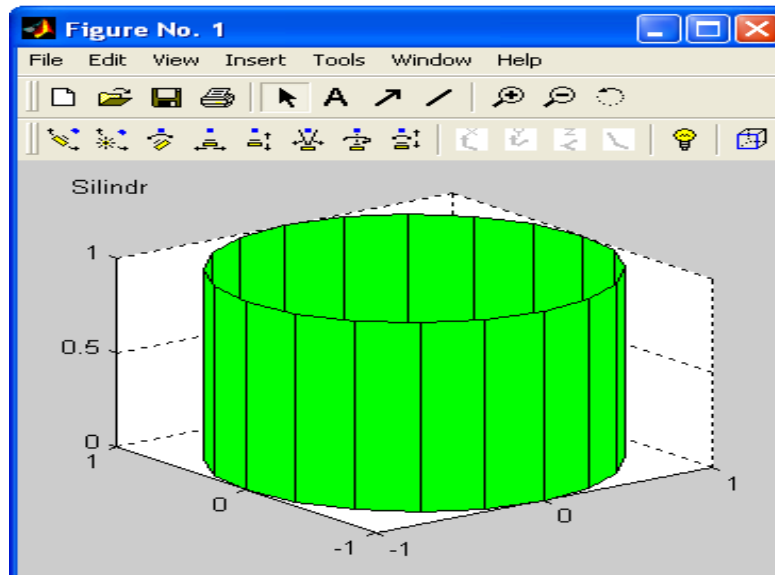


22.6-rasm.Mesh buyrug'i..

Fazoda geometrik jismlarni tasvirlashda sphere, cylinder, bar3h, ellipsoid kabi bir qator buyruqlardan foydalaniladi.



22.7-rasm. >>sphere buyrug'i.



22.8-rasm. >> cylinder buyrug'i..

Konstantalar jadvali

pi	The π number, $\pi = 3.14159\dots$
i, j	The imaginary unit i , $\sqrt{-1}$
Inf	The infinity, ∞
NaN	Not a number

Satr boshiga bir necha jadvallar qo'yish

```
>> a=7; b=cos(a), c=cosh(a)
```

```
b =
```

```
0.6570
```

```
c =
```

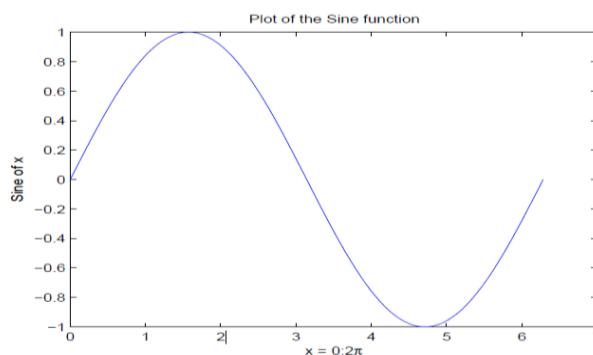
```
548.3170
```

Plot grafik funksiyasidan foydalanish

```
>> xlabel('x = 0:2\pi')
```

```
>> ylabel('Sine of x')
```

```
>> title('Plot of the Sine function')[A-6, 16 bet]
```

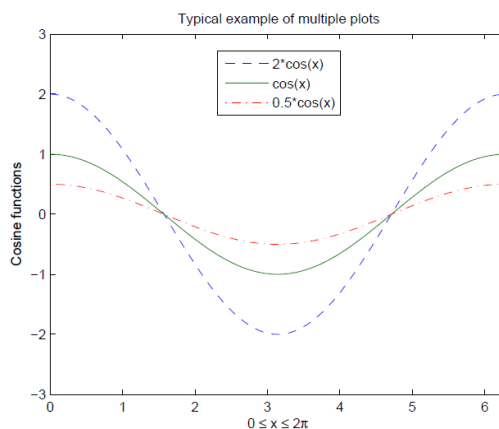


```

>> x = 0:pi/100:2*pi;
>> y1 = 2*cos(x);
>> y2 = cos(x);
>> y3 = 0.5*cos(x);
>> plot(x,y1,'--',x,y2,'-',x,y3,':')
>> xlabel('0 \leq x \leq 2\pi')
>> ylabel('Cosine functions')
>> legend('2*cos(x)','cos(x)','0.5*cos(x)')

>> title('Typical example of multiple plots')
>> axis([0 2*pi -3 3])

```



Matrisani kiritish uchun Bu yerda, odatda, bir misol keltirilgan.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Ko`rinishi

```
>> A = [1 2 3; 4 5 6; 7 8 9]
```

Savollari:

- 1.Qanday dasturlash tillarini bilasiz?
- 2.Matlab dasturini yuklash ketma-ketligini aytib bering.
- 3.Matlab dasturida nechta menyu bor?”
- 4.Matlab dasturining xarakteristikasini sanab bering.

23-ma'ruza. Matlab dasturida funksiya va grafik

Reja:

1. Matlab dasturida funksiya hamda operator tushunchisi, turlari va ulardan foydalanish.
2. MATLAB dasturida ikki va uch o'lchovli grafiklar qurish.
3. Chizilgan grafiklar va grafik oynalarni loyihalash va boshqarishda grafik oyna menyu va uskunalar paneli elementlari hamda buyruqlar orqali amalga oshirish.

Tayanch so'z va iboralar: Matlab, diagrammalar, vektorli, matrisa, funksiyalar masteri, modellash, algoritm.

MATLAB muhiti grafikasi elementlari bilan ishlash, 2 va 3 o'lchovli funksiyalar grafiklarini chizish ustida amallar bajarib amaliy ko'nikmalarni mustahkamlash mumkin.

Ikki o'lchovli grafika.

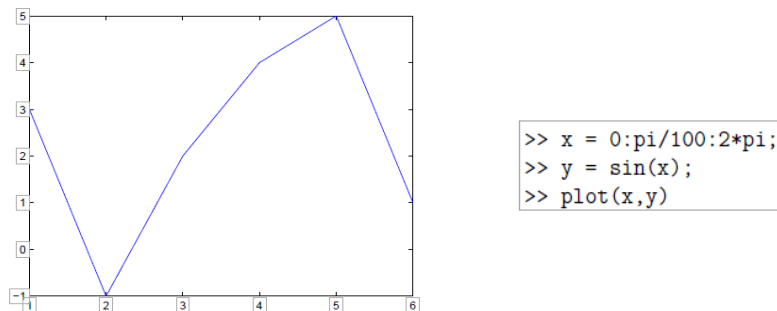
- MATLAB da ikki o'lchovli grafiklarni chizishda asosan quyidagi buyruqlardan foydalaniladi:

- loglog, polar, stairs, area, pcolor, line, pie, plot, semilogx, comet, bar, fill, colormap, ribbon, pie3, strips, semilogy, stem, barh, patch, rectangle, scatter, errorbar, imagesc va h.k.;

- Chizilgan grafiklar va grafik oynalarni loyihalash va boshqarishda grafik oyna menyu va uskunalar paneli elementlari hamda quyidagi buyruqlar orqali amalga oshiriladi:

- grid, axis, hold, figure, shg, clf, subplot va h.k.;

plot buyrug'ini ko'rib chiqamiz



23.1-rasm. Plot buyrug'i

Odatda bir o'zgaruvchili funksiya grafiklarini chizishda plot buyrug'i va uning turli xil ko'rinishlaridan foydalaniladi.

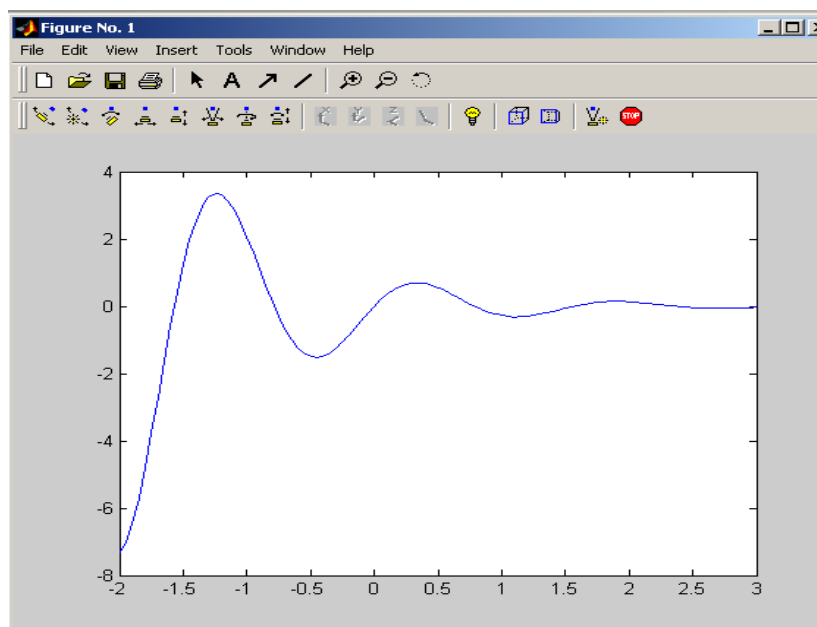
```
>> x=[a:h:b];  
>> y=f(x);  
>> plot(x,y)
```

Bajarilishi:

Bunda dastlab berilgan h qadam bilan bo'laklarga bo'lingan oraliq kiritiladi, keyin funksiya kiritiladi, undan keyin plot buyrug'i kiritiladi.

1-mashq.

- ◆ $y = e^{-x}\sin(4x)$ funksiya grafigini $[-2; 3]$ oraliqda chizamiz:
- ◆ `>> x=[-2:0.05:3];`
- ◆ `>> y=exp(-x).*sin(4*x);`
- ◆ `>> plot(x,y)`



23.2-rasm.Funksiya grafigi.

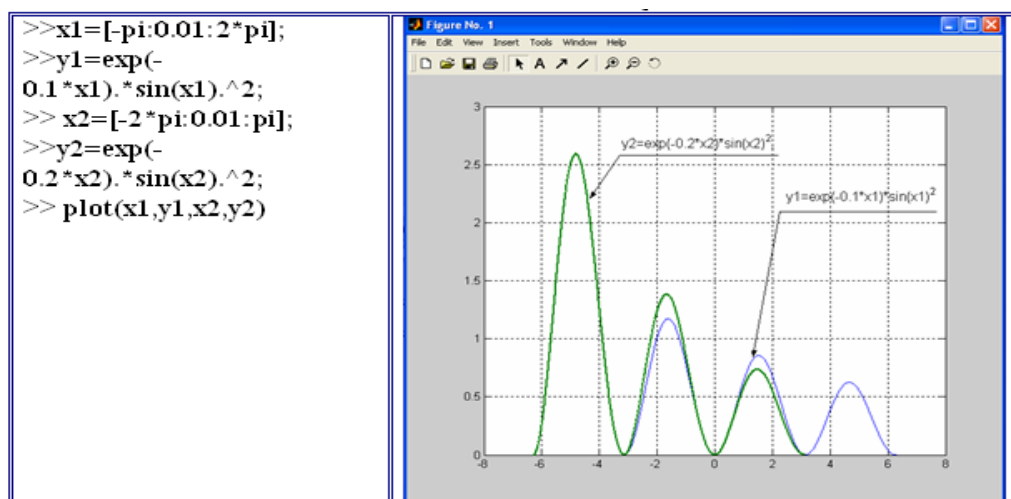
CHIZMANI TAHRIRLASH

Chiziq rangi, tipi va stilini quyidagi jadvalda ko'rsatilgan belgilardan yoki grafik maydon imkoniyatlaridan foydalanib o'zgartirish mumkin:

Chiziq rangi		Chiziq nuqtalari tipi	
y	Sariq	▪	Nuqtali chiziq
m	Pushtirang	°	Aylanali
c	Havorang	x	Krest(xoch)
r	Qizil	+	Qo'shish belgisi
g	Yashil	*	Yulduzcha
b	Ko'k	s	Kvadrat
w	Oq	d	Romb
k	Qora	v	burchakli
	va hokazo	^	burchakli
Chiziq tipi		<	burchakli
—	Uzluksiz, to'liq	>	burchakli
:	Punktli	p	Besh yulduzli
.-	Shtrix-punktli	h	Olti yulduzli
--	Shtrixli		va hokazo

23.3-rasm.Funksiya grafigi.

2-mashq. Bitta grafik oynada bir necha grafik

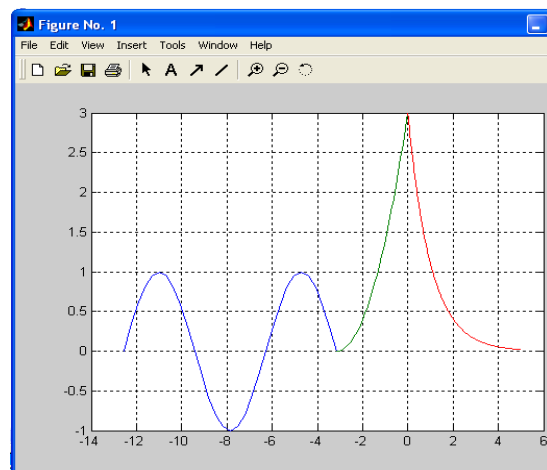


23.4-rasm.Funksiya grafigi.

3-mashq.

$$y(x) = \begin{cases} \sin x & -4\pi \leq x \leq -\pi \\ 3(x/\pi + 1)^2 & -\pi < x \leq 0 \\ 3e^{-x} & 0 < x \leq 5 \end{cases}$$

- ◆ >> x1=[-4*pi:pi/10:-pi];
- ◆ >> y1=sin(x1);
- ◆ >> x2=[-pi:pi/30:0];
- ◆ >> y2=3*(x2/pi+1).^2;
- ◆ >> x3=[0:0.02:5];
- ◆ >> y3=3*exp(-x3);
- ◆ >> plot(x1,y1,x2,y2,x3,y3)



23.5-rasm.Funksiya grafigi.

Plot buyrug`I uchun atributlar

SYMBOL	COLOR	SYMBOL	LINE STYLE	SYMBOL	MARKER
k	Black	—	Solid	+	Plus sign
r	Red	--	Dashed	o	Circle
b	Blue	:	Dotted	*	Asterisk
g	Green	-.	Dash-dot	.	Point
c	Cyan	none	No line	×	Cross
m	Magenta			s	Square
y	Yellow			d	Diamond

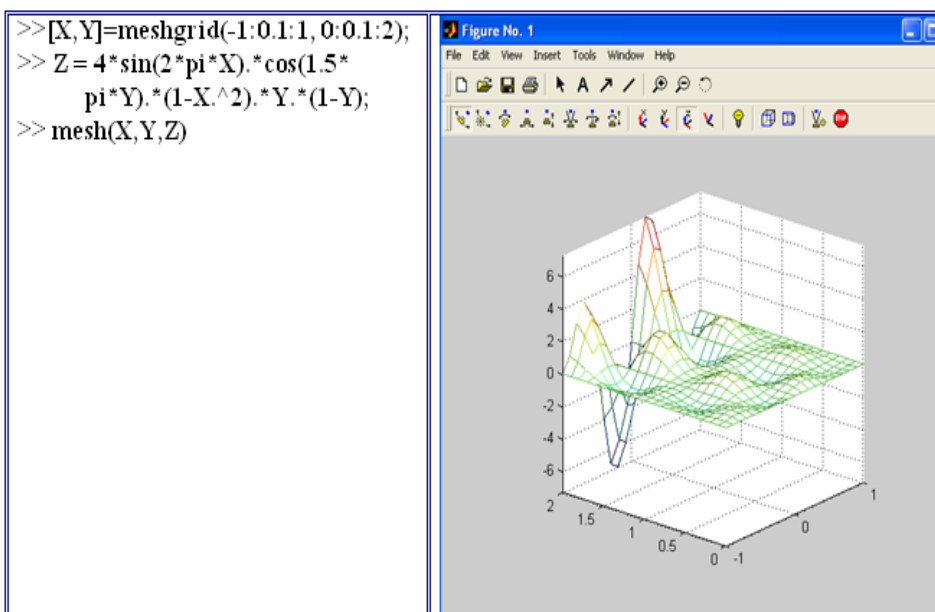
Uch o`lchovli grafika.

MATLAB da uch o'lchovli grafiklarni chizishda asosan quyidagi buyruqlardan foydalaniladi:

◆ bar3, plot3, mesh, surf, sphere, cylinder, bar3h, contour, meshgrid, fill3, ellipsoid, logo va h.k.;

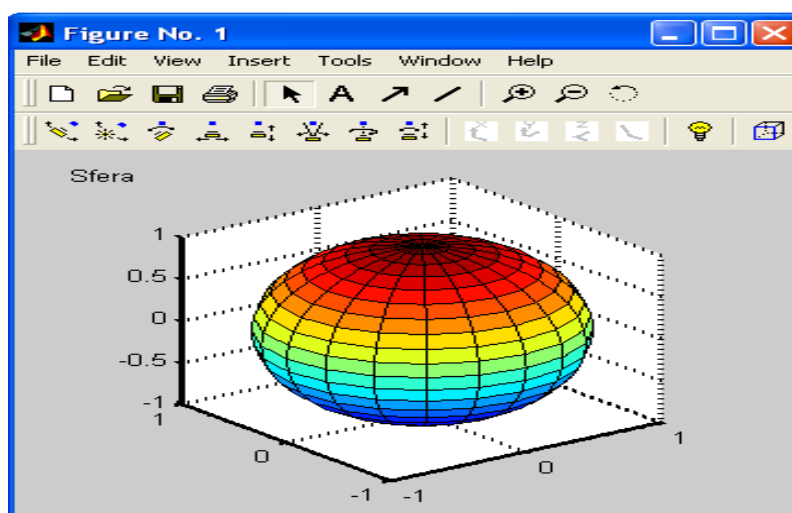
Mesh buyrug'i

Odatda ko'p o'zgaruvchili funksiya grafiklarini chizishda mesh buyrug'i va uning turli xil ko'rinishlaridan foydalaniladi. Misol:

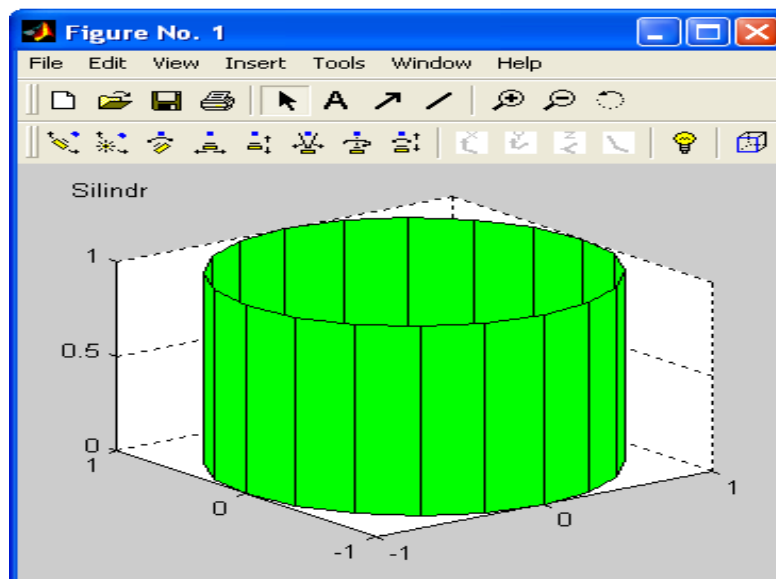


23.6-rasm.Mesh buyrug'i..

Fazoda geometrik jismlarni tasvirlashda sphere, cylinder, bar3h, ellipsoid kabi bir qator buyruqlardan foydalaniladi.



22.7-rasm. >>sphere buyrug'i



23.8-rasm. >> cylinder buyrug'i.
Konstantalar jadvali

pi	The π number, $\pi = 3.14159 \dots$
i, j	The imaginary unit i , $\sqrt{-1}$
Inf	The infinity, ∞
NaN	Not a number

Satr boshiga bir necha jadvallar qo'yish

```
>> a=7; b=cos(a), c=cosh(a)
```

```
b =
```

```
0.6570
```

```
c =
```

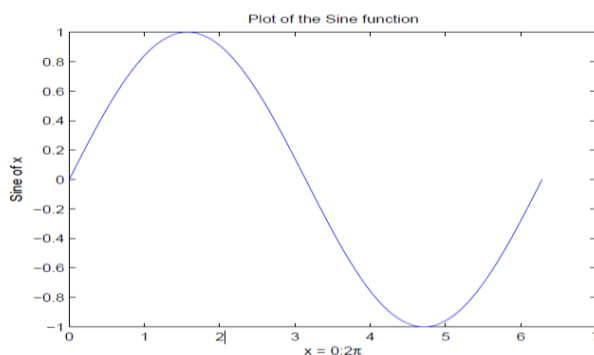
```
548.3170
```

Plot grafik funksiyasidan foydalanish

```
>> xlabel('x = 0:2\pi')
```

```
>> ylabel('Sine of x')
```

```
>> title('Plot of the Sine function')[A-6, 16 bet]
```

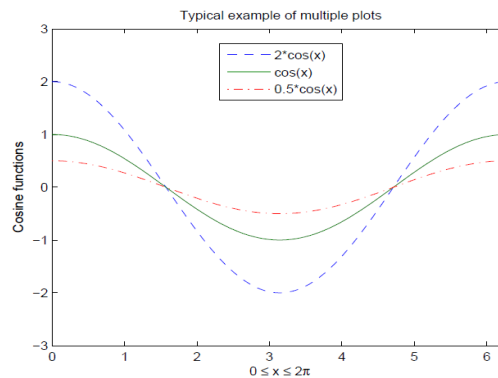


```

>> x = 0:pi/100:2*pi;
>> y1 = 2*cos(x);
>> y2 = cos(x);
>> y3 = 0.5*cos(x);
>> plot(x,y1,'--',x,y2,'-',x,y3,':')
>> xlabel('0 \leq x \leq 2\pi')
>> ylabel('Cosine functions')
>> legend('2*cos(x)','cos(x)','0.5*cos(x)')

>> title('Typical example of multiple plots')
>> axis([0 2*pi -3 3])

```



Matrisani kiritish uchun Bu yerda, odatda, bir misol keltirilgan.

$$A = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$$

Ko`rinishi

```
>> A = [1 2 3; 4 5 6; 7 8 9]
```

Savollari:

- 1.Qanday dasturlash tillarini bilasiz?
- 2.Matlab dasturini yuklash ketma-ketligini aytib bering.
- 3.Matlab dasturida nechta menyu bor”
- 4.Matlab dasturining xarakteristikasini sanab bering.

24-ma'ruza. Aspen amaliy dasturi haqida tushuncha.

Reja:

- 1. Aspen Plus asosiy tushunchalari va xususiyatlari.**
- 2. Texnologik jarayonlarni modellashtirishga kirish.**
- 3. Aspen Plus foydalanuvchi interfeysi.**
- 4. Asosiy ma'lumotlarni kiritish.**

Jarayonni modellashtirish bu -

Jarayonning moddiy va issiqlik balanslarini hisoblash uchun kompyuter dasturidan foydalanish.

Asosiy jismoniy munosabatlardan foydalanish:

- moddiy va issiqlik balanslari;
- fazalar muvozanati
- reaksiyalar va massa uzatish

Bashoratli hisoblash:

- oqim tezligi, tarkibi va xususiyatlari
- ish rejimlari

Simulyatsiya foydalari.

O'rnatishni loyihalash vaqtini qisqartiradi:

- Muhandisga turli xil o'rnatish konfiguratsiyalarini tezda sinab ko'rish imkonini beradi;
- Ishlatilgan o'rnatishlarni yaxshilashga yordam beradi;
- Savollarga "agar bo'lsa ..." javob beradi;
- Berilgan doiradagi optimal jarayon shartlarini aniqlaydi;
- Muammolarni bartaraf etishga yordam beradi.

Modellashtirish usullari. Serial modulli (SM):

- har bir qurilma alohida ketma-ketlikda ma'lum bir ketma-ketlikda hal qilinadi
- murakkab sxemalarni echishda foydalaniladi.

Tenglamaga yo'naltirilgan (EO):

- barcha tenglamalar bir vaqtning o'zida echiladi;
- yaxshi boshlang'ich taxmin qilish kerak;

Ikkita usulning kombinatsiyasi:

- Boshlang'ich, murakkab yechim uchun SM yondashuvidan foydalaning, so'ngra aniqroq yechim uchun EO yondashuvidan foydalaning;
- Dastlabki yechimni olish uchun SM yondashuvidan foydalaning, so'ngra modelni optimallashtirish yoki sozlash uchun EO yondashuvidan foydalaning.

- Jarayon modellarini qurish uchun foydali maslahatlar
Bloklarda murakkab texnologik jarayonlarni yarating:
bu muammolar yuzaga kelganda muammolarni bartaraf qilishni osonlashtiradi
Aspen Plus-da barcha jihozlarni simulyatsiya qilishning hojati yo'q.
Jarayon kiritish ma'lumotlarining aniqligi va izchilligini ta'minlash.
Natijalar nomuvofiq va haqiqiy emasligiga ishonch hosil qiling.

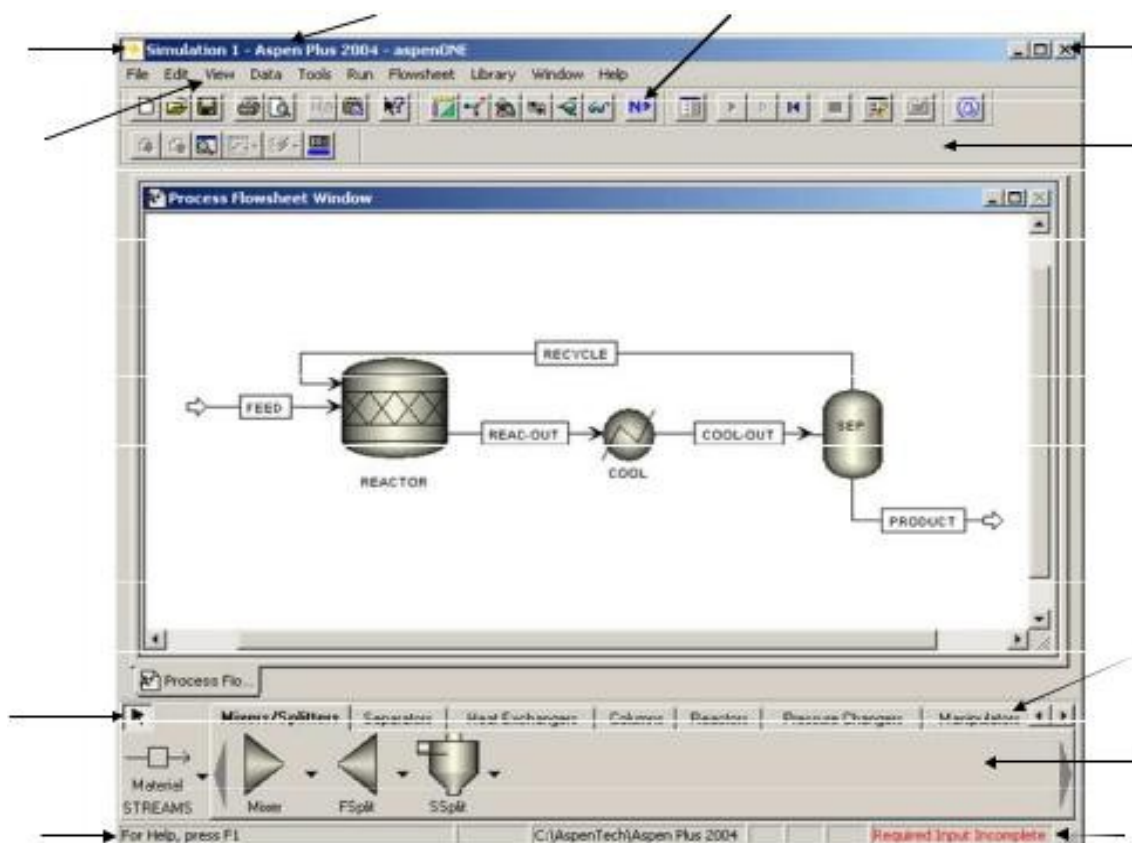
Aspen Plus-ning ba'zi muhim xususiyatlari.

- Elektrolitlarni aniq modelashtirish.
- Qattiq moddalar bilan ishlash.
- Ma'lumotlar regressiyasi.
- Ma'lumotni moslashtirish.
- Xususiyatlarni baholash.
- Foydalanuvchi hisoblari.

Onlayn optimallashtirish tizimlarida foydalanish uchun modelashtirish, optimallashtirish va ma'lumotlarni muvofiqlashtirishni o'z ichiga olgan tenglamaga yo'naltirilgan modelash.

1.2.Aspen Plus foydalanuvchi interfeysi.

Foydalanuvchi interfeysi



Ekranda quyidagilar aks etgan:

- 1.Idintifikator ishlash.
- 2.Sarlavha satri.
- 3.Menyu qatori.

4. "Keyingi" tugma.
5. Tugmalar o'lchamlarini o'zgartirish.
6. Panellar oynasi.
7. Blok sxemasi jarayoni.
8. Asboblarning paneli oynasi.
9. Modellar kutubxonasi xatcho'plari
10. Kutubxona modellari.
11. Qator.
12. Ko'rsatmalar.

Oqim va bloklarning avtomatik nomi

Ob'ektni yaratishda foydalanuvchi Aspen Plus yordamida ob'ekt nomini qo'lda yoki avtomatik ravishda kiritishi mumkin.

Nomlash parametrlarini o'zgartirish uchun:

1. Asboblarning menyusidagi Variantlarni tanlang;
2. Flowsheet jadvaliga o'ting;
3. Kerakli nomlash variantlarini belgilang va keraksizlarini olib tashlang.

Bloklash va oqim nomlari ko'rsatilishi yoki yashirin bo'lishi mumkin, ob'ektni tanlang, o'ng tugmani bosib va menyudan yashirish variantini tanlang.

Bloklar - Jarayon oqimining grafik operatsiyalari .

Jarayon sxemasiga blokni joylashtirish uchun:

1. Namunaviy model kutubxonasida kerakli xatcho'pni tanlang.
2. Texnologik ob'ektning modelini tanlang. Ochiladigan o'qni bosib va model uchun belgini tanlang.
3. Blokni joylashtirish uchun avval blok belgisini, so'ngra jarayon sxemasini bosib. Shuningdek, belgini bosishingiz va uni diagrammaga sudrab borishingiz mumkin.
4. Diagrammadagi bo'sh joyni yana bir marta bosib bilan siz shunga o'xshash boshqa blokni va boshqalarni qo'shishingiz mumkin.
5. Bo'sh maydonga bloklarni joylashtirish rejimidan chiqish uchun sichqonchaning o'ng tugmachasini bosib.

Oqimlar - Jarayon oqimining grafik operatsiyalari -

Oqimni jarayon jadvaliga o'rnatish uchun:

1. Modellar kutubxonasidagi "Oqimlar" (STREAMS) belgisini bosib.
2. Belgining yonidagi "pastga" tugmani bosib, oqim turini (material, energiya yoki ish) tanlang.
3. Ulanish uchun jihozning ajratilgan portini (kirish yoki chiqish) bosib.
4. Oqimning qolgan qismlarini ulash uchun 3-bosqichni takrorlang.
5. Oqimning uchidan birini texnologik jarayonning kirishiga yoki chiqishiga joylashtirish uchun kontaktlarning zanglashiga olib keladigan qismni bosib.
6. Oqimni yaratish rejimidan chiqish uchun sichqonchaning o'ng tugmachasini bosib.

Sichqoncha tugmalaridan foydalanish

chap tugma: - blok, oqim, ob'ekt yorlig'i yoki izohni tanlaydi;

o'ng tugma: - Tanlangan oqim, blok yoki diagramma uchun menyuga qo'ng'iroq qiladi;

Ikki marta tez-tez bosib -oqim yoki blok uchun ma'lumot kiritish formasini yoki oraliq oqim uchun natija shaklini ochadi;
yorliqli matn yoki izohlarni tahrir qiladi

Bloklar va oqimlarni o'zgartirish.

Ma'lumotlar brauzerida kiritish shakllari va natijalarini ko'rsatish uchun:

1.a) Sizni qiziqtirgan ob'ektni ikki marta bosib yoki (agar hisoblamagan bo'lsangiz), Blok yoki oqimni tanlang, o'ng tugmani bosib va menyu punktini tanlang Input...

2. Blok yoki oqim ko'rinishini o'zgartirish uchun:

3.Ob'ektni tanlang va sichqoncha chap tugmasini bosib yoki Ctrl+K tugmalarini bosib.

4. Belgilangan ob'ektning belgisini ustiga olib boring va sichqonchani o'ng tugmachasini bosib, ushbu ob'ektning menyusini chaqiring;

5. "Exchange Icon" menyusini tanlang.

O'z-o'zini modellashtirish vazifasi.

Benzolni olish uchun texnologik jarayonning grafik diagrammasini tuzish kerak. Bloklar uchun mos belgilarni tanlang. Oqimni tanlash uchun o'ng tugmachasini bosib, Qayta ulash joyi / Resurs manbasini tanlang va oqim tugashi / boshlanishini tanlangan blok portiga ulang, simulyatsiya tugagach, vazifani BENZENE.BKP sifatida saqlang.

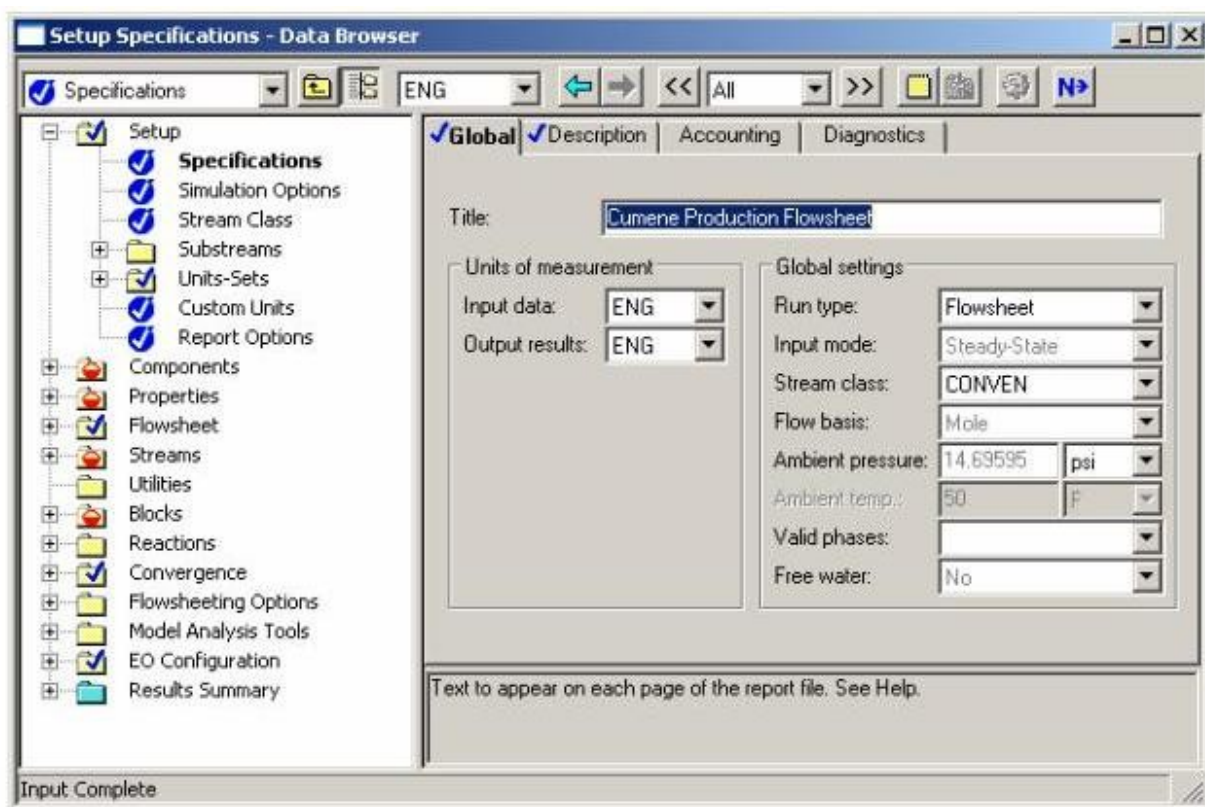
3.Asosiy ma'lumotlarni kiritish

Aspen Plus interfeysi dastur parametrlari va buyruqlarni belgilash uchun grafik interfeys menyusidan foydalanadi, boshqaruv panellari - eng tez-tez ishlatiladigan ba'zi funktsiyalarga kirish, modellar kutubxonasi - oqim jadvalini tuzish uchun ishlatiladi.

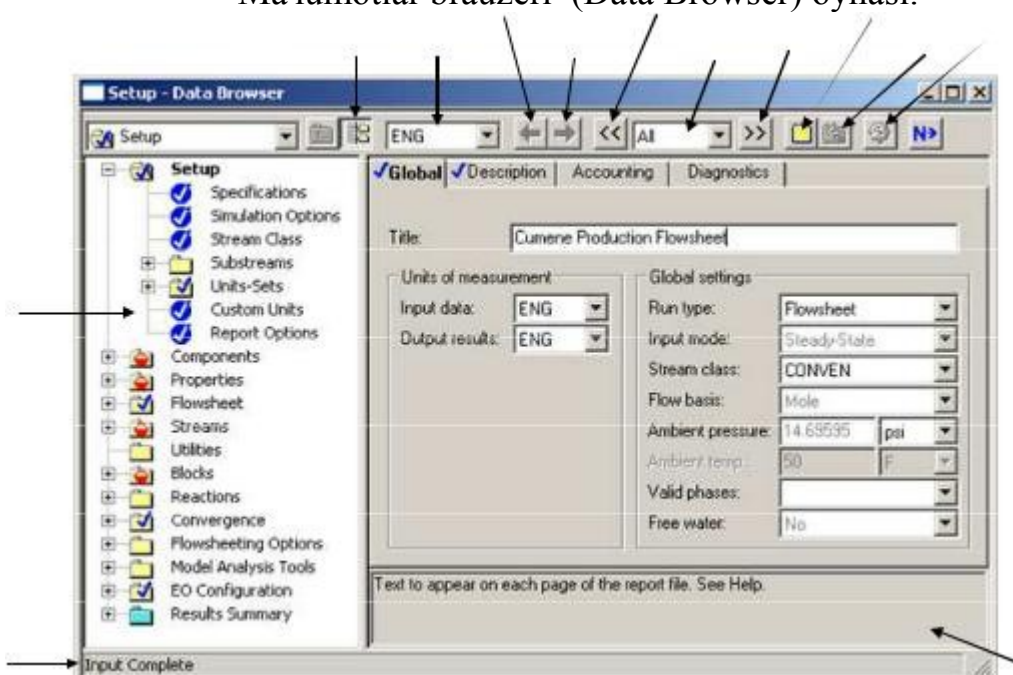
“Ma'lumotlar Brauzeri”(Data Browser) dan ma'lumotlar kiritish shakllari bilan ishlash uchun foydalaniladi, «Далее» tugmasi yordamida uni ko'chirish, hajmini o'zgartirish, ixchamlashtirish, ko'paytirish va yopish texnologik sxema ob'ektlarini boshqaruvchisiga ega, joriy forma to'ldirilganligini va ma'lumotlarni kiritish talab qilinadigan keyingi shaklga o'tishini tekshiradi

Ma'lumotlar brauzeri (Data Browser) - bu o'zgaruvchilarga kirish, hisoblash natijalari va ob'ektlarining ierarxik ravishda tuzilgan vakili. Undan hisoblash sozlamalari, oqimlari, bloklari, yordamchi dasturlari va boshqa shakllarini yo'naltirish uchun foydalanish mumkin.

Texnologik sxema bo'yicha ma'lumotlarni kiritish yoki natijalarni va (yoki) boshqa papkalarni, ob'ektlarni kiritish uchun shakllardan iborat



Ma'lumotlar brauzeri (Data Browser) oynasi.



Formadagi maydonni tanlash uchun sichqonchaning chap tugmachasini bosganingizda, ushbu maydon haqidagi ma'lumot tavsiflash maydonida paydo bo'ladi. Ushbu maydoncha ma'lumotlarni kiritishga yordam beradi.

Ushbu maydon uchun mumkin bo'lgan kiritish variantlarining ro'yxatini ko'rsatish uchun ochiladigan menyu o'qini bosib.

Maktubni bosish bosilgan xatdan boshlanib, ro'yxatning keyingi qismiga o'tadi

Tab tugmachasi kursorni formadagi keyingi maydonchaga o'tkazadi. Aspen Plus har doim jadvalning oxirgi qatoridan keyin bo'sh qator qo'shadi.

Yordam qo'ng'irog'i

Onlayn yordamni ishga tushirish uchun "Yordam" (Help) menyusidan "Yordam mavzularini" (Help Topics) tanlang:

Tarkibi. Hujjatlarni, shu jumladan foydalanuvchi qo'llanmalarini va qo'llanmalarini ko'rib chiqish.

Indeks. Indeks yozuvlari bo'yicha har qanday mavzu bo'yicha maslahatlarni qidirish.

Qidiruv. So'z yoki so'zlar to'plamini o'z ichiga olgan mavzular bo'yicha maslahatlarni qidiring

"What's This?" (что это?) "Bu nima?" Shoshilinch: (nima bu?)

"Bu nima?" asboblari panelida va ko'rsatmalar olishni istagan joyni tanlang.

F1 Maslahat. Maydonga kursorni joylashtiring va <F1> tugmachasini bosib va ushbu maydon va (yoki) forma uchun maslahatni chaqiring.

Ma'lumot kiritish.

Modelni hisoblash uchun quyidagi ma'lumotlarni kiritishingiz kerak:

- Настройка расчёта (Setup) - Hisoblashni sozlash;
- Компоненты (Components) – Komponentlar;
- Свойства (Properties) – Xususiyatlari;
- Потоки (Streams) – Oqimlar;
- Блоки (Blocks) - Bloklar.

«Keyingi» (« далее ») tugmasini bosish orqali ma'lumotlarni yuqoridagi tartibda kiriting. Siz shuningdek, ushbu kirish shakllarini Ma'lumotlar (Data) menyusini orqali yoki asboblari panelidagi ma'lumotlar brauzerining tugmachalarini bosish orqali tezda ochishingiz mumkin.

Holat ko'rsatkichlari. Ranglar va shakllar ma'lumotlar va natijalarning joriy holatini tavsiflaydi:

Belgi	Holat
	Shakl to'liq kiritilmagan
	Shakl to'liq kiritilgan
	Formaga hech narsa kiritilmagan. Kirish majburiy emas
	Formada kirish va chiqish ma'lumotlari mavjud
	Formada natijalar mavjud
	Formada natijalar mavjud, ammo hisob-kitoblarda ogohlantirishlar mavjud edi
	Formada natijalar mavjud, ammo hisob-kitoblarda ogohlantirishlar mavjud edi
	Formada natijalar mavjud, ammo men kiritishni hisoblab chiqqandan so'ng, men ma'lumotni o'zgartirdim
	

Hisoblashni sozlash.

Ko'pgina sozlamalar O'rnatish> Xususiyatlar> Global shaklda kiritilgan (Setup>Specifications>Global).

Hisobotlarda ishlatiladigan jadvalning nomi.

Hisoblash turi.

Kirish va chiqish miqdorlarining birliklari.

Mumkin bo'lgan fazalar (ya'ni bug'-suyuq yoki bug 'suyuq-suyuqlik).

Bosim va atrof-muhit harorati.

Oqim bo'yicha hisoblash natijalarini chiqarish parametrlari Setup> Hisobot parametrlari> Oqim (Setup> Report Options>Stream) shaklida.

Hisoblash sozlamalari vazifa shakli.

Hisoblash turini tanlash (O'rnatish (Setup) > Xususiyatlar(Specifications)> (Run type) Run turi maydonidagi global shakl.

Bajarish turi	Ta'rif
Flowsheet	Jarayonlar oqimining standart sxemasi mulkni baholash, test ma'lumotlarini tahlil qilish va / yoki mulk tahlilini hisoblashni o'z ichiga olishi mumkin
Assay Data	Quyida ko'rsatilgandek, ushbu parametrlarni jarayon oqimining diagrammasisiz alohida saqlash mumkin.
Analysis	Aspen Plus jarayonlari sinov ma'lumotlarini tahlil qilish va texnologik sxemalarda keyinchalik foydalanish uchun soxta tarkibiy qismlarni yaratish uchun Assay Data Analysis-dan foydalaning.
Data Regression	Jismoniy mulk modelini tanlash uchun “Ma'lumot Regression” turidan foydalaning. Aspen Plus aralashmaning sof tarkibiy qismlarini, bug '-suyuqlik balansini, suyuqlikni ajratib olish va boshqa parametrlarni aniqlash uchun zarur. Ma'lumot turida regressda mulkning reytingi va mulkni tahlilini hisoblash ham ko'rsatilishi mumkin.
Properties Plus	Aspen Custom Modeler yoki Aspen Pinch bilan foydalanish uchun mulk paketini, uchinchi tomonning tijorat muhandislik dasturlari yoki kompaniyangizning shaxsiy ishlanmalarini tayyorlash va Properties Plus turidan foydalanish kerak.
Property Analysis	Xususiyatlar jadvallari, diapazonlar yaratish uchun, harorat va bosim, suyuqlikning qoldiq fraksiyalari egri chizig'i va boshqalarida “Mulkni tahlil” qilishdan foydalaning. “ Mulkni tahlil” qilish test ma'lumotlarini analitik hisoblash, mulkni baholashni o'z ichiga olishi mumkin.
Property Estimation	Mulk parametrlarini baholash uchun mulkni baholashni xohlamasangiz, xuddi shu qismda jarayonni simulyatsiya qilishdan foydalaning.
Assay Data Analysis	Sinov ma'lumotlarini tahlil qilish, keyinchalik texnologik diagrammalarda foydalanish, Aspen Plus jarayonlari uchun soxta tarkibiy qismlar yaratish uchun Assay Data Analysis-dan foydalaning

Birlik sozlamalari. Aspen Plus-dagi bo'linmalarni uch xil darajada aniqlash mumkin:

1. Global daraja (Eslatmaning "Kirish ma'lumotlari" ("Input Data") va "Chiqish natijalari" ("Output Results") satrlari Sozlash> Xususiyatlar> Global).

2. Ob'ektlar darajasi (ob'ektning har bir shakli tepasida (blok yoki oqim) ustidagi "birliklar" (birlik)).

3. Maxsus maydonlar darajasi

4. Setup> Units-Sets sahifasini ochib, foydalanuvchilar o'zlarining shaxsiy to'plamlarini yaratishlari mumkin. O'lchov birliklari mavjud to'plamlar asosida yaratilishi mumkin va keyin ularni o'zgartiradi.

Komponentlar (oddiy moddalar). Modelda zarur bo'lgan barcha tarkibiy qismlarni aniqlash uchun Komponent> Xususiyatlar (Components>Specifications) shaklidan foydalaning. Agar imkon bo'lsa, har bir komponentning fizik xususiyatlari ma'lumotlar bazasidan olinadi. **Simple-** Oddiy moddalarning ma'lumotlar bazalarida molekulyar og'irlik, muhim xususiyatlar va boshqalar kabi parametrlar mavjud. Ma'lumotlar banklarida qidirish tartibi "Databanks" yorlig'ida aniqlanadi.

Find -tugmachasini ro'yxatga tarkibiy qismlarni qidirish va qo'shish uchun ishlatishingiz mumkin.

Electrolyte Wizard - elektrolitlar tarkibiy qismlarini qo'shish uchun sehgardan foydalanishingiz mumkin.

Komponent parametrlari shakli. Komponentlarni tanlash.

Komponent identifikatori kirish ma'lumotlarini kiritishda va simulyatsiya natijalarini ko'rsatishda komponent nomini ko'rsatish uchun ishlatiladi.

Komponent nomlarini to'g'ridan-to'g'ri kiritish quyidagi usullardan biri bilan amalga oshiriladi:

Formula: Komponentning kimyoviy formulasiga muvofiq (masalan, C_6H_6).
Izomerlar uchun quyidagicha yozing: C_2H_6O-2

Component Name: Ingliz tilidagi komponentning to'liq nomi bilan (masalan, BENZENE - benzol).

Find "Topish" tugmasi ma'lumotlar banklarida tarkibiy qismlarni qidirish oynasini ochadi.

Ma'lumotlar bazalarida tarkibiy qismlarni qidirish. Komponent nomi, formulasi, komponentlar sinfi, molekulyar og'irligi, qaynash nuqtasi yoki CAS raqami bo'yicha qidiruv.

Belgilangan shartlarga javob beradigan barcha tarkibiy qismlar topiladi:

Agar bir nechta mezonlar ko'rsatilgan bo'lsa, qidirish "AND" mantiqiy operatsiyasi yordamida amalga oshiriladi.

25-ma'ruza. Aspen amaliy dasturi haqida tushuncha.

Reja:

- 1. Aspen Plus asosiy tushunchalari va xususiyatlari.**
- 2. Texnologik jarayonlarni modellashtirishga kirish.**
- 3. Aspen Plus foydalanuvchi interfeysi.**
- 4. Asosiy ma'lumotlarni kiritish.**

Jarayonni modellashtirish bu -

Jarayonning moddiy va issiqlik balanslarini hisoblash uchun kompyuter dasturidan foydalanish.

Asosiy jismoniy munosabatlardan foydalanish:

- moddiy va issiqlik balanslari;
- fazalar muvozanati
- reaksiyalar va massa uzatish

Bashoratli hisoblash:

- oqim tezligi, tarkibi va xususiyatlari
- ish rejimlari

Simulyatsiya foydalari.

O'rnatishni loyihalash vaqtini qisqartiradi:

- Muhandisga turli xil o'rnatish konfiguratsiyalarini tezda sinab ko'rish imkonini beradi;
- Ishlatilgan o'rnatishlarni yaxshilashga yordam beradi;
- Savollarga "agar bo'lsa ..." javob beradi;
- Berilgan doiradagi optimal jarayon shartlarini aniqlaydi;
- Muammolarni bartaraf etishga yordam beradi.

Modellashtirish usullari. Serial modulli (SM):

- har bir qurilma alohida ketma-ketlikda ma'lum bir ketma-ketlikda hal qilinadi
- murakkab sxemalarni echishda foydalaniladi.

Tenglamaga yo'naltirilgan (EO):

- barcha tenglamalar bir vaqtning o'zida echiladi;
- yaxshi boshlang'ich taxmin qilish kerak;

Ikkita usulning kombinatsiyasi:

- Boshlang'ich, murakkab yechim uchun SM yondashuvidan foydalaning, so'ngra aniqroq yechim uchun EO yondashuvidan foydalaning;
- Dastlabki yechimni olish uchun SM yondashuvidan foydalaning, so'ngra modelni optimallashtirish yoki sozlash uchun EO yondashuvidan foydalaning.

- Jarayon modellarini qurish uchun foydali maslahatlar

Bloklarda murakkab texnologik jarayonlarni yarating:
 bu muammolar yuzaga kelganda muammolarni bartaraf qilishni osonlashtiradi
 Aspen Plus-da barcha jihozlarni simulyatsiya qilishning hojati yo'q.
 Jarayon kiritish ma'lumotlarining aniqligi va izchilligini ta'minlash.
 Natijalar nomuvofiq va haqiqiy emasligiga ishonch hosil qiling.

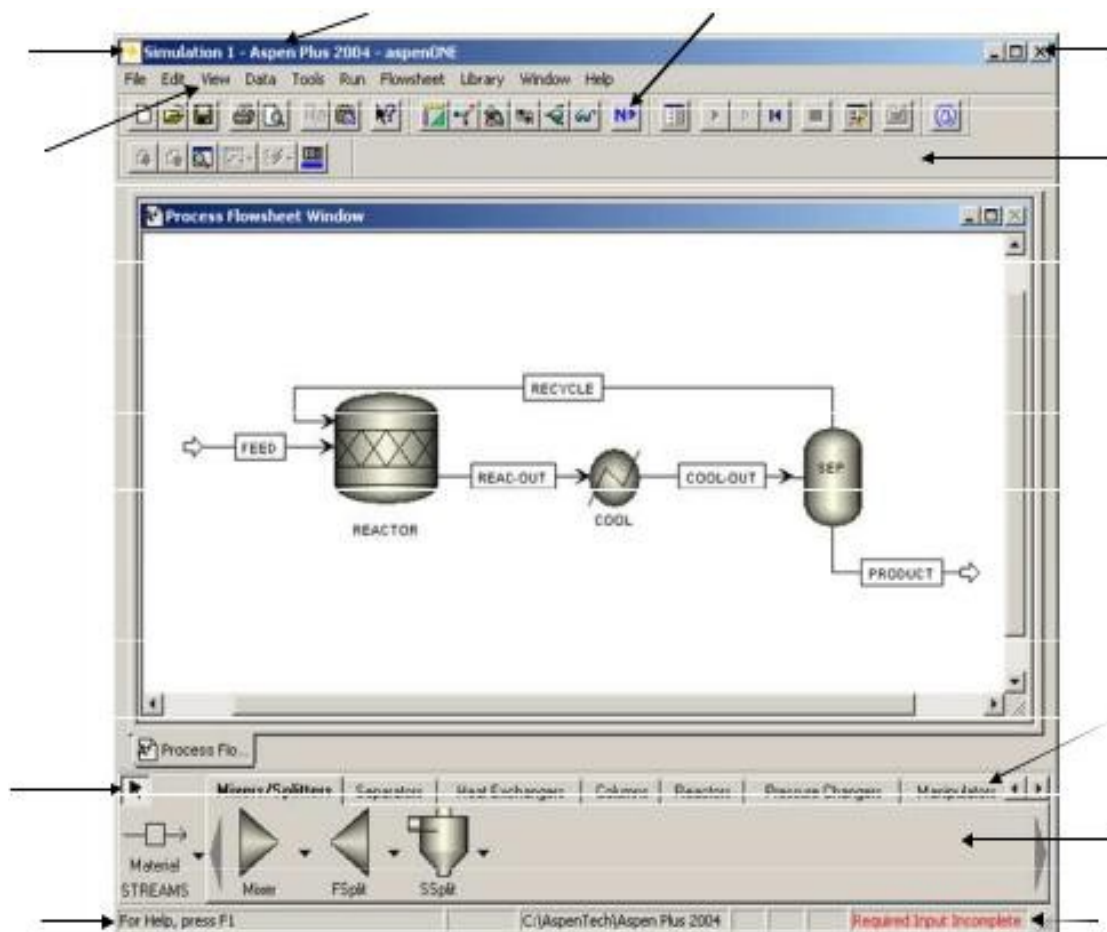
Aspen Plus-ning ba'zi muhim xususiyatlari.

- Elektrolitlarni aniq modellashtirish.
- Qattiq moddalar bilan ishlash.
- Ma'lumotlar regressiyasi.
- Ma'lumotni moslashtirish.
- Xususiyatlarni baholash.
- Foydalanuvchi hisoblari.

Onlayn optimallashtirish tizimlarida foydalanish uchun modellashtirish, optimallashtirish va ma'lumotlarni muvofiqlashtirishni o'z ichiga olgan tenglamaga yo'naltirilgan modellash.

1.2. Aspen Plus foydalanuvchi interfeysi.

Foydalanuvchi interfeysi



Ekranda quyidagilar aks etgan:

1. Idintifikator ishlash.
2. Sarlavha satri.
3. Menyu qatori.

4. "Keyingi" tugma.
5. Tugmalar o'lchamlarini o'zgartirish.
6. Panellar oynasi.
7. Blok sxemasi jarayoni.
8. Asboblarning paneli oynasi.
9. Modellar kutubxonasi xatcho'plari
10. Kutubxona modellari.
11. Qator.
12. Ko'rsatmalar.

Oqim va bloklarning avtomatik nomi

Ob'ektni yaratishda foydalanuvchi Aspen Plus yordamida ob'ekt nomini qo'lda yoki avtomatik ravishda kiritishi mumkin.

Nomlash parametrlarini o'zgartirish uchun:

1. Asboblarning menyusidagi Variantlarni tanlang;
2. Flowsheet jadvaliga o'ting;
3. Kerakli nomlash variantlarini belgilang va keraksizlarini olib tashlang.

Bloklash va oqim nomlari ko'rsatilishi yoki yashirin bo'lishi mumkin, ob'ektni tanlang, o'ng tugmani bosib va menyudan yashirish variantini tanlang.

Bloklar - Jarayon oqimining grafik operatsiyalari .

Jarayon sxemasiga blokni joylashtirish uchun:

1. Namunaviy model kutubxonasida kerakli xatcho'pni tanlang.
2. Texnologik ob'ektning modelini tanlang. Ochiladigan o'qni bosib va model uchun belgini tanlang.
3. Blokni joylashtirish uchun avval blok belgisini, so'ngra jarayon sxemasini bosib. Shuningdek, belgini bosishingiz va uni diagrammaga sudrab borishingiz mumkin.
4. Diagrammadagi bo'sh joyni yana bir marta bosib bilan siz shunga o'xshash boshqa blokni va boshqalarni qo'shishingiz mumkin.
5. Bo'sh maydonga bloklarni joylashtirish rejimidan chiqish uchun sichqonchaning o'ng tugmachasini bosib.

Oqimlar - Jarayon oqimining grafik operatsiyalari -

Oqimni jarayon jadvaliga o'rnatish uchun:

1. Modellar kutubxonasidagi "Oqimlar" (STREAMS) belgisini bosib.
2. Belgining yonidagi "pastga" tugmani bosib, oqim turini (material, energiya yoki ish) tanlang.
3. Ulanish uchun jihozning ajratilgan portini (kirish yoki chiqish) bosib.
4. Oqimning qolgan qismlarini ulash uchun 3-bosqichni takrorlang.
5. Oqimning uchidan birini texnologik jarayonning kirishiga yoki chiqishiga joylashtirish uchun kontaktlarning zanglashiga olib keladigan qismni bosib.
6. Oqimni yaratish rejimidan chiqish uchun sichqonchaning o'ng tugmachasini bosib.

Sichqoncha tugmalaridan foydalanish

chap tugma: - blok, oqim, ob'ekt yorlig'i yoki izohni tanlaydi;

o'ng tugma: - Tanlangan oqim, blok yoki diagramma uchun menyuga qo'ng'iroq qiladi;

Ikki marta tez-tez bosib -oqim yoki blok uchun ma'lumot kiritish formasini yoki oraliq oqim uchun natija shaklini ochadi;
yorliqli matn yoki izohlarni tahrir qiladi

Bloklar va oqimlarni o'zgartirish.

Ma'lumotlar brauzerida kiritish shakllari va natijalarini ko'rsatish uchun:

1.a) Sizni qiziqtirgan ob'ektni ikki marta bosib yoki (agar hisoblamagan bo'lsangiz), Blok yoki oqimni tanlang, o'ng tugmani bosib va menyu punktini tanlang Input...

2. Blok yoki oqim ko'rinishini o'zgartirish uchun:

3.Ob'ektni tanlang va sichqoncha chap tugmasini bosib yoki Ctrl+K tugmalarini bosib.

4. Belgilangan ob'ektning belgisini ustiga olib boring va sichqonchaning o'ng tugmachasini bosib, ushbu ob'ektning menyusini chaqiring;

5. "Exchange Icon" menyusini tanlang.

O'z-o'zini modellashtirish vazifasi.

Benzolni olish uchun texnologik jarayonning grafik diagrammasini tuzish kerak. Bloklar uchun mos belgilarni tanlang. Oqimni tanlash uchun o'ng tugmachasini bosib, Qayta ulash joyi / Resurs manbasini tanlang va oqim tugashi / boshlanishini tanlangan blok portiga ulang, simulyatsiya tugagach, vazifani BENZENE.BKP sifatida saqlang.

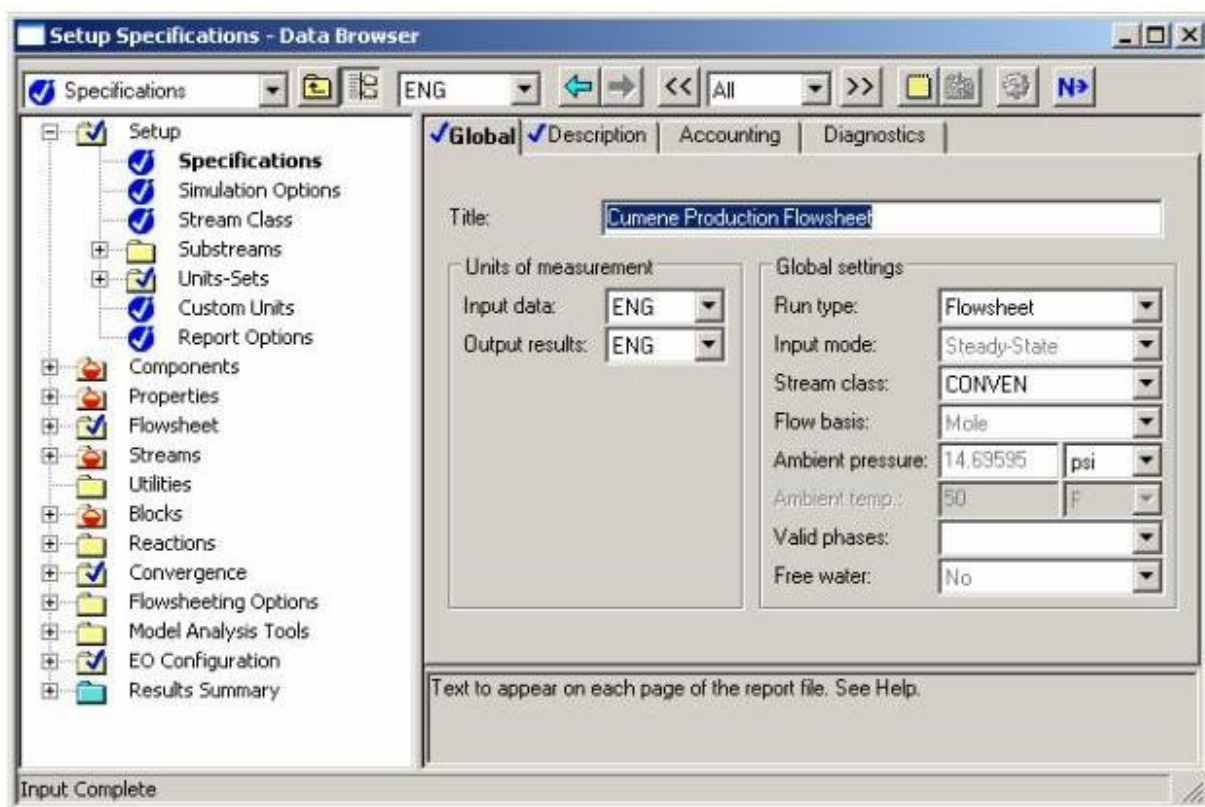
3.Asosiy ma'lumotlarni kiritish

Aspen Plus interfeysi dastur parametrlari va buyruqlarni belgilash uchun grafik interfeys menyusidan foydalanadi, boshqaruv panellari - eng tez-tez ishlatiladigan ba'zi funktsiyalarga kirish, modellar kutubxonasi - oqim jadvalini tuzish uchun ishlatiladi.

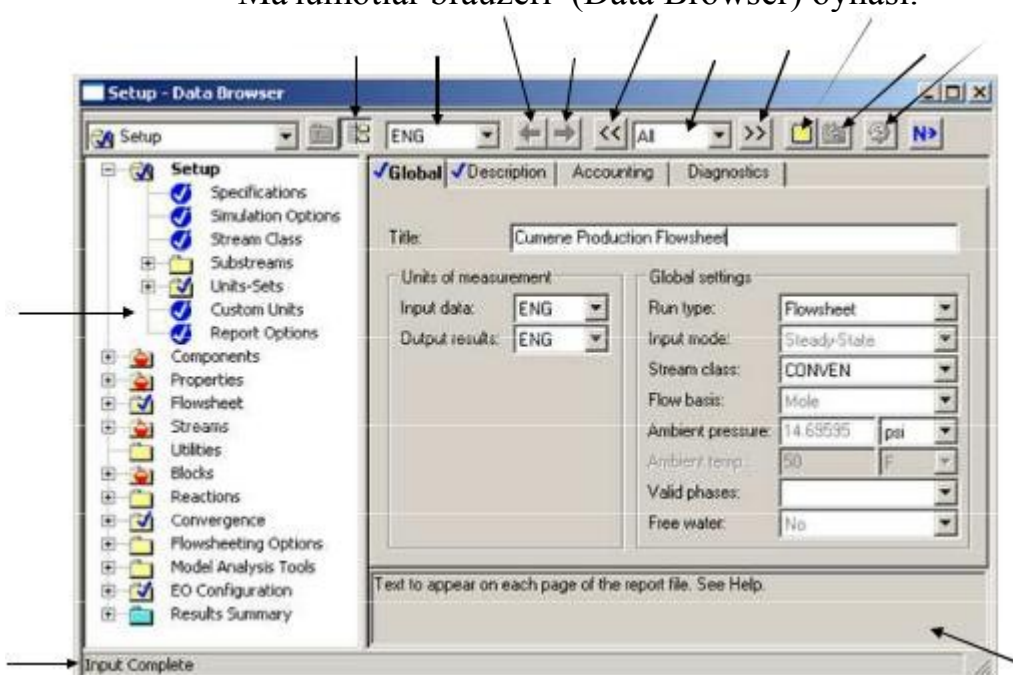
“Ma'lumotlar Brauzeri”(Data Browser) dan ma'lumotlar kiritish shakllari bilan ishlash uchun foydalaniladi, «Далее» tugmasi yordamida uni ko'chirish, hajmini o'zgartirish, ixchamlashtirish, ko'paytirish va yopish texnologik sxema ob'ektlarini boshqaruvchisiga ega, joriy forma to'ldirilganligini va ma'lumotlarni kiritish talab qilinadigan keyingi shaklga o'tishini tekshiradi

Ma'lumotlar brauzeri (Data Browser) - bu o'zgaruvchilarga kirish, hisoblash natijalari va ob'ektlarining ierarxik ravishda tuzilgan vakili. Undan hisoblash sozlamalari, oqimlari, bloklari, yordamchi dasturlari va boshqa shakllarini yo'naltirish uchun foydalanish mumkin.

Texnologik sxema bo'yicha ma'lumotlarni kiritish yoki natijalarni va (yoki) boshqa papkalarni, ob'ektlarni kiritish uchun shakllardan iborat



Ma'lumotlar brauzeri (Data Browser) oynasi.



Formadagi maydonni tanlash uchun sichqonchaning chap tugmachasini bosganingizda, ushbu maydon haqidagi ma'lumot tavsiflash maydonida paydo bo'ladi. Ushbu maydoncha ma'lumotlarni kiritishga yordam beradi.

Ushbu maydon uchun mumkin bo'lgan kiritish variantlarining ro'yxatini ko'rsatish uchun ochiladigan menyu o'qini bosing.

Maktubni bosish bosilgan xatdan boshlanib, ro'yxatning keyingi qismiga o'tadi

Tab tugmachasi kursorni formadagi keyingi maydonchaga o'tkazadi. Aspen Plus har doim jadvalning oxirgi qatoridan keyin bo'sh qator qo'shadi.

Yordam qo'ng'irog'i

Onlayn yordamni ishga tushirish uchun "Yordam" (Help) menyusidan "Yordam mavzularini" (Help Topics) tanlang:

Tarkibi. Hujjatlarni, shu jumladan foydalanuvchi qo'llanmalarini va qo'llanmalarini ko'rib chiqish.

Indeks. Indeks yozuvlari bo'yicha har qanday mavzu bo'yicha maslahatlarni qidirish.

Qidiruv. So'z yoki so'zlar to'plamini o'z ichiga olgan mavzular bo'yicha maslahatlarni qidiring

"What's This?" (что это?) "Bu nima?" Shoshilinch: (nima bu?)

"Bu nima?" asboblari panelida va ko'rsatmalar olishni istagan joyni tanlang.

F1 Maslahat. Maydonga kursorni joylashtiring va <F1> tugmachasini bosib va ushbu maydon va (yoki) forma uchun maslahatni chaqiring.

Ma'lumot kiritish.

Modelni hisoblash uchun quyidagi ma'lumotlarni kiritishingiz kerak:

- Настройка расчёта (Setup) - Hisoblashni sozlash;
- Компоненты (Components) – Komponentlar;
- Свойства (Properties) – Xususiyatlari;
- Потоки (Streams) – Oqimlar;
- Блоки (Blocks) - Bloklar.

«Keyingi» (« далее ») tugmasini bosish orqali ma'lumotlarni yuqoridagi tartibda kiriting. Siz shuningdek, ushbu kirish shakllarini Ma'lumotlar (Data) menyusini orqali yoki asboblari panelidagi ma'lumotlar brauzerining tugmachalarini bosish orqali tezda ochishingiz mumkin.

Holat ko'rsatkichlari. Ranglar va shakllar ma'lumotlar va natijalarning joriy holatini tavsiflaydi:

Belgi	Holat
	Shakl to'liq kiritilmagan
	Shakl to'liq kiritilgan
	Formaga hech narsa kiritilmagan. Kirish majburiy emas
	Formada kirish va chiqish ma'lumotlari mavjud
	Formada natijalar mavjud
	Formada natijalar mavjud, ammo hisob-kitoblarda ogohlantirishlar mavjud edi
	Formada natijalar mavjud, ammo hisob-kitoblarda ogohlantirishlar mavjud edi
	Formada natijalar mavjud, ammo men kiritishni hisoblab chiqqandan so'ng, men ma'lumotni o'zgartirdim
	

Hisoblashni sozlash.

Ko'pgina sozlamalar O'rnatish> Xususiyatlar> Global shaklda kiritilgan (Setup>Specifications>Global).

Hisobotlarda ishlatiladigan jadvalning nomi.

Hisoblash turi.

Kirish va chiqish miqdorlarining birliklari.

Mumkin bo'lgan fazalar (ya'ni bug'-suyuq yoki bug 'suyuq-suyuqlik).

Bosim va atrof-muhit harorati.

Oqim bo'yicha hisoblash natijalarini chiqarish parametrlari Setup> Hisobot parametrlari> Oqim (Setup> Report Options>Stream) shaklida.

Hisoblash sozlamalari vazifa shakli.

Hisoblash turini tanlash (O'rnatish (Setup) > Xususiyatlar(Specifications)> (Run type) Run turi maydonidagi global shakl.

Bajarish turi	Ta'ri
Flowsheet	Jarayonlar oqimining standart sxemasi mulkni baholash, test ma'lumotlarini tahlil qilish va / yoki mulk tahlilini hisoblashni o'z ichiga olishi mumkin
Assay Data	Quyida ko'rsatilgandek, ushbu parametrlarni jarayon oqimining diagrammasisiz alohida saqlash mumkin.
Analysis	Aspen Plus jarayonlari sinov ma'lumotlarini tahlil qilish va texnologik sxemalarda keyinchalik foydalanish uchun soxta tarkibiy qismlarni yaratish uchun Assay Data Analysis-dan foydalaning.
Data Regression	Jismoniy mulk modelini tanlash uchun "Ma'lumot Regression" turidan foydalaning. Aspen Plus aralashmaning sof tarkibiy qismlarini, bug '-suyuqlik balansini, suyuqlikni ajratib olish va boshqa parametrlarni aniqlash uchun zarur. Ma'lumot turida regressda mulkning reytingi va mulkni tahlilini hisoblash ham ko'rsatilishi mumkin.
Properties Plus	Aspen Custom Modeler yoki Aspen Pinch bilan foydalanish uchun mulk paketini, uchinchi tomonning tijorat muhandislik dasturlari yoki kompaniyangizning shaxsiy ishlanmalarini tayyorlash va Properties Plus turidan foydalanish kerak.
Property Analysis	Xususiyatlar jadvallari, diapazonlar yaratish uchun, harorat va bosim, suyuqlikning qoldiq fraksiyalari egri chizig'i va boshqalarida "Mulkni tahlil" qilishdan foydalaning. " Mulkni tahlil" qilish test ma'lumotlarini analitik hisoblash, mulkni baholashni o'z ichiga olishi mumkin.
Property Estimation	Mulk parametrlarini baholash uchun mulkni baholashni xohlamasangiz, xuddi shu qismda jarayonni simulyatsiya qilishdan foydalaning.
Assay Data Analysis	Sinov ma'lumotlarini tahlil qilish, keyinchalik texnologik diagrammalarda foydalanish, Aspen Plus jarayonlari uchun soxta tarkibiy qismlar yaratish uchun Assay Data Analysis-dan foydalaning

Birlik sozlamalari. Aspen Plus-dagi bo'linmalarni uch xil darajada aniqlash mumkin:

1. Global daraja (Eslatmaning "Kirish ma'lumotlari" ("Input Data") va "Chiqish natijalari" ("Output Results") satrlari Sozlash> Xususiyatlar> Global).

2. Ob'ektlar darajasi (ob'ektning har bir shakli tepasida (blok yoki oqim) ustidagi "birliklar" (birlik)).

3. Maxsus maydonlar darajasi

4. Setup> Units-Sets sahifasini ochib, foydalanuvchilar o'zlarining shaxsiy to'plamlarini yaratishlari mumkin. O'lchov birliklari mavjud to'plamlar asosida yaratilishi mumkin va keyin ularni o'zgartiradi.

Komponentlar (oddiy moddalar). Modelda zarur bo'lgan barcha tarkibiy qismlarni aniqlash uchun Komponent> Xususiyatlar (Components>Specifications) shaklidan foydalaning. Agar imkon bo'lsa, har bir komponentning fizik xususiyatlari ma'lumotlar bazasidan olinadi. **Simple-** Oddiy moddalarning ma'lumotlar bazalarida molekulyar og'irlik, muhim xususiyatlar va boshqalar kabi parametrlar mavjud. Ma'lumotlar banklarida qidirish tartibi "Databanks" yorlig'ida aniqlanadi.

Find -tugmachasini ro'yxatga tarkibiy qismlarni qidirish va qo'shish uchun ishlatishingiz mumkin.

Electrolyte Wizard - elektrolitlar tarkibiy qismlarini qo'shish uchun sehgardan foydalanishingiz mumkin.

Komponent parametrlari shakli. Komponentlarni tanlash.

Komponent identifikatori kirish ma'lumotlarini kiritishda va simulyatsiya natijalarini ko'rsatishda komponent nomini ko'rsatish uchun ishlatiladi.

Komponent nomlarini to'g'ridan-to'g'ri kiritish quyidagi usullardan biri bilan amalga oshiriladi:

Formula: Komponentning kimyoviy formulasiga muvofiq (masalan, C_6H_6).
Izomerlar uchun quyidagicha yozing: C_2H_6O-2

Component Name: Ingliz tilidagi komponentning to'liq nomi bilan (masalan, BENZENE - benzol).

Find "Topish" tugmasi ma'lumotlar banklarida tarkibiy qismlarni qidirish oynasini ochadi.

Ma'lumotlar bazalarida tarkibiy qismlarni qidirish. Komponent nomi, formulasi, komponentlar sinfi, molekulyar og'irligi, qaynash nuqtasi yoki CAS raqami bo'yicha qidiruv.

Belgilangan shartlarga javob beradigan barcha tarkibiy qismlar topiladi:

Agar bir nechta mezonlar ko'rsatilgan bo'lsa, qidirish "AND" mantiqiy operatsiyasi yordamida amalga oshiriladi.

26-ma'ruza. Mathcad , Matlab va aspen amaliy dasturiy paketlarini kimyo injinerligida qo'llanilishi.

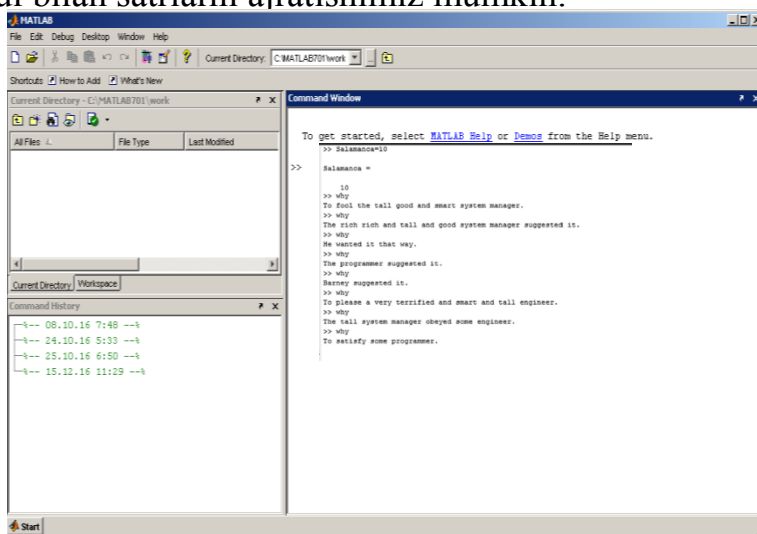
Reja:

1. Matlab dasturining kimyo injinerligida qo'llanilishi.
2. Matlabda ma'lumotlarni saqlash.
3. Matlabda grafik.

MATLAB maketini, 26.1-rasmda ko'rish mumkin. Foydalanuvchi belgilashi uchun bu erda bir qator turli maketlar bor. Biz ish joyiga ega bitta asosiy maketni tanlaymiz, u rasmda chap yuqori burchagida, o'zgaruvchilar saqlanadigan joyda bo'ladi; buyruq tarixi; o'ng pastki burchakda bo'ladi, bu yerda faoliyatlarni ishlatish ham saqlanadi va buyruqlar oynasi ham shu erda bo'lib u hisob-kitoblarni amalga oshirish imkonini beradi.

MATLAB ichida biz barcha algebraik, chizmachilik qobiliyatlari bilan birga oldindan belgilangan statistik vazifalarga egamiz deb aytish uchun hojat yo'qdir. Asosiy vazifalarni qisqacha sharhlashdan oldin, buyruq oynasining yordamidan foydalanishni ishora qilib o'tish qiziqarlidir. Biz "Yordam" funksiyasidan foydalanganda, qurilma funksiyalar va MATLAB kichik namoyish bo'lib shu funksiyaga qaytadi, dalillarni va imkoniyatlari dan foydalaniladi. (masalan, "hel ode45" ishlatib ko'ring Har qanday yangi hisobni boshlashdan oldin, biz ish maydonini tozalash uchun "barchasini o'chiq" buyrug'idan foydalanishimiz mumkin.

MATLAB operatsiyalari matrislarga tayanadi; shunday qilib, birinchi mavzu bu qanday qilib vektorlar va Matrislarni kiritishdan iborat. A satr vektori " $a = [a_{11} \ a_{12} \ a_{13}]$ " dan foydalanilgan holda yozilgan va unda elementlar bo'shliqlar yoki vergul bilan ajratilgandir. Ustun vektorini kiritish o'rniga, biz transpoz funksiyasini satr vektoriga qo'llashimiz yoki to'g'ridan-to'g'ri $a[a_{11} \ a_{12} \ a_{13}]$ dan foydalanib nuqta-vergul bilan satrlarni ajratishimiz mumkin.



26.1-rasm. Matlab dasturi umumiy ko'rinishi.

Workspace			
Name ▲	Value	Min	Max
a	[1 3 5 7]	1	7

26.2-rasm. Ish Maydoni.

Vektorni bir qancha raqamlardan ham yasash mumkin, bu ayniqsa mustaqil o'zgaruvchilar uchun qadriyatlar ro'yxatini taqdim etishda xususan foydalidir. Ushbu vazifasini ishlatish uchun, biz boshlang'ich elementi belgilaymiz, qadam (u salbiy ham bo'lishi mumkin), va yakuniysi:

$a = 10:-2:2$

$a = [10\ 8\ 6\ 4\ 2]$

matrislar kiritish uchun sintaksis o'xshaydi:

$A = [[a]; [b]; [c]]$

yoki

$A = [a; b; c]$

a, b, c lar vektor

Vektorli va Matrislar (3D gacha matrislarni saqlash mumkin) ma'lumotlarni, eksperimental axborot, hisob natijalarini saqlash uchun ishlatilishi mumkin. Har qachon vektorga kirilganda, o'zgaruvchi ism, qiymatlar va tur keltirilgan ish joyida hosil bo'ladi; 26.2-rasmga qarang.

Biz ish maydonini fayl (.mat) sifatida saqlashimiz mumkin va hamma vaqt ishga tushirishimiz mumkin bo'ladi. Bu Hisoblash natijalarini saqlab qolish uchun, ayniqsa, qiziqarlidir. Biz uni to'g'ridan-to'g'ri ish maydoni ustidan disk ramziga bosgan holda yoki buyruq yordamida saqlashimiz mumkin:

"Save FILENAME"

Shu bilan bir qatorda, biz FILENAME keyin ismlarini qo'shib faqat ba'zi parametrlarni saqlashingiz mumkin.

Keyinchalik, biz "LOAD FILENAME" dan foydalanamiz.

Oldindan saqlangan ma'lumotlarni yuklash uchun yoki biz ish maydoni ustida ochiq belgisidan foydalanib, uni ochishimiz mumkin.

Ish maydoniga saqlangan o'zgaruvchilar ish maydonida ular ustiga ikki marta bosilib ochilishi yoki taxrir qilinishi mumkin va shunda buyruq oynasi ustidan yangi oyna ochiladi 26.3-rasm, muxarrir ochilib qiymatlar taqdim etiladi. Biz ma'lumotlarni to'g'ridan-to'g'ri bu erda kiritishimiz mumkin.

Bundan tashqari, siz ma'lumotlarni saqlash va yuklash uchun Excel®dan foydalanishingiz mumkin. Biz ma'lumotlarni Excel faylidan ish maydoniga olib o'tish uchun "xlsread" dan foydalanishimiz mumkin. Umumiy syntax bu:

$Matrix_name = xlsread(file\ name, lavha, xlRange, «asos»)$

Variables - a				
ans x a x				
a <1x4 double>				
	1	2	3	4
1	1	3	5	7
2				
3				
4				

26.3-rasm. Array muharriri.

Barcha parametrlardan foydalanishga hojat yo'q:

- Fayl nomi bu manbai sifatida foydalaniladigan faylning nomi.
- Varraq bu ma'lumotlar joylashgan varaqning soni.
- xlRange bu siz joylashtirmoqchi bo'lgan kerakli kataklar oralig'i.
- asos formatini ifodalaydi.

Siz MATLAB dan Excel ga qarama qarshi yo'nalishda borishingiz mumkin, va bunda siz "xlswrite" buyrug'idan foydalanasiz matritsani eksport qilish uchun. Umumiy formulatsiya bu:

xlswrite (filename, A, varaq, xlRange) bunda:

- Fayl nomi bu siz sana yubormoqchi bo'lgan faylning nomi.
- A matrisadir.
- Varraq bu siz yozadigan Excel varag'i.
- xlRange bu siz eksport qiladigan kataklar oralig'i.

Tajribali foydalanuvchilar uchun, biz elektron jadval linkining xususiyatini qoldiramiz.

Vektorl yoki matriksalar shaklida saqlanadigan ma'lumot element ma element nomlanishi mumkin, masalan, birinchi qator elementini va birinchi ustun elementini olish uchun, bizda quyidagilar bor

$$b = a (1, 1)$$

yoki biz kabi, o'z navbatida, butun qatorni yoki bir ustunni shunday atashimiz mumkin:

$$c (i, :) \text{ yoki } c (:, i)$$

matrislar bilan shug'ullanganda ayrim qiziqarli funksiyalar borligini ko'ramiz:

Hajmi: vektorlarni matritsa yoki vektor ustunlari va satrlari soni bilan qaytadi

Length: vektor elementlari sonini qaytaradi

Ko'p hollarda, bunday hisoblash uchun ishlaydigan ma'lumotni saqlash qiziqarlidir.

Bundan tashqari, umumiy matrix operatsiyalari ham ajralmas funktsiyadir huddi quyidagilar kabi

INV (A): matrix A teskarisini qaytaradi

Det (A): matrix A aniqlovchisini qaytadi

Rank (A): matrix A unvonini qaytadi

Ko'z (n): hajm Matrixini aniqlash [n, n].

Birlar (m, n): hajmi Matrixi [m, n] birlarning to'lig'i.

Nol'lar (m, n): hajmi Matrixi [m, n] nollarning to'lig'i.

[L, U] = LU (A): matritsa A ning dekompozitsiyasi. Odatda chiziqli tenglamalar tizimini echish uchun MATLAB ichida foydalaniladi. Siz [L, U] ko'rsatishingiz

kerak, shunda ikkala matrixlar ham saqlanadi.

[Q, R] = QR (A): A matrisalarning QR decompositsiyasi Q va R matrixlarida saqlanadi.

ikki matritsalarini qo'shish yoki ayirish uchun, biz "+" va "-" operatorlaridan foydalanishimiz mumkin. Biz "^" belgisini matritsalar kuchini yoki (A)ni eksponensialini hisoblash uchun foydalanishimiz mumkin.

Boshqa tomondan, biz element ma element kuchini ko'paytirishimiz / bo'lishimiz yoki hisoblashimiz mumkin. Bu holda, biz vektor yoki Matrix nomidan keyin nuqtani ishlatamiz va keyin operatorni:

Joylashtirish va chizmachilik

Ko'pchilik muhandislik va ilmiy sohadagi masalalar eksperimental ma'lumotlarni moslashtirish va ularni chizish uchun grafik vositalaridan foydalanishni talab qiladi. Biz agar chizmachilikdan noshlasak, asosiy vazifasi chizmachilik bo'ladi. Biz turlichizmachilik kundalik ishlarini misollarda ko'rsatamiz. Biz birinchi bo'lib vektorlarni aniqlaymiz:[A-1,95-bet]

$t = [1: 50]$

$C1 = 25 * \exp(-0.05 * t)$

$c2 = 0,5 * t$

Maslahat: Biz hisoblash natijalarini buyruq oynasida ko'rishni istamasak biz ";" ni chiziqdan so'ng ishlatamiz. Nihoyat, chizmachilik funksiyasidan foydalanish vektorlarni talab etadi va biz variantlar tariqasida ramzlarni va chiziq turlari har bir ";" lar orasidagi seriyalarga qo'shishimiz mumkin:

chizmachilik(t, C1, 'k-', t, C2, 'k---')

Shu bilan bir qatorda, biz foydalanishingiz mumkin

*chizmachilik(t [25 * exp(-0.05 * t)], 'k-', t, 0,5 * t, 'k----')*

chizmachilik turlarining qisqacha hulolasi buyruq oynasida ko'rinishi mumkin, bunda "yordam chizmasi" qo'llaniladi.

Biz belgi, yozuv, va nomni to'g'ridan-to'g'ri buyruq oynasida kiritishimiz mumkin:

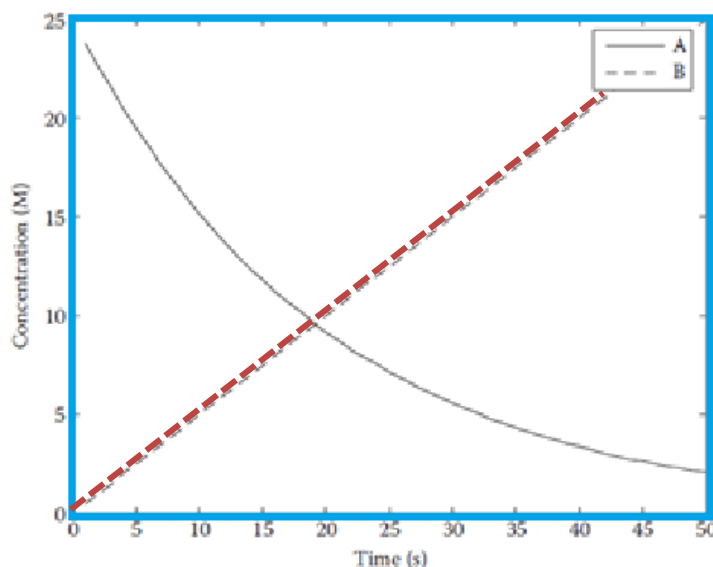
Xbelgi («Time (s)»)

ybelgi(konsentratsiya(M))

yozuvi («A», «B»)

Natija chizmasi 26.4-rasmda ko'rinishi mumkin.

A Maslahat: buyruq oynasida, biz o'qlarni tepada ishlatish bilan oldingi operatsiyalarini qaytarib olib va ma'lumotni qayta ishlatishimiz mumkin, bunda niz nomni qayta kiritishimizga hojat qolmaydi.



26.4-rasm. Oddiy chizmachilik misoli.

Biroq, shakl chizilgandan so'ng, biz unichiziq turi va shuningdek belgilar yoki yozuvlar joylashgan joyini o'zgartirish uchun uni tahrirlashimiz mumkin; 26.5-rasmga qarang.

Bir muhim xususiyati shundaki, shaklni .tiff yoki .eps. sifatida saqlashdir, bular dastur ishlatilishi uchun boshqa tasvirlar uchun umumiy formatlardir.

Ma'lumotlarni sig'irish uchun biz polyfitdan foydalanishimiz mumkin. Funktsiya vektorlar va polinominallarning darajasini input talab etadi.

Polyfit (t, c1,3)

funktsiya polinominallarning koeffitsiyentlarini qaytaradi:

polyfit (t, c1,3)

ans = -0,0002 0,0219 -1,1541 24,7519

Biz, shuningdek, defolt vektorlar o'rniga koeffitsiyentlarni to'g'ri vektorda saqlashimiz ham mumkin:

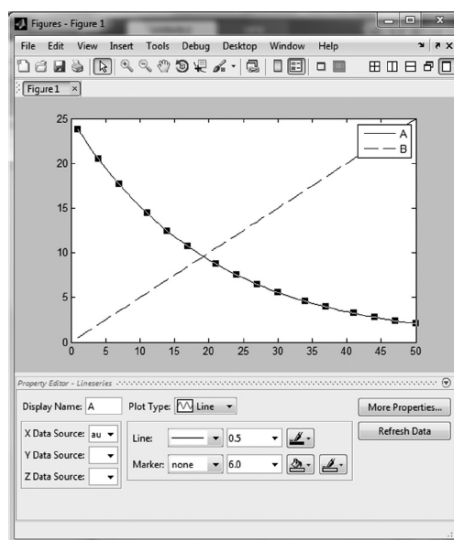
ans

coeff = ans

Joylashtirilgan polinomial sifatida qiymatlarni hisoblash uchun, biz polyvaldan foydalanishimiz mumkin:

C125 = polyval (coeff, 12,5) 13.4352

Biz, shuningdek, kamida-kvadratchalar modelidan lsqcurvefit kabilarni joylashtirishda foydalanishimiz mumkin.



16.5-rasm. Tahrir chizmalari.

Nazorat savollari.

1. Matlab dasturini ishga tushunish ketma-ketligini ayting.
2. Matlab dasturida ma'lumotlarni xotiraga saqlash uchun qaysi buyruqdan foydalanamiz?
3. Matlabda chizma chizishda qaysi buyruqlardan foydalaniladi?

27-ma'ruza. Mathcad , Matlab va aspen amaliy dasturiy paketlarini kimyo injinerligida qo'llanilishi.

Reja:

1. Aspen amaliy dasturining kimyo injinerligida qo'llanilishi.
2. Matlab dasturining kimyo injinerligida qo'llanilishi.
3. MatCad dasturining kimyo injinerligida qo'llanilishi.

Aspen Plus-da barcha jihozlarni simulyatsiya qilishning hojati yo'q. Jarayon kiritish ma'lumotlarining aniqligi va izchilligini ta'minlash. Natijalar nomuvofiq va haqiqiy emasligiga ishonch hosil qiling.

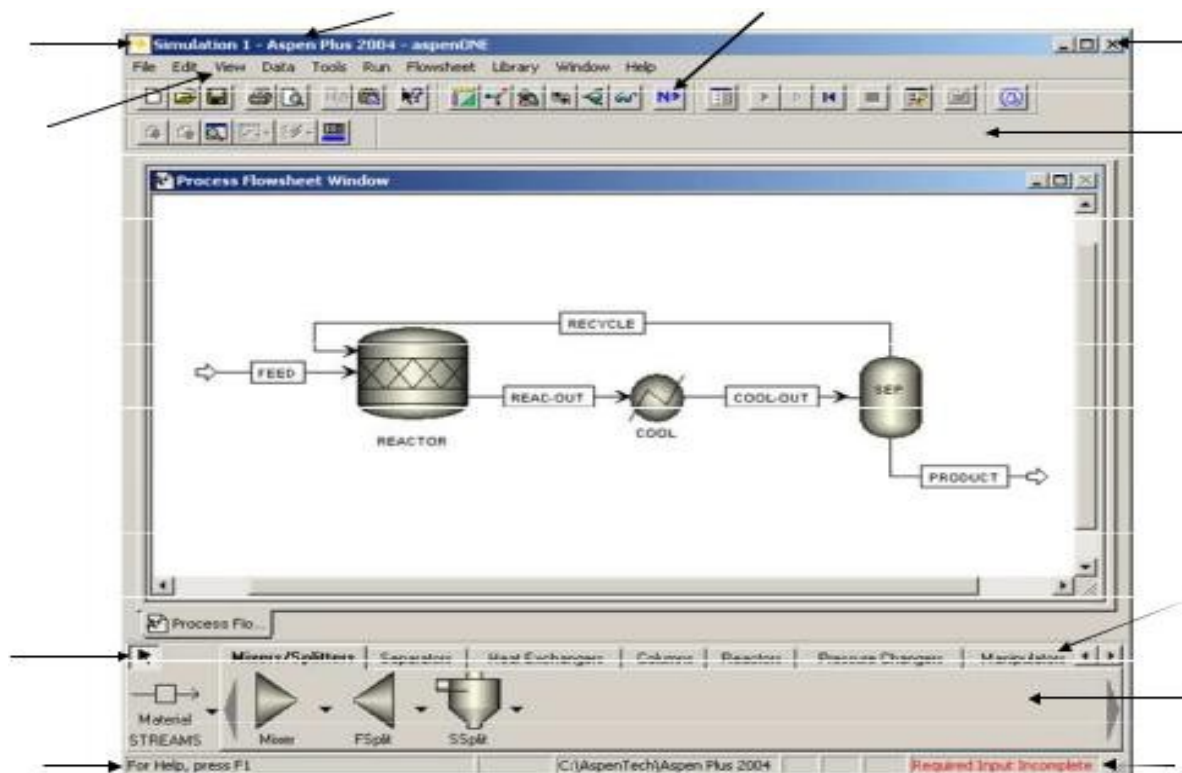
Aspen Plus-ning ba'zi muhim xususiyatlari.

- Elektrolitlarni aniq modellashtirish.
- Qattiq moddalar bilan ishlash.
- Ma'lumotlar regressiyasi.
- Ma'lumotni moslashtirish.
- Xususiyatlarni baholash.
- Foydalanuvchi hisoblari.

Onlayn optimallashtirish tizimlarida foydalanish uchun modellashtirish, optimallashtirish va ma'lumotlarni muvofiqlashtirishni o'z ichiga olgan tenglamaga yo'naltirilgan modellashtirish.

Aspen Plus foydalanuvchi interfeysi.

Foydalanuvchi interfeysi



27.1-rasm Aspen Plyus dasturi umumiy ko'rinishi.

Ekranda quyidagilar aks etgan:

- 1.Idintifikator ishlash.
- 2.Sarlavha satri.
- 3.Menyu qatori.
- 4.“Keyingi” tugma.
- 5.Tugmalar o'lchamlarini o'zgartirish.
- 6.Panellar oynasi.
- 7.Blok sxemasi jarayoni.
- 8.Asboblar paneli oynasi.
- 9.Modellar kutubxonasi xatcho'plari
- 10.Kutubxona modellari.
- 11.Qator.
- 12.Ko'rsatmalar.

Oqim va bloklarning avtomatik nomi

Ob'ektni yaratishda foydalanuvchi Aspen Plus yordamida ob'ekt nomini qo'lda yoki avtomatik ravishda kiritishi mumkin.

Nomlash parametrlarini o'zgartirish uchun:

1. Asboblar menyusidagi Variantlarni tanlang;
2. Flowsheet jadvaliga o'ting;
3. Kerakli nomlash variantlarini belgilang va keraksizlarini olib tashlang.

Bloklash va oqim nomlari ko'rsatilishi yoki yashirin bo'lishi mumkin, ob'ektni tanlang, o'ng tugmani bosing va menyudan yashirish variantini tanlang.

Bloklar - Jarayon oqimining grafik operatsiyalari .

Jarayon sxemasiga blokni joylashtirish uchun:

1. Namunaviy model kutubxonasida kerakli xatcho'pni tanlang.
2. Texnologik ob'ektning modelini tanlang. Ochiladigan o'qni bosing va model uchun belgini tanlang.
3. Blokni joylashtirish uchun avval blok belgisini, so'ngra jarayon sxemasini bosing. Shuningdek, belgini bosishingiz va uni diagrammaga sudrab borishingiz mumkin.
4. Diagrammadagi bo'sh joyni yana bir marta bosish bilan siz shunga o'xshash boshqa blokni va boshqalarni qo'shishingiz mumkin.
5. Bo'sh maydonga bloklarni joylashtirish rejimidan chiqish uchun sichqonchaning o'ng tugmachasini bosing.

Oqimlar - Jarayon oqimining grafik operatsiyalari -

Oqimni jarayon jadvaliga o'rnatish uchun:

1. Modellar kutubxonasidagi "Oqimlar" (STREAMS) belgisini bosing.
2. Belgining yonidagi "pastga" tugmani bosib, oqim turini (material, energiya yoki ish) tanlang.
3. Ulanish uchun jihozning ajratilgan portini (kirish yoki chiqish) bosing.
4. Oqimning qolgan qismlarini ulash uchun 3-bosqichni takrorlang.
5. Oqimning uchidan birini texnologik jarayonning kirishiga yoki chiqishiga joylashtirish uchun kontaktlarning zanglashiga olib keladigan qismni bosing.
6. Oqimni yaratish rejimidan chiqish uchun sichqonchaning o'ng tugmachasini bosing.

Sichqoncha tugmalaridan foydalanish

chap tugma: - blok, oqim, ob'ekt yorlig'i yoki izohni tanlaydi;

o'ng tugma: - Tanlangan oqim, blok yoki diagramma uchun menyuga qo'ng'iroq qiladi;

Ikki marta tez-tez bosing -oqim yoki blok uchun ma'lumot kiritish formasini yoki oraliq oqim uchun natija shaklini ochadi;

yorliqli matn yoki izohlarni tahrir qiladi

Bloklar va oqimlarni o'zgartirish.

Ma'lumotlar brauzerida kiritish shakllari va natijalarini ko'rsatish uchun:

1.a) Sizni qiziqtirgan ob'ektni ikki marta bosing yoki (agar hisoblamagan bo'lsangiz), Blok yoki oqimni tanlang, o'ng tugmani bosing va menyu punktini tanlang Input...

2. Blok yoki oqim ko'rinishini o'zgartirish uchun:

3. Ob'ektni tanlang va sichqoncha chap tugmasini bosing yoki Ctrl+K tugmalarini bosing.

4. Belgilangan ob'ektning belgisini ustiga olib boring va sichqonchaning o'ng tugmachasini bosib, ushbu ob'ektning menyusini chaqiring;

5. "Exchange Icon" menyusini tanlang.

O'z-o'zini modellashtirish vazifasi.

Benzolni olish uchun texnologik jarayonning grafik diagrammasini tuzish kerak. Bloklar uchun mos belgilarni tanlang. Oqimni tanlash uchun o'ng tugmachasini bosing, Qayta ulash joyi / Resurs manbasini tanlang va oqim tugashi / boshlanishini tanlangan blok portiga ulang, simulyatsiya tugagach, vazifani

BENZENE.BKP sifatida saqlang.

3.Asosiy ma'lumotlarni kiritish

Aspen Plus interveysi dastur parametrlari va buyruqlarni belgilash uchun grafik interfeys menyusidan foydalanadi, boshqaruv panellari - eng tez-tez ishlatiladigan ba'zi funktsiyalarga kirish, modellar kutubxonasi - oqim jadvalini tuzish uchun ishlatiladi.

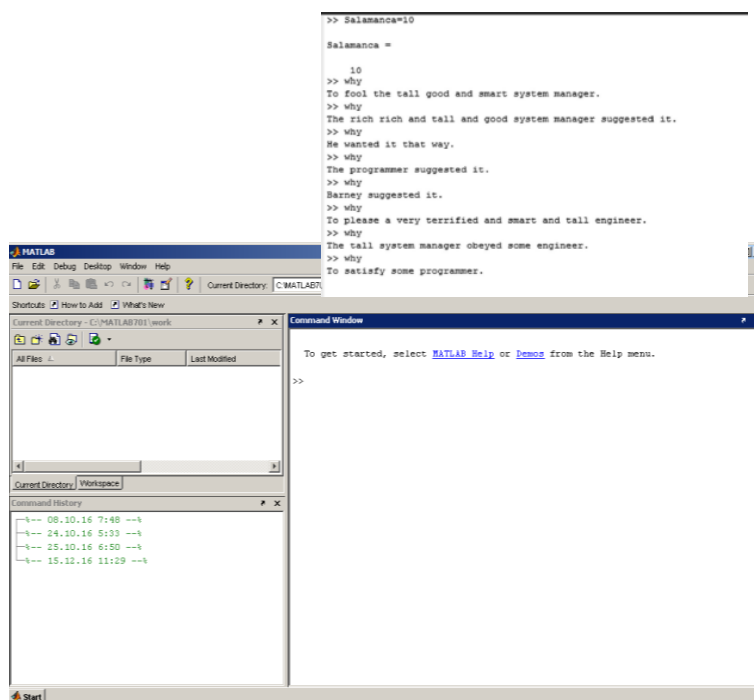
“Ma'lumotlar Brauzeri”(Data Browser) dan ma'lumotlar kiritish shakllari bilan ishlash uchun foydalaniladi, «Далее» tugmasi yordamida uni ko'chirish, hajmini o'zgartirish, ixchamlashtirish, ko'paytirish va yopish texnologik sxema ob'ektlarini boshqaruvchisiga ega, joriy forma to'ldirilganligini va ma'lumotlarni kiritish talab qilinadigan keyingi shaklga o'tishini tekshiradi

Ma'lumotlar brauzeri (Data Browser) - bu o'zgaruvchilarga kirish, hisoblash natijalari va ob'ektlarining ierarxik ravishda tuzilgan vakili. Undan hisoblash sozlamalari, oqimlari, bloklari, yordamchi dasturlari va boshqa shakllarini yo'naltirish uchun foydalanish mumkin.

Texnologik sxema bo'yicha ma'lumotlarni kiritish yoki natijalarni va (yoki) boshqa papkalarni, ob'ektlarni kiritish uchun shakllardan iborat.

MATLAB ichida biz barcha algebraik, chizmachilik qobiliyatlari bilan birga oldindan belgilangan statistik vazifalarga egamiz deb aytish uchun hojat yo'qdir. Asosiy vazifalarni qisqacha sharhlashdan oldin, buyruq oynasining yordamidan foydalanishni ishora qilib o'tish qiziqarlidir. Biz “Yordam” funksiyasidan foydalanganda, qurilma funksiyalar va MATLAB kichik namoyish bo'lib shu funksiyaga qaytadi, dalillarni va imkoniyatlari dan foydalaniladi. (masalan, “hel ode45” ishlatib ko'ring Har qanday yangi hisobni boshlashdan oldin, biz ish maydonini tozalash uchun "barchasini o'chiq" buyrug'idan foydalanishimiz mumkin.

MATLAB operatsiyalari matrislarga tayanadi; shunday qilib, birinchi mavzu bu qanday qilib vektorlar va Matrislarni kiritishdan iborat. A satr vektori " $a = [a_{11} \ a_{12} \ a_{13}]$ " dan foydalanilgan holda yozilgan va unda elementlar bo'shliqlar yoki vergul bilan ajratilgandir. Ustun vektorini kiritish o'rniga, biz transpoz funksiyasini satr vektoriga qo'llashimiz yoki to'g'ridan-to'g'ri $a[a_{11} \ a_{12} \ a_{13}]$ dan foydalanib nuqta-vergul bilan satrlarni ajratishimiz mumkin.



27.2-rasm. Matlab dasturi umumiy ko'rinishi.

Workspace			
Name ▲	Value	Min	Max
a	[1 3 5 7]	1	7

27.3-rasm. Ish Maydoni.

Vektorni bir qancha raqamlardan ham yasash mumkin, bu ayniqsa mustaqil o'zgaruvchilar uchun qadriyatlar ro'yxatini taqdim etishda xususan foydalidir Ushbu vazifasini ishlatish uchun, biz boshlang'ich elementi belgilaymiz, qadam (u salbiy ham bo'lishi mumkin), va yakuniysi:

$a = 10: -2: 2$

$a = [10 \ 8 \ 6 \ 4 \ 2]$

matrislar kiritish uchun sintaksis o'xshaydi:

$A = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$

yoki

$A = [a; b; c]$

a, b, c lar vektor

Vektorli va Matrislar (3D gacha matrislarni saqlash mumkin) ma'lumotlarni, eksperimental axborot, hisob natijalarini saqlash uchun ishlatilishi mumkin. Har qachon vektorga kirilganda, o'zgaruvchi ism, qiymatlar va tur keltirilgan ish joyida hosil bo'ladi; 27.3-rasmga qarang.

Biz ish maydonini fayl (.mat) sifatida saqlashimiz mumkin va hamma vaqt ishga tushirishimiz mumkin bo'ladi. Bu Hisoblash natijalarini saqlab qolish uchun, ayniqsa, qiziqarlidir. Biz uni to'g'ridan-to'g'ri ish maydoni ustidan disk ramziga bosgan holda yoki buyruq yordamida saqlashimiz mumkin:

"Save FILENAME"

Shu bilan bir qatorda, biz FILENAME keyin ismlarini qo'shib faqat ba'zi parametrlarni saqlashingiz mumkin.

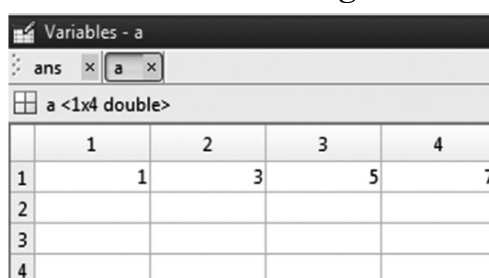
Keyinchalik, biz "LOAD FILENAME" dan foydalanamiz.

Oldindan saqlangan ma'lumotlarni yuklash uchun yoki biz ish maydoni ustida ochiq belgisidan foydalanib, uni ochishimiz mumkin.

Ish maydoniga saqlangan o'zgaruvchilar ish maydonida ular ustiga ikki marta bosilib ochilishi yoki taxrir qilinishi mumkin va shunda buyruq oynasi ustidan yangi oyna ochiladi 27.4-rasm, muxarrir ochilib qiymatlar taqdim etiladi. Biz ma'lumotlarni to'g'ridan-to'g'ri bu erda kiritishimiz mumkin.

Bundan tashqari, siz ma'lumotlarni saqlash va yuklash uchun Excel®dan foydalanishingiz mumkin. Biz ma'lumotlarni Excel faylidan ish maydoniga olib o'tish uchun "xlsread" dan foydalanishimiz mumkin. Umumiy syntax bu:

Matrix_name = xlsread (file nome, lavha, xlRange, «asos»)



	1	2	3	4
1	1	3	5	7
2				
3				
4				

27.4-rasm. Array muharriri.

Barcha parametrlardan foydalanishga hojat yo'q:

- Fayl nomi bu manbai sifatida foydalaniladigan faylning nomi.
- Varraq bu ma'lumotlar joylashgan varaqning soni.
- xlRange bu siz joylashtirmoqchi bo'lgan kerakli kataklar oralig'i.
- asos formatini ifodalaydi.

Siz MATLAB dan Excel ga qarama qarshi yo'nalishda borishingiz mumkin, va bunda siz "xlswrite" buyrug'idan foydalanasiz matritsani eksport qilish uchun. Umumiy formulatsiya bu:

xlswrite (filename, A, varaq, xlRange) bunda:

- Fayl nomi bu siz sana yubormoqchi bo'lgan faylning nomi.
- A matrisadir.
- Varraq bu siz yozadigan Excel varag'i.
- xlRange bu siz eksport qiladigan kataklar oralig'i.

Tajribali foydalanuvchilar uchun, biz elektron jadval linkining xususiyatini qoldiramiz.

Vektorl yoki matriksalar shaklida saqlanadigan ma'lumot element ma element nomlanishi mumkin, masalan, birinchi qator elementini va birinchi ustun elementini olish uchun, bizda quyudagilar bor

$b = a (1, 1)$

yoki biz kabi, o'z navbatida, butun qatorni yoki bir ustunni shunday atashimiz mumkin:

$c (i, :) \text{ yoki } c (:, i)$

matrislar bilan shug'ullanganda ayrim qiziqarli funksiyalar borligini ko'ramiz:

Hajmi: vektorlarni matritsa yoki vektor ustunlari va satrlari soni bilan qaytadi

Length: vektor elementlari sonini qaytaradi

Ko'p hollarda, bunday hisoblash uchun ishlaydigan ma'lumotni saqlash qiziqarlidir.

Bundan tashqari, umumiy matrix operatsiyalari ham ajralmas funktsiyadir huddi quyidagilar kabi

INV (A): matrix A teskarisini qaytaradi

Det (A): matrix A aniqlovchisini qaytadi

Rank (A): matrix A unvonini qaytadi

Ko'z (n): hajm Matrixini aniqlash [n, n].

Birlar (m, n): hajmi Matrixi [m, n] birlarning to'lig'i.

Nol'lar (m, n): hajmi Matrixi [m, n] nollarning to'lig'i.

[L, U] = LU (A): matritsa A ning dekompozitsiyasi. Odatda chiziqli tenglamalar tizimini echish uchun MATLAB ichida foydalaniladi. Siz [L, U] ko'rsatishingiz kerak, shunda ikkala matrixlar ham saqlanadi.

[Q, R] = QR (A): A matritsalarining QR dekompozitsiyasi Q va R matrixlarida saqlanadi.

ikki matritsalarini qo'shish yoki ayirish uchun, biz "+" va "-" operatorlaridan foydalanishimiz mumkin. Biz "^" belgisini matritsalar kuchini yoki (A)ni eksponensialini hisoblash uchun foydalanishimiz mumkin.

Boshqa tomondan, biz element ma element kuchini ko'paytirishimiz / bo'lishimiz yoki hisoblashimiz mumkin. Bu holda, biz vektor yoki Matrix nomidan keyin nuqtani ishlatamiz va keyin operatorni:

Joylashtirish va chizmachilik

Ko'pchilik muhandislik va ilmiy sohadagi masalalar eksperimental ma'lumotlarni moslashtirish va ularni chizish uchun grafik vositalaridan foydalanishni talab qiladi. Biz agar chizmachilikdan noshlasak, asosiy vazifasi chizmachilik bo'ladi. Biz turlichizmachilik kundalik ishlarini misollarda ko'rsatamiz. Biz birinchi bo'lib vektorlarni aniqlaymiz:

$t = [1: 50 \ 1]$

$C1 = 25 * \exp(-0.05 * t)$

$c2 = 0,5 * t$

Maslahat: Biz hisoblash natijalarini buyruq oynasida ko'rishni istamasak biz ";" ni chiziqdan so'ng ishlatamiz. Nihoyat, chizmachilik funksiyasidan foydalanish vektorlarni talab etadi va biz variantlar tariqasida ramzlarni va chiziq turlari har bir ";" lar orasidagi seriyalarga qo'shishimiz mumkin:

chizmachilik (t, C1, 'k-', t, C2, 'k---')

Shu bilan bir qatorda, biz foydalanishingiz mumkin

*chizmachilik (t [25 * exp(-0.05 * t)], 'k-', t, 0,5 * t, 'k----')*

chizmachilik turlarining qisqacha hulolasi buyruq oynasida ko'rinishi mumkin, bunda "yordam chizmasi" qo'llaniladi.

Biz belgi, yozuv, va nomni to'g'ridan-to'g'ri buyruq oynasida kiritishimiz mumkin:

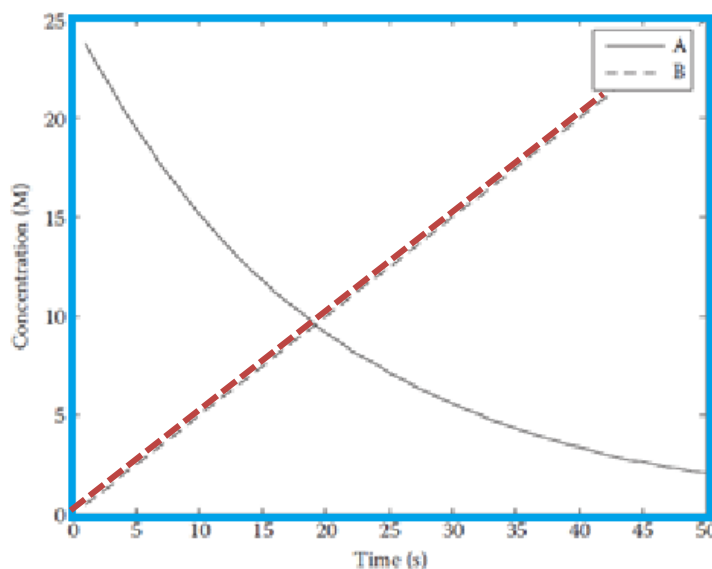
Xbelgi («Time (s)»)

ybelgi(konsentratsiya(M))

yozuvi («A», «B»)

Natija chizmasi 18.5-rasmda ko'rinishi mumkin.

A Maslahat: buyruq oynasida, biz o'qlarni tepada ishlatish bilan oldingi operatsiyalarini qaytarib olib va ma'lumotni qayta ishlatishimiz mumkin, bunda niz nomni qayta kiritishimizga hojat qolmaydi.



27.5-rasm. Oddiy chizmachilik misoli.

Biroq, shakl chizilgandan so'ng, biz unichiziqli turi va shuningdek belgilar yoki yozuvlar joylashgan joyini o'zgartirish uchun uni tahrirlashimiz mumkin; 18.6-rasmga qarang.

Bir muhim xususiyati shundaki, shaklni .tiff yoki .eps. sifatida saqlashdir, bular dastur ishlatilishi uchun boshqa tasvirlar uchun umumiy formatlardir.

Ma'lumotlarni sig'irish uchun biz polyfitdan foydalanishimiz mumkin. Funktsiya vektorlar va polinominallarning darajasini input talab etadi

Polyfit (t, c1,3)

funktsiya polinominallarning koeffitsiyentlarini qaytaradi:

polyfit (t, c1,3)

ans = -0,0002 0,0219 -1,1541 24,7519

Biz, shuningdek, defolt vektorlar o'rniga koeffitsiyentlarni to'g'ri vektorda saqlashimiz ham mumkin:

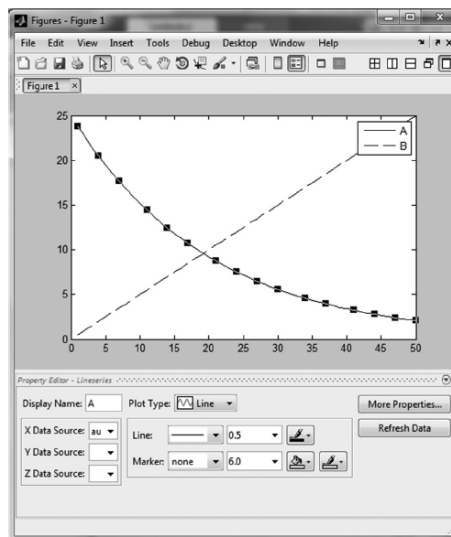
ans

coeff = ans

Joylashtirilgan polinomial sifatida qiymatlarni hisoblash uchun, biz polyvaldan foydalanishimiz mumkin:

C125 = polyval (coeff, 12,5) 13.4352

Biz, shuningdek, kamida-kvadratchalar modelidan lsqcurvefit kabilarni joylashtirishda foydalanishimiz mumkin.



27.6-rasm. Tahrir chizmalari.

Nazorat savollari.

1. Matlab dasturini ishga tushunish ketma-ketligini ayting.
2. Matlab dasturida ma'lumotlarni xotiraga saqlash uchun qaysi buyruqdan foydalanamiz?
3. Matlabda chizma chizishda qaysi buyruqlardan foydalaniladi?

**O‘ZBEKISTON RESPUBLIKASI
OLIIY VA O‘RTA MAXSUS TA‘LIM VAZIRLIGI**

**ISLOM KARIMOV NOMIDAGI TOSHKENT
DAVLAT
TEXNIKA UNIVERSITETI TERMIZ FILIALI**

**KIMYO INJINERINGIDA
AXBOROT TEXNOLOGIYALARI
o‘quv fanidan**

AMALIY MASHG‘ULOTLAR

1-amaliy mashg'ulot.Kimyo injiniringda axborot texnologiyalari faniga kirish.

Ishdan maqsad.Axborot texnologiyalari fani, uning ishlatilish sohalari, axborotning turlari, xossalari bilan tanishish.

Vazifa.

- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Nazariy ma'lumot yuzasidan berilgan testlarni bajarish.

1 . Berilgan nazariy materiallarni o'rganish.

Axborot texnologiyalari fanini o'qitishdan maqsad, zamonaviy axborot texnologiyalar haqida tushuncha berish, zamonaviy dasturiy vositalar, zamonaviy kompyuter dasturlarida ishlashni bilishni o'rgatish va ta'lim jarayoniga mos o'quv resrularini kompyuterdan foydalanib, yaratish. Ta'lim jarayoniga mos elektron darsliklarini, o'quv elektron resurslarni, on-layn tizimida va off-layn tizimida ishlaydigan vositalarni yaratishni o'rgatishdan iboratdir. Bu fan o'zining oldiga bir necha vazifalarni qo'yib olgan, ya'ni birinchi istiqbol rejalaridan birinchisi, yaratilgan va yaratilayotgan yangi kompyuter dasturlaridan xabardor bo'lib turish, mavjudlarini amaliyotda qo'llashni o'rgatish.

Axborot kommunikatsion texnologiyalar-bu audio, video va kommunikatsiya vositalarini axborotlarni hosil qilish, olish, ishlab chiqish va qayta ishlash hamda tarqatish jarayonidir. Bu jarayonda inson resurlari va texnik resurlar qatnashadi. Biri ikkinchilari uchun xizmat qiladi. Ya'ni texnik resurslar insonlarni axborotni olishlari uchun va boshqa odamlarga tarqatish uchun xizmat qiladi.

Texnologiya-maxorat, san'at degan ma'noga ega, logos esa-ta'lim, — Axborot texnologiyalari kursining maqsadi AKTdan ta'limda, jumladan, xorijiy til va adabiyoti fanini o'qitishda qo'llashni o'rgatish, talabalarning esa ta'limda qo'llaniladigan dasturlar, dasturiy vositalardan o'z mutahassisliklarida foydalanishni bilish ta'limda qo'llashni bilish, ularni o'rganish va foydalanish hamda qo'llashning zamonaviy tendensiyalardan biri, bu axborot texnologiyalarining turlari bilan tanishish, ularning qaysi sohaga oidligini farqlay bilish va ilg'or texnologiyalarni, bir tildan ikkinchi tilga o'girish texnologiyalari, matndan giper-matnga o'girish texnologiyalarini, taqdimot yaratish texnologiyalarini, ekspert tizimlar texnologiyalari, elektron xukumat, masofaviy ta'lim texnologiyalari, video-kurslar, elektron resurslar, on-layn taqdimotlar va boshqa ta'lim resurlarini yaratuvchi dasturlar va dasturiy vositalarni o'rganish maqsadga muvofiqdir.Ta'lim jarayoniga axborot kommunikatsion texnologiyalarini qo'llash ta'lim jarayonini mazmunli, chiroyli, unumli, qulay, vaqtni tejovchi va o'zlashtirish sifatini oshiruvchi etib amalga oshirib beradi. O'qituvchining dars berish jarayonini unumli, vaqtni tejovchi va mazmunli tarzda o'tkazib beradi. O'zlashtirish sifatida vaqtning qo'llanish evaziga, ya'ni kompyuter dasturlari qo'shimcha kanallari orqali ko'z xotirasini qabul qilish jarayonini ishga tushirib, ularga ta'sirni kuchytirib, rangli rasmlar, xarakatli animatsiyalarni o'qitish jarayoniga tadbiq etib, tasavvurni to'g'ri shakllantirish orqali bilish jarayonini, tafakkur etish qobiliyatini rivojlantiradi.

Ta'lim jarayoniga axborot kommunikatsiya texnologiyalarini qo'llashning

dolzarb vazifalaridan biri, bu elektron darsliklarni yaratuvchi, elektron resurlarni, taqdimotlarni, zamonaviy ta'lim vositalarini yaratuvchi dasturlarni o'rgatishdan iborat. Axborot katta jadallik bilan kirib kelmoqda, kim axborot bilan birinchi tanish bo'lsa, o'sha voqeani boshqaradi, jarayonni boshqarish uni qo'lida, ya'ni u dunyoni boshqaradi, deyishadi.

Katta xajmdagi axborotlarni etkazish, uni samarali egasiga berish, vaqtni tejash vazifasini bajarishda axborot texnologiyalar yordamga keladi. Konsepsiya-bu ta'limot deganidir.

1. AKT. O'zbekistonda AKT sohasini rivojlantirishdagi davlat siyosatining asosiy yo'nalishlari, amaldagi qonunlar, farmonlar va qarorlar.

2. Axborot va uning turlari, xossalari, uzatilishi, o'lchov birliklari.

3. Axborotni kodlash.

4. Ma'lumotlarning kompyuter xotirasida tasvirlanishi/ O'zbekistonda AKT sohasini rivojlantirish uchun 2005 yidagi bir qator qororlar va qonunlar isbot bo'la oladi. XXI asr axborot texnologiyalari asri bo'lgani uchun ta'limni ham bu texnologiyalarisiz tasavvur etish qiyin. Hukumatimiz tomonidan Axborotlashtirish to'g'risidagi, Elektron hukumat tashkil etish, uni rivojlantirish haqidagi bir qator qonunlar jamiyatni rivojlantirish, hukumatni boshqarishga yordam qilishga yo'naltirgan.

Texnologiyalar. —Texnologiya grekcha so'z bo'lib mohirlik, ustalik, biror ishni uddalay olishni anglatadi. Bu ma'lum bir jarayonga nisbatan qo'llanilgan. Jarayon deganda esa maqsadga erishishga yo'naltirilgan xatti-harakatlar majmui tushunilgan. Ushbu jarayon inson tomonidan tanlangan strategiya bilan belgilanadi va turli xildagi vositalar, usullar yordamida amalga oshiriladi. Umumiy hollarda texnologiya deganda, mahsulotni ishlab chiqarish jarayonida amalga oshiriladigan xomashyo, material yoki yarim tayyor mahsulot shakli, xususiyati, holatining o'zgarishi, uni qayta ishlash, tayyorlash usullarining majmui tushuniladi. Bu biror bir ishni yuqori darajada uddalash deganidir. Axborot texnologiyalari to'g'risida gap ketganda, material sifati ham, mahsulot sifatida ham axborot ishtirok etadi. Biroq bu ob'ekt, jarayon yoki hodisa to'g'risidagi sifat jihatidan yangi ma'lumot bo'ladi. Texnologiya xodimning axborot bilan ishlash usuli va uslubi hamda texnik vositalar orqali namoyon bo'ladi.

Axborot texnologiyasi – ob'ektning (axborot mahsulotining) holati, jarayon yoki voqeaning yangi xususiyati to'g'risida axborot olish uchun ma'lumotlarni yig'ish, qayta ishlash va uzatish vositalari va usullari majmuidan foydalaniladigan jarayondir.

Moddiy ishlab chiqarish texnologiyasining maqsadi – inson yoki tizimning ehtiyojini qondiruvchi mahsulot ishlab chiqarish sanaladi. Axborot texnologiyasining maqsadi esa – axborot ishlab chiqarish bo'lib, uni tahlil etish va uning asosida biror bir harakatga qo'l urish uchun tegishli qaror qabul qilish.

*Axborotlarni yig'ish, uzatish, to'plash, saqlash, taqdim etish va foydalanish uslublari va usullari tizimi **axborot texnologiyasi** deb yuritiladi. Axborot texnologiyasi avtomatlashgan va an'anaviy (qog'oz) ko'rinishija*

amalgga oshiriladi. Avtomatlashtirish hajmi va texnik vositalardan foydalanish turi aniq bir texnologiyaning mohiyatiga bog'liq.

Avtomatlashtirish – bu inson ish faoliyatini mashina va mexanizmlar bilan almashtirish demakdir. U texnik, iashkiliy va iqtisodiy mazmundagi hattiharakatlar hamda tadbirlar kompleksidan iborat bo'ladi, ishlab chiqarish jarayoni, boshqaruv jarayonining u yoki bu ishini amalga oshirishda inson ishtirokini qisman yoki butunlay cheklash imkonini beradi.

Moddiy va axborot texnologiyasining asosiy komponentlarini qiyoslash 1.1-jadvalda berilgan.

1.1-jadval.

Mahsulot ishlab chiqarish uchun texnologiya komponentlari	
<i>Moddiy mahsulot</i>	<i>Axborot mahsuloti</i>
Xom ashyo va materiallar tayyorlash	Ma'lumotlar yoki boshlang'ich axborotni yig'ish
Moddiy mahsulot ishlab chiqarish	Ma'lumotlarni qayta ishlash va yakuniy axborotga ega bo'lish
Iste'molchilarga ishlab chiqarilgan mahsulotni sotish	Uning asosida qaror qabul qilish uchun yakuniy axborotlar uzatish

Axborot texnologiyasining asosiy tavsifi.

Avtomatlashtirilgan axborot texnologiyasi (AAT) – boshqaruv vqazifalarini hal etish uchun tizimli tashkil etilgan axborot jarayonlarini amalga oshirish usul va vositalari majmuidir. U hisoblash texnikasi va aloqa vositalaridan foydalaniladigan rivojlangan dasturiy ta'minotni qo'llash bazasida bajariladi. Axborot texnologiyasining amalga oshirishdagi texnik vositalarning asosiy qisini komp yuter texnikasi tashkil etgan uchun axborot texnologiyasi, ayniqsa zamonaviy axborot texnologiyasi deganda komp yuter axborot texnologiyasi tushuniladi.

2.Zamonaviy axborot texnologiyasi

Zamonaviy axborot texnologiyasi (ZAT) (kompyuter axborot texnologiyasi) – shaxsiy komp yuter va telekommunikatsiya vositalaridan foydalanuvchi uchun qulay —interfeyslli axborot texnologiyasidir. Ma'lum bir turdagi komp yuter uchun mo'ljallangan bir yoki bir necha o'zaro bog'liq dasturiy mahsulotlar zamonaviy axborot texnologiyalarining vositasi sanaladi. Zamonaviy axborot texnologiyasining asosiy elementlari.

ZAT ning asosiy elementlari quyidagicha:

- ma'lum bir vaqt davomida axborotni kiritish va qayta o'zgartirish;
- tasvirni kiritish va unga ishlov berish;
- signal axboroti paydo bo'lgan yerda uni qayta ishlash;
- og'zaki axborotni qayta ishlash;
- foydalanuvchining ShK bilan faol muloqoti;
- turli axborot tizimlarida mashinali modellashtirish;

- axborot almashuvining tarmoq texnologiyasi (dialog yuritish, video va telekommunikatsiya, elektron pocha, vidioteka va x.z.);
- taqsimlangan tarmoq tizimlarida ma'lumotlarni mul'tiprotssessor asosida qayta ishlash;
- mahalliy, mintaqaviy va xalqaro tarmoqlar bo'yicha axborotni tezkor tarqatish.

Axborot texnologiyasining bazaviy texnologiyasi bo'lib quyidagilar hisoblanadi:

- a) texnik ta'minot texnologiyasi;
- b) telekommunikatsiya texnologiyasi;
- v) dasturiy ta'minot texnologiyasi.

Bu texnologiyalar hisoblash tizimlari va tarmoqlari arxitekturasining aniq variantlari doirasida birgalikda harakat qiladi va birlashadi. Ularni ayrimlari axborot texnologiyasi rivojlanishida hal qiluvchi rol o'ynaydi.

Axborot texnologiyalari taraqqiy etishining asosiy bosqichlari va rivojlanish tendensiyasi.

XIX asrning ikkinchi yarmigacha "qo'l" axborot texnologiyasi rivojlangan bo'lib, uning asosini pero, siyohdon va buxgalteriya daftari tashkil etgan. Kommunikatsiya (aloqa) paket (rasmiy xujjatlar solingan konvert) yuborish orqali amalga oshirilardi.

"Qo'l" axborot texnologiyasi o'rniga XIX asr oxirida "mexanik" texnologiya kirib keldi. Yozuv mashinasi, telefon, diktafonning kashf etilishi, jamoa pochta tizimining takomillashuvi – bular bari avvaliga axborotni qayta ishlash texnologiyasida, so'ng ish mahsuldorligida sezilarli o'zgarishlar yuz berishiga zamin bo'ldi.

XX asrning 40 - 60–yillarida —elektrli texnologiyasi paydo bo'lib, u yechib almashtiriladigan elementlarga ega elektr yozuv mashinalari, oddiy qog'ozdan foydalanuvchi nusxa ko'chirish mashinasi, portativ diktafonlardan iborat edi. 60-yillarning ikkinchi yarmidan esa —elektron (yoki kompyuter)texnologiyasi yuzaga kela boshladi va axborotni shaklini emas, mazmunini o'zgartirishga urg'u berila boshlandi.

Axborot texnologiyasining keyingi taraqqiyoti asosan kommunikatsiya vositalari bilan bog'liq.

Xorijiy mutaxassislar axborot texnologiyalari rivojlanishiing beshta asosiy tendentsiyasini ajratib ko'rsatadi:

1. Axborot mahsulotlarining murakkablashuvi. Axborot vositasi ko'rinishidagi axborot mahsuloti, ekspert ta'minoti txizmatining ma'lumotlar bazasi strategik ahamiyat kasb eta boradi. Turli shakldagi (nutq, ma'lumot, tasvir) axborot mahsulotlari eshitish, ko'rish va anglash uchun foydalanuvchining talabiga ko'ra ishlab chiqiladi hamda unga qulay vaqtda va shaklda mahsulotni yetkazib berish vositasi mavjud bo'ladi.

2. Birgalikda harakat qilish qobiliyati. Axborot mahsulotining ahamiyati oshib borishi bilan mazkur mahsulotlarni komp'yuter va inson yoki axborot tizimlari o'rtasida ideal tarzda almashuvini o'tkazish imkoniyati ilg'or texnologik

muammo kasb etadi.

3. Oraliq bo'g'inlarni tugatish. Birgalikda harakatlanish qobiliyatining rivojlanishi axborot mahsulotlari almashish jarayonining takomillashuviga, so'ngra, axborot manbai yo'lidan iste'molchiga qarab oraliq bo'g'inlar tugatiladi.

4. Globallashtirish. Tashkilot yo'ldosh aloqa va Internet tarmog'ida foydalanib axborot texnologiyalari yordamida hohlagan joyda va hohlagan paytda ish olib borish mumkin.

5. Konvergenstsiya (uyg'unlashish). Konvergenstsiya AATning zamonaviy rivojlanish jarayonining oxirigi bosqichi sifatida ko'rib chiqiladi. Bunda mahsulotlaro va xizmatlar, axborot va dam olish, shuningdek, ovozli, raqamli hamda videosignallarni uzatish kabi ish rejimlari o'rtisidagi farq yo'qoladi.

O'zbekiston Respublikasining axborotlashtirish dasturi.

Kompyuter texnikasi va axborot texnologiyalari rivojlanishi turli xil axborotlardan foydalanishga qaratilgan va axborot jamiyati nomini olgan jamiyatning rivojlanish omili bo'ldi, ya'ni axborot ishlab chiqarish jamiyatni rivojlantirishda harakatlantiruvchi kuchna aylandi. Axborot texnologiyalari bugungi kunda hayotimizning hamma sohalarini qamrab olgan.

O'zbekistonda axborot texnologiyasi rivojlanishini tezlashtirish va zamonaviy darajaga ko'tarish uchun Respublika Vazirlar Mahkamasi tomonidan bir qator qarorlar qabul qilingan. 1994 yilda Vazirlar Mahkamasi O'zbekiston Respublikasini axborotlashtirish kontseptsiyasini qabul qilgan. Kontseptsiya asosida —O'zbekiston Respublikasini axborotlashtirish Dasturi ishlab chiqilgan. Dastur uch maqsadni o'z ichiga oladi:

- 1) milliy axborot hisoblash tarmog'i;
- 2) EHM ni matematik va dasturiy ta'minlash;
- 3) shaxsiy kompyuter.

Milliy axborot hisoblash tarmog'i davlat aloqa tizimi negizida ishlaydigan va yagona qoidalarga rioya qilish asosida qurilgan davlat va idoraviy xususiyatga ega bo'lgan axborot hisoblash tarmoqlarini mujassamlashgan ochiq tizimdir. EHM ni matematik va dasturiy ta'minlash axborotlashtirish masalalarining matematik modeli, hisoblash algoritmi va yechim olish dasturini yaratishdan iboratdir.

Dasturning shaxsiy komp yuter qismi informatika industriyasini yaratish va rivojlantirishdan iborat.

Istalgan biror ob'ektni yoki jarayonni boshqarish uchun oldindan tayyorlangan axborotlardan foydalaniladi, ya'ni axborot tizimi yaratiladi. Axborot tizimi boshqaruv funktsiyasini amalga oshirish uchun turli xil axborotlarni yig'ish, uzatish va qayta ishlash bo'yicha ma'lumotlar va kommunikatsion tizimdir. Axborot tizimining texnik asosi bo'lib kompyuterlar hisoblanadi.

Informatika 1960 – yillarda Frantsiyada EHM yordamida axborotlarni qayta ishlash bilan shug'ullanuvchi sohani ifodalovchi atama sifatida yuzaga keldi. Informatika atamasi lotincha —informatik so'zidan kelib chiqqan bo'lib, tushuntirish, xabar qilish, bayon etish ma'nolarini anglatadi. Informatikaning fan sifatida ajralib chiqishi birinchi navbatda komp yuter

texnikasining rivojlanishi bilan uzviy bog'liqdir. Informatika inson faoliyatining turli jabhalardagi axborotlarni izlash, to'plash, qayta ishlash qonuniyatlarini EHM tizimlariga asoslangan holda o'rganuvchi fandir.

Informatikaning asosiy uchta yo'nalishi mavjud.

Birinchi yo'nalish axborotni uzatish, yig'ish va qayta ishlashning texnik vositalarini rivojlantirish nazariyasi bilan bog'liq. U o'z ichiga hisoblash komplekslarini, lokal va global hisoblash tarmoqlari, aloqa nazariyasini olgan keng ilmiy-ommaviy sohadir.

Ikkinchi yo'nalish ma'lumotlarni qayta ishlash bo'yicha har xil amaliy vazifalarni hal etish yuzasidan turli kategoriyadagi foydalanuvchilar uchun texnik vositalar bilan samarali ishlashni tashkil qilish imkonini beradigan, dasturiy ta'minotni ishlab chiqishga yo'naltirilgan, matematik va amaliy fanlar kompleksini o'z ichiga olgan dasturlashtirishdir.

Bu yo'nalishga algoritmlashtirish tillari nazariyasi, ma'lumotlarni tashkil etish, saqlash, qidirish va qayta ishlash nazariyasi, tizimli hamda amaliy dasturlashtirish nazariyasi kiradi.

Uchinchi yo'nalish – avtomatlashtirilgan usulda turli darajadagi vazifalarni hal etish modellari, algoritmlari, tartibi, texnologiyasini ishlab chiqish va tashkil qilishdir.

Informatika o'zaro aloqador uch qismdan iborat: algoritmik vositalar, dasturiy vositalar va texnik vositalar.

Informatikaning moddiy-texnik bazasi boshqaruv faoliyatining turli sohasi va darajalarida ishlayotgan mutaxassislar uchun avtomatlashtirilgan ish joylaridan keng foydalanish imkonini beradi va shuningdek, ekspert – professional (ekspert tizimi) darajasida qaror qabul qilishga qodir bo'lgan ixtisoslashtirilgan predmet sohasida hisoblash tizimini va axborot-kommunikatsiya tarmog'ini yaratish imkonini yuzaga keladi.

Informatika fanining *asosiy ashyosi* axborotdir. Axborot atrof-muhit, ob'ektlar va hodisalar, ularning o'lchamlari, xosiyatlari va holatlari to'g'risidagi ma'lumotdir. Ma'lumot bilan axborot bir-biridan farq qiladi. *Axborot foydalaniladigan ma'lumotdir.*

Informatikaning tarkibiy qismi bo'lib *axborot texnologiyasi* va *telekommunikatsiya* hisoblanadi.

Axborot va uning xossalari.

Axborot texnologiyasi ob'ekt, jarayon yoki hodisa haqida yangi sifat axboroti olish uchun ma'lumotlar yig'ish, qayta ishlash va uzatish (boshlang'ich axborot) vositasi va uslublari jamlanmasidan foydalanadigan jarayondir. *Telekommunikatsiya* kompyuter tarmoqlari va zamonaviy texnik aloqa vositalari negizida ma'lumotlarni masofaga uzatishdir. «Axborot» so'zi lotincha —information so'zidan olingan bo'lib, biror ish holati yoki kishi faoliyati haqida ma'lum qilish, xabar berish, biror narsa haqidagi ma'lumot, degan ma'noni anglatadi. Informatika nazariyasida saqlash, qayta tuzish va uzatish ob'ekti sanalgan barcha ma'lumotlar axborot deb yuritiladi. Axborot – bu, yaratuvchisi doirasida qolib ketmagan va xabarga aylangan, bilimlar noaniqligi, to'liqsizligi darajasini kamaytiradigan hamda og'zaki, yozma yoki boshqa usullar (shartli

signallar, texnik vositalar, hisoblash vositalari va x.k.) orqali ifodalash mumkin bo'lgan atrof-muhit (ob'ektlar, voqea-hodisalar) to'g'risidagi ma'lumotlardir.

Mazkur yo'nalishda quyidagilar muhim sanaladi:

- axborot – bu har qanday ma'lumot emas, balki u mavjud noaniqliklarni kamaytiruvchi yangi bir ma'lumotdir;

- axborot uni yaratuvchisidan tashqarida mavjud bo'ladi, u o'z yaratuvchisidan uzoqlashgan, inson tafakkurida aks etgan bilimdir;

- axborot xabarga aylanadi, chunki u belgilar ko'rinishida ma'lum bir tilda ifodalanadi;

- xabar moddiy tashuvchiga yozib qo'yilishi mumkin (xabar axborotni uzatish shaklidir);

- xabar uning muallifi ishtirokisiz aks ettirilishi mumkin;

- u jamoat kommunikatsiyasi kanallari orqali uzatiladi.

Axborotning asosiy xossalari:

1. to'liqlik;
2. yaroqlilik;
3. ishonchlilik;
4. dolzarblik;
5. tushunarlilik.

Axborotning ifodalanish shakllari va uning turlari. Axborotning muhim xarakteristikalaridan biri uning adekvatligi hisoblanadi. Axborotning adekvatligi – olingan axborot yordamida yaratilgan obrazning real ob'ekt, jarayon, hodisa va shunga o'xshashlarga mosligining ma'lum darajasi. Axborotning adekvatligi uchta shaklda ifodalanishi mumkin: semantik, sintaktik va pragmatik.

Semantik (ma'noli) adekvatlik – ob'ektning uning obraziga (qiyofasiga) muvofiqlik darajasini aniqlaydi. Semantik nuqtai nazar axborotning ma'noli mazmunini hisoblashni ko'zlaydi. Bunda axborot aks ettirgan ma'lumotlar tahlil qilinadi, ma'nolar bog'liqligi ko'riladi.. Masalan, axborotni kodlar orqali ifodalashni ko'rsatish mumkin.

Sintaktik adekvatlik – axborotning mazmuniga tegmagan holda, uning rasmiy-strukturaviy xarakteristikalarini ifodalaydi. Sintaktik darajadagi axborotni ifodalash usulida axborot elituvchi turi, uzatish va qayta ishlash tezligi, ifodalash kodining o'lchamlari, bu kodlarni o'zgartirish aniqliligi va ishonchliligi hisobga olinadi. Axborotning mazmuniga ahamiyat berilmaganligi sababi, bunday axborot ma'lumot deb ataladi.

Pragmatik (foydalanuvchanlik) adekvatlik – axborot bilan foydalanuvchining munosabatlarini aks ettiradi, axborotni uning asosida amalga oshiriladigan boshqarish sistemasi maqsadiga muvofiqligini ifodalaydi. Axborotning pragmatik xususiyatlari faqat axborot (ob'ekt), foylanuvchi va boshqarish maqsadlarining umumiyligida namoyon bo'ladi. Adekvatlikning ushbu shakli axborotdan amaliy foylanish bilan bevosita bog'langan. Informatikada asosiy masala bo'lib hisoblash texnikasi qurilmalaridan axborotni saqlash, qayta ishlash va uzatishda qanday foydalanish hisoblanadi.

Shuning uchun informatikada axborotning ikki xil turi bilan ish ko'riladi, ya'ni analog va raqamli. Ko'pgina hisoblash texnikasi qurilmalari raqamli axborotni qayta ishlaydi. Analog axborotni raqamli axborotga o'zgartiruvchi maxsus qurilmalar mavjud bo'lib, bunday o'zgartirishni *analog-raqamli o'zgartirish* deyiladi.

Kompyuter faqat raqamli ko'rinishdagi axborotni qayta ishlaydi. Barcha boshqa turdagi axborot (ovozlar, tasvirlar va boshq.) komp yuterda qayta ishlanishi uchun raqamli ko'rinishga keltirilishi zarur. Ovozni raqamli ko'rinishga o'zgartirish uchun kichik vaqt oralig'ida ovoz intensivligini o'lchash va har bir o'lchash natijalarini raqam ko'rinishida aks ettirish zarur. Kompyuter dasturi yordamida olingan axborotni qayta ishlab, hosil bo'lgan natijani ovoz shakliga qaytarish mumkin. Komp yuterda matni axborotni qayta ishlash uchun matni kompyuterga kiritilayotganda har bir xarf ma'lum bir raqam bilan kodlanadi. Tashqi qurilmalarga (monitor ekrani yoki printer) chiqarilayotganda esa inson qabul qilish uchun ushbu raqamlar orqali xarflarning tasviri quriladi. Xarflar to'plami va raqamlar o'rtasidagi moslik belgilarni kodlashtirish deb ataladi. Kompyuterda barcha raqamlar 0 va 1 orqali ifodalanadi. Kompyuter ikkilik sanoq sistemasida ishlaydi. Kompyuterda axborot birligining o'lchovi bo'lib bit, ya'ni 0 yoki 1 qiymatni qabul qilishi mumkin bo'lgan ikkilik razryad hisoblanadi. Komp yuter komandalari alohida bitlar bilan emas, balki sakkiz bit bilan birgalikda ishlaydi. Sakkizta ketma-ket bit bir baytni tashkil etadi. Baytlar yordamida raqamli ko'rinishda ifodalangan har qanday axborotni kodlashtirish mumkin. Bir baytda 256 xil belgilardan birining qiymatini kodlashtirish mumkin ($256=2^8$) bo'ladi. Baytning qiymati uchun undagi bitlarning joylashgan o'rni muhimdir. Axborotda qatnashgan har qanday belgi 1 bayt hajmli deb hisoblanadi. Masalan, "Sh" harfi – 1 bayt, "Kitob" – 5 bayt hajmga ega. Bir bayt 0 dan 255 qiymatni qabul qilishi mumkin. Axborotning yirik birliklari:

Bayt – ketma-ket yozilgan 8 ta bitdan iborat bo'lib, uning xar birida bittadan belgini ifodalash mumkin. Bunday belgilar soni $2^8=256$ tadan iborat. Bundan tashqari axborotlarning hosilaviy o'lchov birliklari xam mavjud bo'lib, ular *kilobayt(Kb)*, *megobayt(Mb)*, *Gigobayt(Gb)*, *eksobayt(Eb)* lardan iboratdir:

$$1 \text{ Kb} = 2^{10} = 1024 \text{ bayt};$$

$$1 \text{ Mb} = 2^{10} = 1024 \text{ Kb};$$

$$1 \text{ Gb} = 2^{10} = 1024 \text{ Mb};$$

$$1 \text{ Eb} = 2^{10} = 1024 \text{ Gb};$$

Misol sifatida *Ekonometrika* so'zi nechta belgidan iborat bo'lsa kompyuter xotirasida shuncha baytni, ya'ni 12 baytni yoki $12 \times 8 = 96$ bitni egallaydi.

2 . Nazariy ma'lumot yuzasidan berilgan testlarni bajarish.

1.Informatika fani nimani o'rganadi?

A) Axborotlarni o'lchashni o'rganadi.

B) Axborotlarni qabul qilish, saqlash, qayta ishlash va uzatish usullari va qonun - qoidalarini o'rganadi.

C) Belgilar to'plamini o'rganadi.

D) Atrof muxitdagi ob'ektlarni o'rganadi.

2. Informatika atamasi qaysi so'zdan olingan?

A) ingliz

B) nemis

C) fransuz

D) lotin

3. Ijtimoiy hayotga tatbiqan axborot nimani anglatadi?

A) Tushuntirish, anglatish, bildirishni

B) Predmetlar, faktlar, kishilar, hodisalar haqidagi ma'lumotni

C) Bayon etish, tushuntirish, izhor etishni

D) Og'zaki va yozma uzatilgan ma'lumotni

4. Axborot qidirish, yig'ish, qayta ishlash, saqlash va tarqatish kabi jarayonlar nimani ifodalaydi?

A) Axborot tizimini

B) Informasion jarayonlarni

C) Axborot resurslarini

D) Axborot madaniyatini

5. Informasion jarayonlarni amalga oshiruvchi, tartibga solingan hujjatlar hamda axborot texnologiyalari majmui nima deb nomlanadi?

A) Axborot resursi

B) Axborot tizimi

C) Ijtimoiy-siyosiy axborot

D) Axborotlashgan jamiyat

6. Ijtimoiy-siyosiy axborotlarni ishlab chiqish va tarqatishda nimalar muhim rol o'ynaydi?

A) Texnika va ommaviy axborot vositalari

B) Fan va ommaviy axborot vositalari

C) Siyosiy partiyalar va ommaviy axborot vositalari

D) Texnika, texnologiya va ommaviy axborot vositalari

7. Axborot madaniyatining mohiyati nimada?

A) Texnik-axborot vositalaridan oqilona foydalanishda

B) Texnik-axborot vositalaridan oqilona foydalanish hamda muayyan qadriyatlarni o'zlashtirganlikda

C) Milliy va umuminsoniy qadriyatlarni o'zlashtirganlikda

D) Progressiv mazmunga ega qadriyatlarni bilishda

8. Axborot uzatishda muhim qurol bo'lib nima xizmat qiladi?

A) Xabarlar

B) So'z

C) Raqamlar va dalilllar

D) Ma'lumotlar

9. Axborotlashgan jamiyatda nima muhim o'rinni egallaydi?

A) Texnika va texnologiya

B) Axborot va bilim

C) Bilim va texnologiya

D) Kapital va mehnat

10. Axborot bu - ...

- A) Hozirgi zamon sivilizasiyasining o'ziga xos yutug'i
- B) Xalqlar, davlatlar, mamlakatlarni yaqinlashtiradigan hodisa
- C) Jahon siyosiy ijtimoiy jarayonlarini boshqarishga xizmat qilingan hodisa
- D) Hamma javoblar to'g'ri

Nazorat savollari:

1. "Axborot" deb nima tushuniladi?
2. "Ma'lumotlar" deb nimani tushunamiz?
3. Axborotni qanday belgilar bo'yicha tavsiflash mumkin?
4. Eng kichik axborot o'lchov birligi nima?

2-amaliy mashg'ulot. Kimyo injineriingda axborot texnologiyalari faniga kirish.

Ishdan maqsad. Axborot texnologiyalari fani, uning ishlatilish sohalari, axborotning turlari, xossalari bilan tanishish.

Vazifa.

1. Berilgan nazariy materiallarni o'rganish.
2. Nazariy ma'lumot yuzasidan berilgan topshiriqlarni bajarish.

1. Berilgan nazariy materiallarni o'rganish.

Axborot texnologiyalari fanini o'qitishdan maqsad, zamonaviy axborot texnologiyalar haqida tushuncha berish, zamonaviy dasturiy vositalar, zamonaviy kompyuter dasturlarida ishlashni bilishni o'rgatish va ta'lim jarayoniga mos o'quv resurlarini kompyuterdan foydalanib, yaratish. Ta'lim jarayoniga mos elektron darsliklarini, o'quv elektron resurslarni, on-layn tizimida va off-layn tizimida ishlaydigan vositalarni yaratishni o'rgatishdan iboratdir. Bu fan o'zining oldiga bir necha vazifalarni qo'yib olgan, ya'ni birinchi istiqbol rejalardan birinchisi, yaratilgan va yaratilayotgan yangi kompyuter dasturlaridan xabardor bo'lib turish, mavjudlarini amaliyotda qo'llashni o'rgatish.

Axborot kommunikatsion texnologiyalar-bu audio, video va kommunikatsiya vositalarini axborotlarni hosil qilish, olish, ishlab chiqish va qayta ishlash hamda tarqatish jarayonidir. Bu jarayonda inson resurlari va texnik resurslar qatnashadi. Biri ikkinchilari uchun xizmat qiladi. Ya'ni texnik resurslar insonlarni axborotni olishlari uchun va boshqa odamlarga tarqatish uchun xizmat qiladi.

Texnologiya-maxorat, san'at degan ma'noga ega, logos esa-ta'lim, — Axborot texnologiyalari kursining maqsadi AKTdan ta'limda, jumladan, xorijiy til va adabiyoti fanini o'qitishda qo'llashni o'rgatish, talabalarning esa ta'limda qo'llaniladigan dasturlar, dasturiy vositalardan o'z mutahassisliklarida foydalanishni bilish ta'limda qo'llashni bilish, ularni o'rganish va foydalanish hamda qo'llashning zamonaviy tendensiyalardan biri, bu axborot texnologiyalarining turlari bilan tanishish, ularning qaysi sohaga oidligini farqlay bilish va ilg'or texnologiyalarni, bir tildan ikkinchi tilga o'girish texnologiyalari, matndan giper-matnga o'girish texnologiyalarini, taqdimot yaratish

texnologiyalarini, ekspert tizimlar texnologiyalari, elektron xukumat, masofaviy ta'lim texnologiyalari, video-kurslar, elektron resurslar, on-layn taqdimotlar va boshqa ta'lim resurlarini yaratuvchi dasturlar va dasturiy vositalarni o'rganish maqsadga muvofiqdir. Ta'lim jarayoniga axborot kommunikatsion texnologiyalarini qo'llash ta'lim jarayonini mazmunli, chiroyli, unumli, qulay, vaqtni tejovchi va o'zlashtirish sifatini oshiruvchi etib amalga oshirib beradi. O'qituvchining dars berish jarayonini unumli, vaqtni tejovchi va mazmunli tarzda o'tkazib beradi. O'zlashtirish sifatida vaqtning qo'llanish evaziga, ya'ni kompyuter dasturlari qo'shimcha kanallari orqali ko'z xotirasini qabul qilish jarayonini ishga tushirib, ularga ta'sirni kuchytirib, rangli rasmlar, xarakatli animatsiyalarni o'qitish jarayoniga tadbiq etib, tasavvurni to'g'ri shakllantirish orqali bilish jarayonini, tafakkur etish qobiliyatini rivojlantiradi.

Ta'lim jarayoniga axborot kommunikatsiya texnologiyalarini qo'llashning dolzarb vazifalaridan biri, bu elektron darsliklarni yaratuvchi, elektron resurlarni, taqdimotlarni, zamonaviy ta'lim vositalarini yaratuvchi dasturlarni o'rgatishdan iborat. Axborot katta jadallik bilan kirib kelmoqda, kim axborot bilan birinchi tanish bo'lsa, o'sha voqeani boshqaradi, jarayonni boshqarish uni qo'lida, ya'ni u dunyoni boshqaradi, deyishadi.

Katta hajmdagi axborotlarni etkazish, uni samarali egasiga berish, vaqtni tejash vazifasini bajarishda axborot texnologiyalar yordamga keladi. Konsepsiya-bu ta'limot deganidir.

1. AKT. O'zbekistonda AKT sohasini rivojlantirishdagi davlat siyosatining asosiy yo'nalishlari, amaldagi qonunlar, farmonlar va qarorlar.

2. Axborot va uning turlari, xossalari, uzatilishi, o'lchov birliklari.

3. Axborotni kodlash.

4. Ma'lumotlarning kompyuter xotirasida tasvirlanishi/ O'zbekistonda AKT sohasini rivojlantirish uchun 2005 yidagi bir qator qororlar va qonunlar isbot bo'la oladi. XXI asr axborot texnologiyalari asri bo'lgani uchun ta'limni ham bu texnologiyalarisiz tasavvur etish qiyin. Hukumatimiz tomonidan Axborotlashtirish to'g'risidagi, Elektron hukumat tashkil etish, uni rivojlantirish haqidagi bir qator qonunlar jamiyatni rivojlantirish, hukumatni boshqarishga yordam qilishga yo'naltirgan.

Texnologiyalar. —Texnologiya grekcha so'z bo'lib mohirlik, ustalik, biror ishni uddalay olishni anglatadi. Bu ma'lum bir jarayonga nisbatan qo'llanilgan. Jarayon deganda esa maqsadga erishishga yo'naltirilgan xatti-harakatlar majmui tushunilgan. Ushbu jarayon inson tomonidan tanlangan strategiya bilan belgilanadi va turli xildagi vositalar, usullar yordamida amalga oshiriladi. Umumiy hollarda texnologiya deganda, mahsulotni ishlab chiqarish jarayonida amalga oshiriladigan xomashyo, material yoki yarim tayyor mahsulot shakli, xususiyati, holatining o'zgarishi, uni qayta ishlash, tayyorlash usullarining majmui tushuniladi. Bu biror bir ishni yuqori darajada uddalash deganidir. Axborot texnologiyalari to'g'risida gap ketganda, material sifati ham, mahsulot sifatida ham axborot ishtirok etadi. Biroq bu ob'ekt, jarayon yoki hodisa to'g'risidagi sifat jihatidan yangi ma'lumot bo'ladi. Texnologiya xodimning axborot bilan ishlash usuli va uslubi hamda texnik vositalar orqali namoyon

bo‘ladi.

Axborot texnologiyasi – ob‘ektning (axborot mahsulotining) holati, jarayon yoki voqeaning yangi xususiyati to‘g‘risida axborot olish uchun ma‘lumotlarni yig‘ish, qayta ishlash va uzatish vositalari va usullari majmuidan foydalaniladigan jarayondir.

Moddiy ishlab chiqarish texnologiyasining maqsadi – inson yoki tizimning ehtiyojini qondiruvchi mahsulot ishlab chiqarish sanaladi. Axborot texnologiyasining maqsadi esa – axborot ishlab chiqarish bo‘lib, uni tahlil etish va uning asosida biror bir harakatga qo‘l urish uchun tegishli qaror qabul qilish.

Axborotlarni yig‘ish, uzatish, to‘plash, saqlash, taqdim etish va foydalanish uslublari va usullari tizimi **axborot texnologiyasi** deb yuritiladi. Axborot texnologiyasi avtomatlashgan va an‘anaviy (qog‘oz) ko‘rinishiga amalga oshiriladi. Avtomatlashtirish hajmi va texnik vositalardan foydalanish turi aniq bir texnologiyaning mohiyatiga bog‘liq.

Avtomatlashtirish – bu inson ish faoliyatini mashina va mexanizmlar bilan almashtirish demakdir. U texnik, iashkiliy va iqtisodiy mazmundagi hattiharakatlar hamda tadbirlar kompleksidan iborat bo‘ladi, ishlab chiqarish jarayoni, boshqaruv jarayonining u yoki bu ishini amalga oshirishda inson ishtirokini qisman yoki butunlay cheklash imkonini beradi.

Moddiy va axborot texnologiyasining asosiy komponentlarini qiyoslash 2.1-jadvalda berilgan.

2.1-jadval.

Mahsulot ishlab chiqarish uchun texnologiya komponentlari	
<i>Moddiy mahsulot</i>	<i>Axborot mahsuloti</i>
Xom ashyo va materiallar tayyorlash	Ma‘lumotlar yoki boshlang‘ich axborotni yig‘ish
Moddiy mahsulot ishlab chiqarish	Ma‘lumotlarni qayta ishlash va yakuniy axborotga ega bo‘lish
Iste‘molchilarga ishlab chiqarilgan mahsulotni sotish	Uning asosida qaror qabul qilish uchun yakuniy axborotlar uzatish

Axborot texnologiyasining asosiy tavsifi.

Avtomatlashtirilgan axborot texnologiyasi (AAT) – boshqaruv vqazifalarini hal etish uchun tizimli tashkil etilgan axborot jarayonlarini amalga oshirish usul va vositalari majmuidir. U hisoblash texnikasi va aloqa vositalaridan foydalaniladigan rivojlangan dasturiy ta‘minotni qo‘llash bazasida bajariladi. Axborot texnologiyasining amalga oshirishdagi texnik vositalarning asosiy qisini komp yuter texnikasi tashkil etgan uchun axborot texnologiyasi, ayniqsa zamonaviy axborot texnologiyasi deganda komp yuter axborot texnologiyasi tushuniladi.

2.Zamonaviy axborot texnologiyasi

Zamonaviy axborot texnologiyasi (ZAT) (kompyuter axborot texnologiyasi) – shaxsiy komp yuter va telekommunikatsiya vositalaridan foydalanuvchi uchun qulay —interfeysli axborot texnologiyasidir. Ma'lum bir turdagi komp yuter uchun mo'ljallangan bir yoki bir necha o'zaro bog'liq dasturiy mahsulotlar zamonaviy axborot texnologiyalarining vositasi sanaladi. Zamonaviy axborot texnologiyasining asosiy elementlari.

ZAT ning asosiy elementlari quyidagicha:

- ma'lum bir vaqt davomida axborotni kiritish va qayta o'zgartirish;
- tasvirni kiritish va unga ishlov berish;
- signal axboroti paydo bo'lgan yerda uni qayta ishlash;
- og'zaki axborotni qayta ishlash;
- foydalanuvchining ShK bilan faol muloqoti;
- turli axborot tizimlarida mashinali modellashtirish;
- axborot almashuvining tarmoq texnologiyasi (dialog yuritish, video va telekommunikatsiya, elektron pocha, vidioteka va x.z.);
- taqsimlangan tarmoq tizimlarida ma'lumotlarni mul'tiprotssessor asosida qayta ishlash;
- mahalliy, mintaqaviy va xalqaro tarmoqlar bo'yicha axborotni tezkor tarqatish.

Axborot texnologiyasining bazaviy texnologiyasi bo'lib quyidagilar hisoblanadi:

- a) texnik ta'minot texnologiyasi;
- b) telekommunikatsiya texnologiyasi;
- v) dasturiy ta'minot texnologiyasi.

Bu texnologiyalar hisoblash tizimlari va tarmoqlari arxitekturasining aniq variantlari doirasida birgalikda harakat qiladi va birlashadi. Ularni ayrimlari axborot texnologiyasi rivojlanishida hal qiluvchi rol o'ynaydi.

Axborot texnologiyalari taraqqiy etishining asosiy bosqichlari va rivojlanish tendensiyasi.

XIX asrning ikkinchi yarmigachi "qo'l" axborot texnologiyasi rivojlangan bo'lib, uning asosini pero, siyohdon va buxgalteriya daftari tashkil etgan. Kommunikatsiya (aloqa) paket (rasmiy xujjatlar solingan konvert) yuborish orqali amalga oshirilardi.

"Qo'l" axborot texnologiyasi o'rniga XIX asr oxirida "mexanik" texnologiya kirib keldi. Yozuv mashinasi, telefon, diktafonning kashf etilishi, jamoa pochta tizimining takomillashuvi – bular bari avvaliga axborotni qayta ishlash texnologiyasida, so'ng ish mahsuldorligida sezilarli o'zgarishlar yuz berishiga zamin bo'ldi.

XX asrning 40 - 60–yillarida —elektrli texnologiyasi paydo bo'lib, u yechib almashtiriladigan elementlarga ega elektr yozuv mashinkalari, oddiy qog'ozdan foydalanuvchi nusxa ko'chirish mashinasi, portativ diktafonlardan iborat edi. 60-yillarning ikkinchi yarmidan esa —elektron (yoki kompyuter)texnologiyasi yuzaga kela boshladi va axborotni shaklini emas, mazmunini o'zgartirishga urg'u berila boshlandi.

Axborot texnologiyasining keyingi taraqqiyoti asosan kommunikatsiya vositalari bilan bog'liq.

Xorijiy mutaxassislar axborot texnologiyalari rivojlanishiing beshta asosiy tendentsiyasini ajratib ko'rsatadi:

1. Axborot mahsulotlarining murakkablashuvi. Axborot vositasi ko'rinishidagi axborot mahsuloti, ekspert ta'minoti txizmatining ma'lumotlar bazasi strategik ahamiyat kasb eta boradi. Turli shakldagi (nutq, ma'lumot, tasvir) axborot mahsulotlari eshitish, ko'rish va anglash uchun foydalanuvchining talabiga ko'ra ishlab chiqiladi hamda unga qulay vaqtda va shaklda mahsulotni yetkazib berish vositasi mavjud bo'ladi.

2. Birgalikda harakat qilish qobiliyati. Axborot mahsulotining ahamiyati oshib borishi bilan mazkur mahsulotlarni komp yuter va inson yoki axborot tizimlari o'rtasida ideal tarzda almashuvini o'tkazish imkoniyati ilg'or texnologik muammo kasb etadi.

3. Oraliq bo'g'inlarni tugatish. Birgalikda harakatlanish qobiliyatining rivojlanishi axborot mahsulotlari almashish jarayonining takomillashuviga, so'ngra, axborot manbai yo'ldan iste'molchiga qarab oraliq bo'g'inlar tugatiladi.

4. Globallashtirish. Tashkilot yo'ldosh aloqa va Internet tarmog'ida foydalanib axborot texnologiyalari yordamida hohlagan joyda va hohlagan paytda ish olib borish mumkin.

5. Konvergenstsiya (uyg'unlashish). Konvergenstsiya AATning zamonaviy rivojlanish jarayonining oxirigi bosqichi sifatida ko'rib chiqiladi. Bunda mahsulotlaro va xizmatlar, axborot va dam olish, shuningdek, ovozli, raqamli hamda videosignallarni uzatish kabi ish rejimlari o'rtisidagi farq yo'qoladi.

O'zbekiston Respublikasining axborotlashtirish dasturi.

Kompyuter texnikasi va axborot texnologiyalari rivojlanishi turli xil axborotlardan foydalanishga qaratilgan va axborot jamiyati nomini olgan jamiyatning rivojlanish omili bo'ldi, ya'ni axborot ishlab chiqarish jamiyatni rivojlantirishda harakatlantiruvchi kuchna aylandi. Axborot texnologiyalari bugungi kunda hayotimizning hamma sohalarini qamrab olgan.

O'zbekistonda axborot texnologiyasi rivojlanishini tezlashtirish va zamonaviy darajaga ko'tarish uchun Respublika Vazirlar Mahkamasi tomonidan bir qator qarorlar qabul qilingan. 1994 yilda Vazirlar Mahkamasi O'zbekiston Respublikasini axborotlashtirish kontseptsiyasini qabul qilgan. Kontseptsiya asosida —O'zbekiston Respublkasini axborotlashtirish Dasturi|| ishlab chiqilgan. Dastur uch maqsadni o'z ichiga oladi:

- 1) milliy axborot hisoblash tarmog'i;
- 2) EHM ni matematik va dasturiy ta'minlash;
- 3) shaxsiy kompyuter.

Milliy axborot hisoblash tarmog'i davlat aloqa tizimi negizida ishlaydigan va yagona qoidalarga rioya qilish asosida qurilgan davlat va idoraviy xususiyatga ega bo'lgan axborot hisoblash tarmoqlarini mujassamlashgan ochiq tizimidir. EHM ni matematik va dasturiy ta'minlash axborotlashtirish masalalarining matematik modeli, hisoblash algoritmi va yechim olish dasturini yaratishdan iboratdir.

Dasturning shaxsiy komp yuter qismi informatika industriyasini yaratish va rivojlantirishdan iborat.

Istalgan biror ob'ektni yoki jarayonni boshqarish uchun oldindan tayyorlangan axborotlardan foydalaniladi, ya'ni axborot tizimi yaratiladi. Axborot tizimi boshqaruv funktsiyasini amalga oshirish uchun turli xil axborotlarni yig'ish, uzatish va qayta ishlash bo'yicha ma'lumotlar va kommunikatsion tizimdir. Axborot tizimining texnik asosi bo'lib kompyuterlar hisoblanadi.

Informatika 1960 – yillarda Frantsiyada EHM yordamida axborotlarni qayta ishlash bilan shug'ullanuvchi sohani ifodalovchi atama sifatida yuzaga keldi. Informatika atamasi lotincha —informatik so'zidan kelib chiqqan bo'lib, tushuntirish, xabar qilish, bayon etish ma'nolarini anglatadi. Informatikaning fan sifatida ajralib chiqishi birinchi navbatda komp yuter texnikasining rivojlanishi bilan uzviy bog'liqdir. Informatika inson faoliyatining turli jabhalardagi axborotlarni izlash, to'plash, qayta ishlash qonuniyatlarini EHM tizimlariga asoslangan holda o'rganuvchi fandır.

Informatikaning asosiy uchta yo'nalishi mavjud.

Birinchi yo'nalish axborotni uzatish, yig'ish va qayta ishlashning texnik vositalarini rivojlantirish nazariyasi bilan bog'liq. U o'z ichiga hisoblash komplekslarini, lokal va global hisoblash tarmoqlari, aloqa nazariyasini olgan keng ilmiy-ommaviy sohadir.

Ikkinchi yo'nalish ma'lumotlarni qayta ishlash bo'yicha har xil amaliy vazifalarni hal etish yuzasidan turli kategoriyadagi foydalanuvchilar uchun texnik vositalar bilan samarali ishlashni tashkil qilish imkonini beradigan, dasturiy ta'minotni ishlab chiqishga yo'naltirilgan, matematik va amaliy fanlar kompleksini o'z ichiga olgan dasturlashtirishdir.

Bu yo'nalishga algoritmlashtirish tillari nazariyasi, ma'lumotlarni tashkil etish, saqlash, qidirish va qayta ishlash nazariyasi, tizimli hamda amaliy dasturlashtirish nazariyasi kiradi.

Uchinchi yo'nalish – avtomatlashtirilgan usulda turli darajadagi vazifalarni hal etish modellari, algoritmlari, tartibi, texnologiyasini ishlab chiqish va tashkil qilishdir.

Informatika o'zaro aloqador uch qismdan iborat: algoritmik vositalar, dasturiy vositalar va texnik vositalar.

Informatikaning moddiy-texnik bazasi boshqaruv faoliyatining turli sohasi va darajalarida ishlayotgan mutaxassislar uchun avtomatlashtirilgan ish joylaridan keng foydalanish inkonini beradi va shuningdek, ekspert – professional (ekspert tizimi) darajasida qaror qabul qilishga qodir bo'lgan ixtisoslashtirilgan predmet sohasida hisoblash tizimini va axborot-kommunikatsiya tarmog'ini yaratish imkonini yuzaga keladi.

Informatika fanining *asosiy ashyosi* axborotdir. Axborot atrof-muhit, ob'ektlar va hodisalar, ularning o'lchamlari, xosiyatlari va holatlari to'g'risidagi ma'lumotdir. Ma'lumot bilan axborot bir-biridan farq qiladi. *Axborot foydalaniladigan ma'lumotdir.*

Informatikaning tarkibiy qismi bo'lib *axborot texnologiyasi* va *telekommunikatsiya* hisoblanadi.

Axborot va uning xossalari.

Axborot texnologiyasi ob'ekt, jarayon yoki hodisa haqida yangi sifat

axboroti olish uchun ma'lumotlar yig'ish, qayta ishlash va uzatish (boshlang'ich axborot) vositasi va uslublari jamlanmasidan foydalanadigan jarayondir. *Telekommunikatsiya* kompyuter tarmoqlari va zamonaviy texnik aloqa vositalari negizida ma'lumotlarni masofaga uzatishdir. «Axborot» so'zi lotincha —information so'zidan olingan bo'lib, biror ish holati yoki kishi faoliyati haqida ma'lum qilish, xabar berish, biror narsa haqidagi ma'lumot, degan ma'noni anglatadi. Informatika nazariyasida saqlash, qayta tuzish va uzatish ob'ekti sanalgan barcha ma'lumotlar axborot deb yuritiladi. Axborot – bu, yaratuvchisi doirasida qolib ketmagan va xabarga aylangan, bilimlar noaniqligi, to'liqsizligi darajasini kamaytiradigan hamda og'zaki, yozma yoki boshqa usullar (shartli signallar, texnik vositalar, hisoblash vositalari va x.k.) orqali ifodalash mumkin bo'lgan atrof-muhit (ob'ektlar, voqea-hodisalar) to'g'risidagi ma'lumotlardir.

Mazkur yo'nalishda quyidagilar muhim sanaladi:

- axborot – bu har qanday ma'lumot emas, balki u mavjud noaniqliklarni kamaytiruvchi yangi bir ma'lumotdir;

- axborot uni yaratuvchisidan tashqarida mavjud bo'ladi, u o'z yaratuvchisidan uzoqlashgan, inson tafakkurida aks etgan bilimdir;

- axborot xabarga aylanadi, chunki u belgilar ko'rinishida ma'lum bir tilda ifodalanagan;

- xabar moddiy tashuvchiga yozib qo'yilishi mumkin (xabar axborotni uzatish shaklidir);

- xabar uning muallifi ishtirokisiz aks ettirilishi mumkin;

- u jamoat kommunikatsiyasi kanallari orqali uzatiladi.

Axborotning asosiy xossalari:

1. to'liqlik;
2. yaroqlilik;
3. ishonchlilik;
4. dolzarblik;
5. tushunarlilik.

Axborotning ifodalanish shakllari va uning turlari. Axborotning muhim xarakteristikalaridan biri uning adekvatligi hisoblanadi. Axborotning adekvatligi – olingan axborot yordamida yaratilgan obrazning real ob'ekt, jarayon, hodisa va shunga o'xshashlarga mosligining ma'lum darajasi. Axborotning adekvatligi uchta shaklda ifodalanishi mumkin: semantik, sintaktik va pragmatik.

Semantik (ma'noli) adekvatlik – ob'ektning uning obraziga (qiyofasiga) muvofiqlik darajasini aniqlaydi. Semantik nuqtai nazar axborotning ma'noli mazmunini hisoblashni ko'zlaydi. Bunda axborot aks ettirgan ma'lumotlar tahlil qilinadi, ma'nolar bog'liqligi ko'riladi.. Masalan, axborotni kodlar orqali ifodalashni ko'rsatish mumkin.

Sintaktik adekvatlik – axborotning mazmuniga tegmagan holda, uning rasmiy-strukturaviy xarakteristikalarini ifodalaydi. Sintaktik darajadagi axborotni ifodalash usulida axborot elituvchi turi, uzatish va qayta ishlash tezligi, ifodalash kodining o'lchamlari, bu kodlarni o'zgartirish aniqliligi va ishonchliligi hisobga

olinadi. Axborotning mazmuniga ahamiyat berilmaganligi sababi, bunday axborot ma'lumot deb ataladi.

Pragmatik (foydalanuvchanlik) adekvatlik – axborot bilan foydalanuvchining munosabatlarini aks ettiradi, axborotni uning asosida amalga oshiriladigan boshqarish sistemasi maqsadiga muvofiqligini ifodalaydi. Axborotning pragmatik xususiyatlari faqat axborot (ob'ekt), foylanuvchi va boshqarish maqsadlarining umumiyligida namoyon bo'ladi. Adekvatlikning ushbu shakli axborotdan amaliy foylanish bilan bevosita bog'langan. Informatikada asosiy masala bo'lib hisoblash texnikasi qurilmalaridan axborotni saqlash, qayta ishlash va uzatishda qanday foydalanish hisoblanadi. Shuning uchun informatikada axborotning ikki xil turi bilan ish ko'riladi, ya'ni analog va raqamli. Ko'pgina hisoblash texnikasi qurilmalari raqamli axborotni qayta ishlaydi. Analog axborotni raqamli axborotga o'zgartiruvchi maxsus qurilmalar mavjud bo'lib, bunday o'zgartirishni *analog-raqamli o'zgartirish* deyiladi.

Kompyuter faqat raqamli ko'rinishdagi axborotni qayta ishlaydi. Barcha boshqa turdagi axborot (ovozlar, tasvirlar va boshq.) komp yuterda qayta ishlanishi uchun raqamli ko'rinishga keltirilishi zarur. Ovozni raqamli ko'rinishga o'zgartirish uchun kichik vaqt oralig'ida ovoz intensivligini o'lchash va har bir o'lchash natijalarini raqam ko'rinishida aks ettirish zarur. Kompyuter dasturi yordamida olingan axborotni qayta ishlab, hosil bo'lgan natijani ovoz shakliga qaytarish mumkin. Komp yuterda matni axborotni qayta ishlash uchun matni kompyuterga kiritilayotganda har bir xarf ma'lum bir raqam bilan kodlanadi. Tashqi qurilmalarga (monitor ekrani yoki printer) chiqarilayotganda esa inson qabul qilish uchun ushbu raqamlar orqali xarflarning tasviri quriladi. Xarflar to'plami va raqamlar o'rtasidagi moslik belgilarni kodlashtirish deb ataladi. Kompyuterda barcha raqamlar 0 va 1 orqali ifodalanadi. Kompyuter ikkilik sanoq sistemasida ishlaydi. Kompyuterda axborot birligining o'lchovi bo'lib bit, ya'ni 0 yoki 1 qiymatni qabul qilishi mumkin bo'lgan ikkilik razryad hisoblanadi. Komp yuter komandalari alohida bitlar bilan emas, balki sakkiz bit bilan birgalikda ishlaydi. Sakkizta ketma-ket bit bir baytni tashkil etadi. Baytlar yordamida raqamli ko'rinishda ifodalangan har qanday axborotni kodlashtirish mumkin. Bir baytda 256 xil belgilardan birining qiymatini kodlashtirish mumkin ($256=2^8$) bo'ladi. Baytning qiymati uchun undagi bitlarning joylashgan o'rni muhimdir. Axborotda qatnashgan har qanday belgi 1 bayt hajmli deb hisoblanadi. Masalan, "Sh" harfi – 1 bayt, "Kitob" – 5 bayt hajmga ega. Bir bayt 0 dan 255 qiymatni qabul qilishi mumkin. Axborotning yirik birliklari:

Bayt – ketma-ket yozilgan 8 ta bitdan iborat bo'lib, uning xar birida bittadan belgini ifodalash mumkin. Bunday belgilar soni $2^8=256$ tadan iborat. Bundan tashqari axborotlarning hosilaviy o'lchov birliklari xam mavjud bo'lib, ular *kilobayt(Kb)*, *megobayt(Mb)*, *Gigobayt(Gb)*, *eksobayt(Eb)* lardan iboratdir:

1 Kb = 2^{10} = 1024 bayt;

1 Mb = 2^{10} = 1024 Kb;

1 Gb = 2^{10} = 1024 Mb;

$$1 \text{ Eb} = 2^{10} = 1024 \text{ Gb};$$

Misol sifatida *Ekonometrika* soʻzi nechta belgidan iborat boʻlsa kompyuter xotirasida shuncha baytni, yaʼni 12 baytni yoki $12 \times 8 = 96$ bitni egallaydi.

2 . Nazariy maʼlumot yuzasidan berilgan topshiriqlarni bajarish.

Tushunchalar tahlili:

Chap ustundagi savollarlarning javobiga mosini oʻng ustunga qoʻying:

Telekommunikasiya	-kompyuter tomonidan yaxlit birlik sifatida qabul qilinadigan 8 bitdan iborat axborot.
Texnologiya	- metall yoki plastmassadan ishlangan tekis disk.Uning satxiga lazer nuri yordamida oʻqiladigan axborot yoziladi.
Bit	- axboriy tizim tomonidan axborotni qayta ishlash jarayoniga bogʻlik aniq amal bajarilishini talab qiluvchi son, soʻz yoki jumla.
Piktogramma (ikonka)	- axbortni uzatishga telealoqa vositalarini qoʻllash
Buyruq	– maqola va kitoblarni kompyuterda tashkil qilish va undan foydalanish tizimi.
Bayt	-axborotlarni akslantirish koʻp axboriy vositalar (ovoz, rasm, fotografiya, musiqa va boshqalar)dan foydalanish.
Multimedia	- programma yoki programmalar guruxini ifodalovchi belgi.
Telekommunikatsiya	-obyektning muayyan xususiyatini belgilovchi koʻrsatkich boʻlib, u koʻrilayotgan obyektning aniq bir nusxasi uchun maʼlum sonli, matnli yoki boshqa qiymat qabul qiladi.
Videodisk	- axborotning eng kichik biriligi boʻlib, u ikkilik sanoq sistemasidagi raqam
Avtomatlashtirish	Axborotlarni yigʻish, uzatish, toʻplash, saqlash, taqdim etish va foydalanish uslublari va usullari tizimi
Elektron nashr	kompyuter tarmoqlari va zamonaviy texnik aloqa vositalari negizida maʼlumotlarni masofaga uzatish
Axborot texnologiyasi	– bu inson ish faoliyatini mashina va mexanizmlar bilan almashtirish
Maʼlumot	grekcha soʻz boʻlib mohirlik, ustalik, biror ishni uddalay olish

Nazorat savollari:

1. "Axborot" deb nima tushuniladi?
2. "Ma'lumotlar" deb nimani tushunamiz?
3. Axborotni qanday belgilar bo'yicha tavsiflash mumkin?
4. Eng kichik axborot o'lchov birligi nima?

3-amaliy ish. Algoritmizatsiya asoslari.

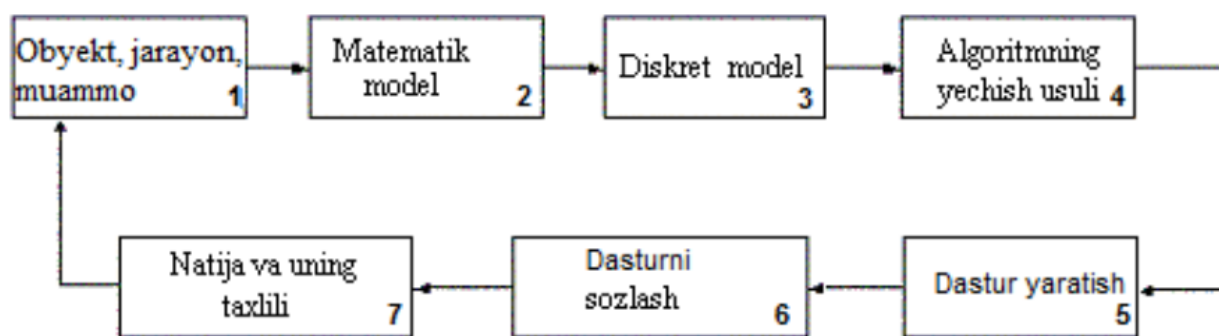
Ishdan maqsad: algoritm ta'rifini bilan tanishish. Algoritmizatsiya asoslari haqidagi tushunchalarni mustahkamlash.

Vazifa.

1. Berilgan nazariy materiallarni o'rganish.
2. Algoritmizatsiya turlari bilan tanishish
3. Berilgan savollarga javob yozish.

1. Berilgan nazariy materiallarni o'rganish.

Kundalik faoliyatimizda biz turli xil muammolar, vazifalarga duch kelamiz. Biz uchun ular talab yoki savol shaklida bo'ladi. Masalan, «Kvadrat tenglamani yechish», «Dengizda nechta tomchi suv bor?», «Ikki karra ikki necha bo'ladi?» kabi. Masalani yechish uchun kerakli amallarni bajarish, ma'lum bir ketma-ketlikda qator harakatlarni amalga oshirish zarur. Aynan shu harakatlar ketma-ketligini to'liq tasavvur etishimiz va tasvirlab berishimiz kerak. Turli muammo, masala yoki jarayonlarni o'rganishni kompyuter yordamida amalga oshirish uchun, birinchi navbatda, tadqiq qilinayotgan masala, jarayon - obyektning matematik ifodasi, ya'ni matematik modelini qurish kerak bo'ladi. Matematik model real obyektning tasavvurimizdagi mavhum ko'rinishi bo'lib, u matematik belgilar va ba'zi-bir qonun-qoidalar bilan ifodalangan bo'ladi. Qurilayotgan obyektning matematik modelini yaratish juda murakkab jarayon bo'lib, o'rganilayotgan obyektga bog'liq ravishda turli soha mutaxassislarining ishtiroki talab etiladi. Umuman, biror masalani kompyuter yordamida yechishni quyidagi bosqichlarga ajratish mumkin (3.1.-rasm)



3.1-rasm. Hisoblash eksperimentining sxemasi

Misol sifatida, kosmik kemani Yerdan Zuhro sayyorasiga eng qulay trayektoriya bo'yicha uchirish masalasini hal qilish talab qilingan bo'lsin. Birinchi navbatda, qo'yilgan masala turli soha mutaxassislarining tomonidan atroflicha o'rganilishi va bu jarayonni ifodalaydigan eng muhim bo'lgan asosiy parametrlarni aniqlash kerak bo'ladi (3.1.- rasm, 1-blok). Masalan, fizik-astronom

muhandis tomonidan masala qo'yilishining o'rinli ekanligi, ya'ni sayyoralar orasidagi masofa va atmosfera qatlamlarining ta'siri, yerning tortish kuchini yengib o'tish va kemandagi og'irligi, zarur bo'lgan yoqilg'ining optimal miqdori va kosmik kemani qurishda qanday materiallardan foydalanish zarurligi, inson sog'lig'iga ta'siri va sarflanadigan vaqt va yana turli tuman ta'sirlarni hisobga olgan holda shu masalaning matematik modelini tuzish zarur bo'ladi. Zikr etilgan ta'sirlarni va fizika qonunlarini hisobga olgan holda bu masalani ifodalaydigan muayyan differensial yoki boshqa ko'rinishdagi modellovchi tenglama hosil qilish mumkin bo'ladi. Balki bu masalani bir nechta alohida masalalarga ajratib o'rganish maqsadga muvofiqdir. Bu matematik modelni o'rganish asosida mazkur masalani ijobiy hal qilish yoki hozirgi zamon sivilizatsiyasi bu masalani yechishga qodir emas degan xulosaga ham kelish mumkin.

Faraz qilaylik, matematik model ishlab chiqildi. Endi uni kompyuter vositasida yechish masalasi paydo bo'ladi. Bizga ma'lumki, kompyuter faqat 0 va 1 diskret qiymatlar va ular ustida arifmetik va mantiqiy amallarni bajara oladi xolos. SHuning uchun matematik modelga mos diskret modelni qurish zaruriyati tug'iladi (3-blok). Odatda, matematik modellarga mos keluvchi diskret modellar ko'p noma'lumli murakkab chiziqsiz algebraik tenglamalar sistemasi (chekli ayirmali tenglamalar-sxemalar) ko'rinishida bo'ladi (4-blok). Endi, hosil bo'lgan diskret modelni sonli yechish usuli – algoritmini yaratish zarur bo'ladi. Algoritm esa tuziladigan dastur uchun asos bo'ladi (5-blok). Odatda, tuzilgan dasturni ishchi holatga keltirish uchun uning xato va kamchiliklarini tuzatish – sozlash zarur bo'ladi (6-blok). Olingan sonli natijalar hali dasturning to'g'ri ishlayotganligiga kafolatini bermaydi. SHuning uchun olingan natijalarni masalaning mohiyatidan kelib chiqqan holda tahlil qilish kerak bo'ladi (7-blok). Agar olingan natija o'rganilayotgan jarayonni ifodalay olmasa, masalani 3.1- rasmdagi sxema asosida qaytadan ko'rib chiqish va zarur bo'lgan joylarda o'zgartirishlar kiritish kerak bo'ladi. Bu jarayon to'g'ri natijani olinguncha davom ettiriladi va bu takrorlanuvchi jarayon hisoblash eksperimenti deb ataladi. Odatda, hisoblash eksperimenti deganda, soddaroq holda, model, algoritm va dastur uchligini (triadasini) tushunish mumkin (3.2-rasm).



3.2 – rasm. Hisoblash eksperimenti uchligi.

Algoritm deb, masalani yechish uchun bajarilishi lozim bo'lgan amallar ketma-ketligini aniq tavsiflaydigan qoidalar tizimiga aytiladi. Boshqacha aytganda, algoritm –boshlang'ich va oraliq ma'lumotlarni masalani echish natijasiga aylantiradigan jarayonni bir qiymatli qilib, aniqlab beradigan qoidalarining biror bir chekli ketma-ketligidir. Buning mohiyati shundan iboratki, agar algoritm ishlab chiqilgan bo'lsa, uni echilayotgan masala bilan tanish bo'lmagan biron bir ijtirochiga, shu jumladan kompyuterga xam bajarish uchun top-shirsa bo'ladi va u algoritmnining qoidalariga aniq rioya qilib masalani echadi.

Hozirgi kunda hisoblash, muhandis-texnik, iqtisodiy, matnli va sonli

axborotlarni taxlil qilish va boshqa masalalarni echish tillari malum. Masalan: FORTRAN tili 1954 yili ishlab chiqilgan bo'lib, FORMula TRANslator - formulalar translyatori degan manoni anglatadi va ilmiy va muhandis - texnik masalalarni hisoblashlarda qo'llaniladi. ALGOL tili 1960 yili yaratilgan bo'lib, ALGORITMIC Language -algoritmik til degan ma'noni anglatadi va ilmiy-texnik masalalarni hisoblashlarda qo'llaniladi. KOBOL tili 1959 yili yaratilgan bo'lib, Common Business Oriented Language - "savdo-sotiq masalalariga mo'ljallangan til" degan ma'noni anglatadi. Korxona va tarmoqning moddiy boyligini, moliyasini, ishlab chiqargan mahsulotini hisobga olish bilan bog'liq iqtisodiy masalalarni echish uchun qo'llaniladi. PASKAL tili 1971 yilda e'lon qilingan bo'lib, frantsuz olimi Blez Paskal nomiga qo'yilgan. Turli xildagi masalalar echimini olishda tartiblangan (strukturaviy) dasturlar tuzishda qo'llaniladi. PL/1 tili 1964 yilda yaratilgan bo'lib, Programming Language/ 1 - 1-tartib raqamli dasturlash tili ma'nosini anglatadi. Ushbu til universal tillar turkumiga kiradi. Bu tilda ishlab chiqilgan dasturlar kompyuterni yangisi bilan almashtirilganda qaytadan tuzib chiqilishi zarur emas. BEYSIK (BASIC - Beginner's All Purpose Symbolic Instruction Code - boshlovchilar uchun ko'p maqsadli dasturlash tili) hisoblash algoritmlarini yozish uchun qo'llaniladigan algoritmik tildir. Bu til 1965 yilda Dartmut kolleji xodimlari Kemini va Kurtslar tomonidan ishlab chiqilgan. Algoritmning asosiy xossalari. Algoritm quyidagi asosiy xossalarga ega: uzluksizlik, aniqlik, natijaviylik va ommaviylik.

UZLUKLILIK. Dastlabki berilgan ma'lumotlarni natijaga aylantirish jarayoni uzluksiz ravishda amalga oshiriladiki, bunda vaqtning xar bir keyingi keladigan daqiqasidagi miqdor (kattalik)larning qiymati vaqtning shundan oldingi daqiqasida bo'lgan miqdorlar qiymatidan ma'lum bir qoidalar bo'yicha olinadi.

ANIQLIK. Algoritmning xar bir qoidasi aniq va bir qiymatli bo'lishi zarurki, bunda vaqtning biror daqiqasida olingan miqdorlar qiymati vaqtning shundan oldingi daqiqasida olingan miqdorlar qiymati bilan bir qiymatli aniqlangan bo'ladi.

NATIJAVIYLIK. Algoritm masalaning yechimiga chekli sondagi qadamlar ichida olib kelishi yoki masalani "yechib bo'lmaydi" degan xabar bilan tugashi kerak.

OMMAVIYLIK. Masalaning echish algoritmi shunday yaratilishi kerakki, uni faqat boshlang'ich ma'lumotlar bilan farqlanadigan masalalarni echish uchun xam qo'llanilishi kerak. Bunda boshlang'ich ma'lumotlar "algoritmni qo'llash sohasi" deb ataladigan birorta sohadan olinadi. Algoritmni ishlab chiqishda uni bir necha xil usul bilan ifodalab bersa bo'ladi. Shulardan uchta keng tarqalgan. Bular: 1. Algoritmni oddiy tilda ifodalash; 2. Algoritmni tuzim ko'rinishida ifodalash; 3. Algoritmni maxsus (algoritmik) tilda yozish.

2 . Algoritmlar turlari bilan tanishish.

ALGORITMNI ODDIY TILDA IFODALASH Algoritmlarni ifodalashning eng keng tarqalgan shakli - oddiy tilda so'zlar bilan bayon qilishdir. Bu nafaqat hisoblash algoritmlarida, balki hayotiy, turmushdagi "algoritm"larga ham tegishlidir. Masalan, biror bir taom yoki qandolat mahsulotini tayyorlashning

retsepti ham oddiy tilda tavsiflangan algoritmdir. Shaharlararo telefon - avtomat orqali aloqa o'rnatishning o'ziga xos algoritmidan foydalanasiz. Do'kondan yangi kir yuvish mashinasi yoki magnitofon sotib olinsa, ishni foydalanishning algoritmi bilan tanishishdan boshlaymiz. Masalani kompyuterda yechishda ham, ko'pincha matematika tilini ham o'z ichiga olgan tabiiy tildan foydalanish mumkin. Algoritmning bunday tildagi yozuvi izlanayotgan natijaga olib keladigan amallar ketmaketligi ko'rinishida bo'lib, odam tomonidan bir ma'noli idrok etilishi kerak. So'zlar bilan ifodalangan har bir amal "algoritmning qadami" deb ataladi. Qadamlar tartib nomeriga ega bo'ladi. Algoritmni tuzim ko'rinishida ifodalash. Nisbatan murakkab masalalarni echishda algoritmdan muayyan kompyuter tilidagi dasturga o'tish juda qiyin. Bunday bevosita o'tishda algoritmning alohida qismlari orasidagi bog'lanish yo'qoladi, algoritm tarkibining asosiy va muhim bo'lmagan qismlarini farqlash qiyin bo'lib qoladi. Bunday sharoitda keyinchalik aniqlash va to'g'rilash ancha vaqt talab qiladigan xatolarga osongina yo'l qo'yish mumkin. Odatda algoritm bir necha marta ishlab chiqiladi, ba'zan xatolarni to'g'rilash, algoritm tarkibini aniqlashtirish va tekshirish uchun bir necha marta orqaga qaytishga to'g'ri keladi. Algoritm ishlab chiqishning birinchi bosqichida algoritmni yozishning eng qulay usuli - algoritmni tuzim ko'rinishida ifodalashdir. Algoritm tuzimi - berilgan algoritmni amalga oshirishdagi amallar ketmaketligining oddiy tildagi tasvirlash elementlari bilan to'ldirilgan grafik tasviridir. Algoritmning har bir qadami tuzimda biror bir geometrik shakl - blok (blok simboli) bilan aks ettiriladi. Bunda bajariladigan amallar turiga ko'ra turlicha bo'lgan bloklarga GOST bo'yicha tasvirlanadigan turli xil geometrik shakllar - to'g'ri to'rtburchak, romb, parallelogramm, ellips, oval va hokazolar mos keladi.

ALGORITMNI MAXSUS TILDA IFODALASH. Bu usulda algoritmni ifodalash uchun "dasturlash tillari" deb ataluvchi sun'iy tillar qo'llaniladi. Buning uchun ishlab chiqilgan algoritm shu tillar yordamida bir manoli va kompyuter tushuna oladigan ko'rinishda tavsiflanishi zarur. Uning tarkibida cheklangan sondagi sintaksis konstruktsiyalar to'plami bor bo'lib, u bilan algoritm yaratuvchi tanish bo'lishi kerak. Ana shu konstruktsiyalardan foydalanib buyruq va ko'rsatmalar formal ifodalarga o'tkaziladi. Zamonaviy dasturlash tillari kompyuterning ichki kompyuter tilidan keskin farq qiladi va kompyuter bevosita ana shu tilda ishlay olmaydi. Buning uchun dasturlash tilidan mashina tushunadigan tilga tarjima qiluvchi maxsus dastur - translyatordan foydalaniladi. Dasturni translyatsiya qilish va bajarish jarayonlari vaqtlarga ajraladi. Avval barcha dastur translyatsiya qilinib, so'ngra bajarish uslubida ishlaydigan translyatorlar "kompilyatorlar" deb ataladi. Dastlabki tilning har bir operatorini o'zgartirish va bajarishni ketma-ket amalga oshiriladigan translyatorlar "interpretatorlar" deb ataladi.

3. Berilgan savollarga javob yozish.

1. "Algoritm" tushunchasining ta'rifini bering.
2. Blok-sxema asosiy figuralar ma'nosini ayting bering.
3. Qanday algoritmlar chiziqli deb ataladi.
4. Qanday algoritmlar tarmoqlanuvchi deb ataladi?

- 5.Qanday algoritmlar takrorlanuvchi deb ataladi?
- 6.Dastur tuzilishida qaysi bloklardan foydalaniladi?
- 7.Algoritmdagi xatolar qanday qilib to'g'rilanadi?
- 8.Yuqori darajadagi dasturlash tillari nima?
9. Diskret model nimadan iborat?
10. Masalani yechish usullari to'g'risida ma'lumot bering.
- 11.Qanday algoritmlar chiziqli deb ataladi.
- 12.Qanday algoritmlar tarmoqlanuvchi deb ataladi?
- 13.Qanday algoritmlar takrorlanuvchi deb ataladi?
- 14.Algoritmdagi xatolar qanday qilib to'g'rilanadi?
- 15.Chiziqli algoritmning ishlash prinsiplari.
- 16.Shartli algoritmning ishlash prinsiplari.
- 17.Takrorlanuvchi algoritmning ishlash usullari.
- 18.Ichma-ich joylashgan takrorlanuvchi algoritmlar nimadan iborat?
- 19.Qaysi masalalarda tarkibiy algoritmdan foydalaniladi?
- 20.Soni noma'lum bo'lgan takrorlash algoritmlar nimadan iborat?

4-amaliy mashg'ulot.Algoritm turlari va xossalari.Ularning xarakteristikallari va xususiyatlari.

Ishdan maqsad: algoritm turlari bilan tanishish. Blok-sxemada ishlatiladigan bloklar bilan tanishish.

Vazifa.

- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Blok-sxemada ishlatiladigan asosiy bloklarni turlari va xossalari bilan tanishish.

1 . Berilgan nazariy materiallarni o'rganish.

Masala yechimini cheklangan qadamlar natijasida hosil qiladigan, oldindan tayinlangan va aniq belgilangan qoidalar yoki buyruqlar ketma-ketligi algoritm deyiladi. Sodaroq qilib aytganda, algoritm bu - oldimizga qo'yilgan masalani yechish uchun zarur bo'lgan amallar ketma-ketligidir. Algoritm tuzish - bu dasturlashdir, algoritmni tuzuvchilar esa dasturchilardir.

Qo'yilgan biror masalani kompyuterda yechish uchun, avval uning matematik modelini, keyin esa yechish algoritmi va dasturini tuzish kerak bo'ladi. Ushbu uchlikda algoritm bloki muhim ahamiyatga ega. Endi algoritm tushunchasining ta'rifi va xossalarini bayon qilamiz. Masala yechimini cheklangan qadamlar natijasida hosil qiladigan, oldindan tayinlangan va aniq belgilangan qoidalar yoki buyruqlar ketma-ketligi **algoritm** deyiladi. Sodaroq qilib aytganda, algoritm bu - oldimizga qo'yilgan masalani yechish uchun zarur bo'lgan amallar ketma-ketligidir. Algoritm tuzish - bu dasturlashdir, algoritmni tuzuvchilar esa dasturchilardir.

Masalan, $ax^2 + bx + c = 0$ kvadrat tenglamani yechish uchun quyidagi amallar ketma-ketligi zarur bo'ladi:

- 1.**a, b, c** koeffitsiyentlar berilgan bo'lsin.
- 2.Berilgan **a, b, c** koeffitsiyentlar yordamida diskriminant.

$D = b^2 - 4ac$ hisoblanadi.

3. $D > 0$ bo'lsa $X_{1,2} = (-b \pm D) / (2 \cdot a)$ - hisoblanadi.

4. $D < 0$ bo'lsa, haqiqiy yechimi yo'q.

Misol sifatida berilgan a, b, c tomonlari bo'yicha uchburchakning yuzasini Geron formulasi bo'yicha hisoblash masalasini ko'rib chiqaylik.

1. a, b, c uchburchak tomonlari uzunliklari.
2. $r = (a+b+c)/2$ - perimetrning yarmi hisoblansin.
3. $T = p(r-a)(r-b)(r-c)$ – hisoblansin.
4. $S = T$ hisoblansin.

Yuqoridagi misollardan ko'rinib turibdiki, algoritmning har bir qadamida bajariladigan amallar tushunarli va aniq tarzda ifodalangan hamda chekli sondagi amallar bajarilgandan keyin aniq natija olish mumkin. "Algoritm" so'zi va tushunchasi IX asrda yashab ijod etgan buyuk alloma Muhammad al-Xorazmiy nomi bilan uzviy bog'liq. Algoritm so'zi Al-Xorazmiy nomini Evropa olimlari tomonidan buzib talaffuz qilinishidan yuzaga kelgan. AlXorazmiy birinchi bo'lib o'nlik canoq sistemasining tamoyillarini va undagi to'rtta amalni bajarish qoidalarini asoslab bergan.

Algoritmning 5 ta asosiy xossasi bor.

1. **Diskretlilik (Cheklilik).** Bu xossaning mazmuni algoritmnlarni doimo chekli qadamlardan iborat qilib bo'laklash imkoniyati mavjudligida. Ya'ni uni chekli sondagi oddiy ko'rsatmalar ketma-ketligi shaklida ifodalash mumkin. Agar kuzatilayotgan jarayonni chekli qadamlardan iborat qilib qo'llay olmasak, uni algoritm deb bo'lmaydi.

2. **Tushunarlilik.** Biz kundalik hayotimizda berilgan algoritmlar bilan ishlayotgan elektron soatlar, mashinalar, dastgohlar, kompyuterlar, turli avtomatik va mexanik qurilmalarni kuzatamiz. Ijrochiga tavsiya etilayotgan ko'rsatmalar uning uchun tushinarli mazmunda bo'lishi shart, aks holda, ijrochi oddiygina amalni ham bajara olmaydi. Bundan tashqari, ijrochi har qanday amalni bajara olmasligi ham mumkin. Har bir ijrochining bajarishi mumkin bo'lgan ko'rsatmalar yoki buyruqlar majmuasi mavjud, u ijrochining ko'rsatmalar tizimi (sistemi) deyiladi. Demak, ijrochi uchun berilayotgan har bir ko'rsatma ijrochining ko'rsatmalar tizimiga mansub bo'lishi lozim. Ko'rsatmalarni ijrochining ko'rsatmalar tizimiga tegishli bo'ladigan qilib ifodalay olishimiz muhim ahamiyatga ega. Masalan, quyi sinfning a'lochi o'quvchisi "son kvadratga oshirilsin" degan ko'rsatmani tushunmasligi natijasida bajara olmaydi, lekin "son o'zini o'ziga ko'paytirilsin" shaklidagi ko'rsatmani bimalol bajaradi, chunki u ko'rsatma mazmunidan ko'paytirish amalini bajarish kerakligini anglaydi.

3. **Aniqlik.** Ijrochiga berilayotgan ko'rsatmalar aniq va mazmunli bo'lishi zarur. Chunki ko'rsatmadagi noaniqliklar mo'ljaldagi maqsadga erishishga olib kelmaydi. Inson uchun tushunarli bo'lgan "3-4 marta silkitilsin", "5-10 daqiqa qizdirilsin", "1-2 qoshiq solinsin", "tenglamalardan biri echilsin" kabi noaniq ko'rsatmalar kompyuterni qiyin ahvolga solib qo'yadi. Bundan tashqari, ko'rsatmalarning qaysi ketma-ketlikda bajarilishi ham muhim ahamiyatga ega. Demak, ko'rsatmalar aniq berilishi va faqat algoritmda ko'rsatilgan tartibda bajarilishi shart ekan.

4. **Ommaviylik.** Har bir algoritm mazmuniga ko'ra bir turdagi masalalarning barchasi uchun ham o'rinli bo'lishi kerak. Ya'ni masaladagi boshlang'ich ma'lumotlar qanday bo'lishidan qat'i nazar algoritm shu xildagi har qanday masalani yechishga yaroqli bo'lishi kerak. Masalan, ikki oddiy kasrning umumiy maxrajini topish algoritmi, kasrlarni turlicha o'zgartirib bersangiz ham ularning umumiy maxrajlari aniqlab beraveradi. Yoki uchburchakning yuzini topish algoritmi, uchburchakning qanday bo'lishidan qat'i nazar, uning yuzasini hisoblab beraveradi.

5. **Natijaviylik.** Har bir algoritm chekli sondagi qadamlardan so'ng, albatta, natija berishi shart. Bajariladigan amallar ko'p bo'lsa ham baribir natijaga olib kelishi kerak. Chekli qadamdan so'ng qo'yilgan masalae chimga ega emasligini aniqlash ham natija hisoblanadi. Agar ko'rilayotgan jarayon cheksiz davom etib natija bermasa, uni algoritm deb atay olmaymiz.

Algoritmni tasvirlash usullari.

Yuqorida ko'rilgan misollarda, odatda, biz masalani yechish algoritmini so'zlar va matematik formulalar orqali ifodaladik. Lekin algoritm boshqa ko'rinishlarda ham berilishi mumkin. Biz endi algoritmning eng ko'p uchraydigan turlari quyidagilar.

1.Algoritmning so'zlar orqali ifodalanishi. Bu usulda ijrochi uchun beriladigan har bir ko'rsatma jumlar, so'zlar orqali buyruq shaklida beriladi.

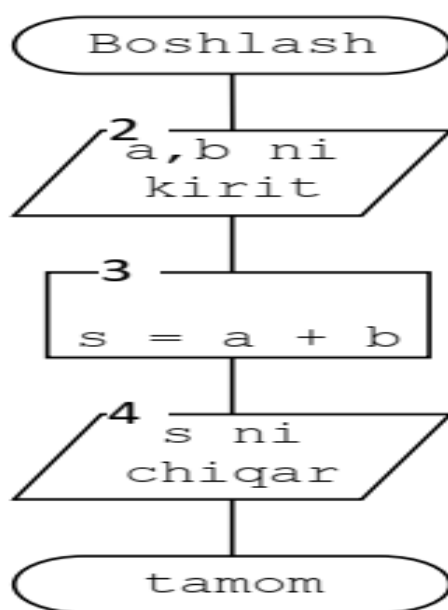
2.Algoritmning formulalar bilan ifodalanish usulidan matematika, fizika, kimyo kabi aniq fanlardagi formulalarni o'rganishda foydalaniladi. Bu usul ba'zan analitik ifodalash deyiladi.

3.Algoritmning maxsus geometrik shakllar yordamida ifodalanishida masala yechish jarayoni aniq va ravon tasvirlanadi va bu ko'rinish blok-sxema deyiladi.

4.Algoritmning jadval ko'rinishda berilishi. Algoritmning bunday ifodasidan ham ko'p foydalanamiz. Masalan, maktabda qo'llanib kelinayotgan to'rt xonali matematik jadvallar yoki turli xil lotereyalar jadvali. Funksiyalarning grafiklarini chizishda ham algoritmning qiymatlari jadvali ko'rinishlaridan foydalanamiz. Bu kabi jadvallardan foydalanish algoritmлари sodda bo'lgani tufayli ularni o'zlashtirib olish oson.

2 . Blok-sxemada ishlatiladigan asosiy bloklarni turlari va xossalari bilan tanishish.

50-yillardan boshlab dasturchilar algoritmni tasvirlash uchun blok-sxema deb nomlanuvchi grafik sxemalardan foydalana boshladi. Masalani yechimini grafik tasvirlash – algoritmni yozish uchun eng qulay usuldir. Algoritmni grafik tasvirlashning keng tarqalgan usullari blok-sxemalar va Nassi-Shnayderman strukturagrammalaridir. Quyida ikki son yig'indisining hisoblash algoritmi blok-sxemasi keltirilgan.

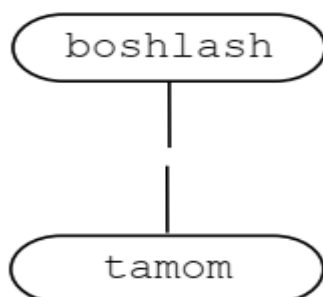


Blok-sxemalar ma'lum qoidalar asosida yaratiladi va amallarning bajarilish tartibini ko'rsatuvchi, o'zaro chiziqlar (strelkalar) bilan bog'langan geometrik shakllar (belgilar bloki) dan iborat. Belgilar bloki standartlashtirilgan va bajarilayotgan amal turiga qarab farqlanadi.

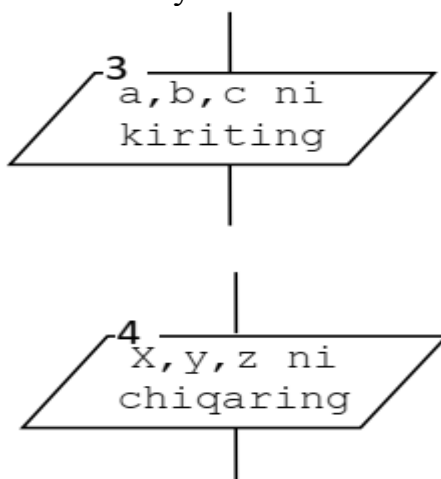
Quyida eng ko'p ishlatilayotgan belgilar bloki keltirilgan:

Figura shakli	Vazifasi
	oval, algoritmnining boshlanishi yoki tugallanishini belgilaydi
	parallelogramm, ma'lumotlarni kiritish yoki chop etishni belgilaydi
	to'g'ri to'rtburchak, amal bajarish jarayonini belgilaydi
	romb, shart bajarilishi tekshirilishini belgilaydi
	yordamchi algoritmgaga murojaatni belgilaydi
	oltiburchak, takrorlash operatorini ifodalashni belgilaydi
	strelka, amallar bajarilish ketma-ketligini aniqlaydi
=> (n)	so'zlar orqali ifodalangan algoritmda n - chi amalga o'tishni ko'rsatadi

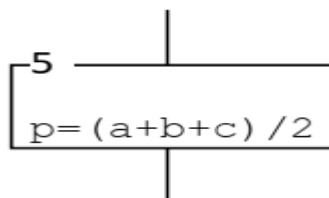
Blok-sxemalar asosiy grafik elementlarining qo'llanilishini batafsil ko'rib chiqamiz. Sxemalarda algoritmning boshlanishi va oxiri hamda yordamchi algoritmlarning kirish va chiqishlari mos ravishda algoritmning boshi va oxirini blokli belgilar bilan ifodalanadi.



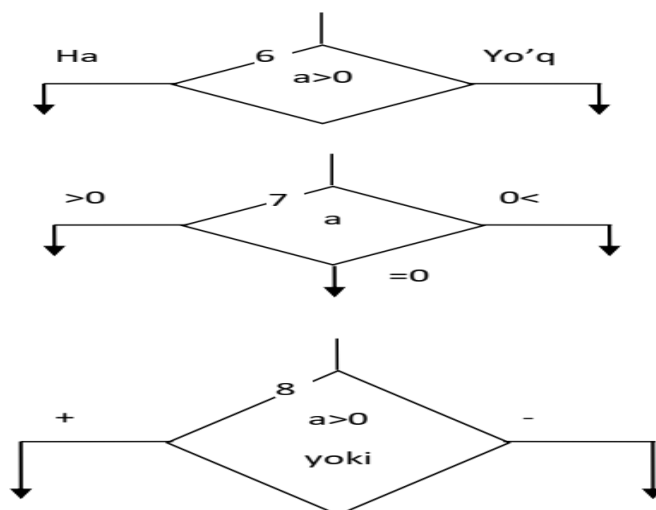
Bu blokli belgilar boshqa belgilardan farqli ravishda bitta kirish va bitta chiqishga ega bo'lib, ma'lumotlarni qayta ishlash yo'lining boshi va oxirini bildiruvchilar sifatida ishlatiladi. Har bir sxema ana shu belgilar bilan boshlanishi va tugashi lozim. Keyingi parallelogramm ko'rinishidagi belgilar ma'lumotlarni kiritish va chiqarish amallari uchun foydaniladi.



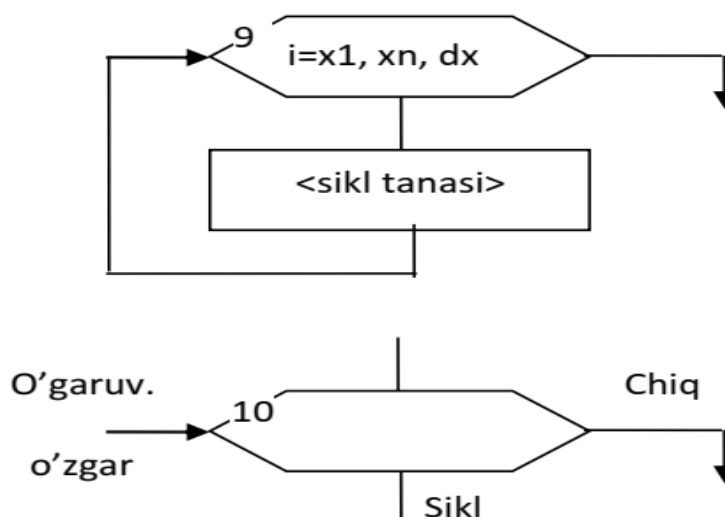
Hisoblash jarayonlarni ifodalovchi blok, amallarni (yoki guruh amallarni), qiymati o'zgaruvchanlarni, tasvirlash shakllarini yoki ma'lumotlar joylashini ifodalashda qo'llaniladi.



Bu blokda tavsiflanuvchi amallar yechimini bildiruvchi matematik formula, ifoda va ixtiyoriy tildagi tushuntirishlarda yozilishi mumkin. Yechimni bildiruvchi, mantiqiy blok belgisi, biror-bir shart asosida algoritm bajarilishi yo'nalishini tanlashda ishlatiladi.



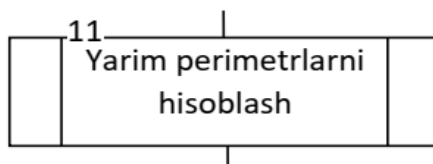
Blokda shart, savol yoki algoritm keyingi yo'nalishni aniqlaydigan yechim ko'rsatiladi. Shart oddiy (6 blok) yoki tarkibli (8 blok) bo'lishi mumkin. Masalalarni yechish jarayonida shart bloklarida yuz berishi mumkin bo'lgan barcha holatlar hisobga olinishi kerak. Bitta chiqishga ega bo'lgan blokli belgilardan farqli ravishda shartli bloklardan ikki yoki uchta informatsiyali oqim yo'nalishlari chiqishi mumkin. Blokdan chiqish chiziqlari algoritm yo'nalishini amalga oshiruvchi shartlar (masalan, "ha" yoki "yo'q", <0 , $=0$ yoki >0 , $+$ yoki $-$ va h.k.) bilan ta'minlangan bo'lishi kerak. Modifikatsiya blok belgilari siklik hisoblashlarning boshlanishini (sikl sarlavhasi) bildirib, siklni boshqarish uchun ishlatiladi.



Blok ichida sikl o'zgaruvchilari va uning o'zgarishni aniqlash qonuniyati xususiyatlari ko'rsatiladi, masalan, $i=x_1, x_n, dx$, bu yerda i – o'zgaruvchisi, x_1 va x_n – sikl parametrining boshlang'ich va oxirgi qiymatlari, dx – o'zgarish qadami (sikl parametri x_1 dan x_n gacha dx qadam bilan o'zgaradi). Agar qadam 1 ga teng bo'lsa, unda dx ni ko'rsatmaslik ham mumkin.

Modifikatsiya blokida kirish chizig'idan tashqari bitta chiqish (rasmda "Chiqish" orqali ko'rsatilgan) ham mavjud bo'lib, bu chiziq sikl o'zgarishini hisoblash jarayonining boshiga qayta ishlash uchun uzatadi ("Sikl") va sikl parametrini o'zgarish uchun boshiga qaytarish ("O'zgaruv. o'zgar.").

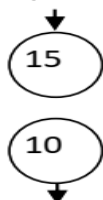
Qismaniy dastur hisoblashlariga murojaat qilish uchun (standart yoki foydalanuvchi tomonidan yaratilgan) sxemada oldindan yaratilgan jarayon blok belgisi ishlatiladi.



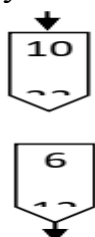
Bu blok qismaniy dastur (yordamchi algoritm) algoritmi o'rnida ishlatilib, ma'lumotlar oqimining qismaniy dasturga uzatilishini bildiradi. Qismaniy dasturdagi hisoblash jarayoni tugallanganidan so'ng, hisoblash natijalari asosiy algoritmgaga qaytariladi, hisoblash jarayoni qismaniy dasturga murojaat blokidan keyingi jarayonidan boshlab tiklanadi.

Oldindan tayyorlangan jarayon bloki alohida modul sifatida hosil qilingan yordamchi algoritmlarni tashkillashtirishda yoki kutubxona qismaniy dasturlariga murojaatda ishlatiladi.

Ma'lumotlar oqimi yo'llari uzunligini va kesishishlar sonini kamaytirish uchun ularni uzish va uzilgan joylarga ulagichlarni o'rnatish mumkin. Agar bloklar o'rtasidagi chiziqlar uzilishi bitta sahifada yuz bersa, unda ulagich sifatida maxsus belgilardan foydalaniladi.



Blok belgilari ichiga hisoblash jarayoni uzatiladigan yoki qayerdan kelganligini bildiruvchi blok raqamlari yoziladi. Xudi shunday yuqori ulagich hisoblash jarayoni 15 – blok kirishga uzatilayotganligini, quyi ulagich esa hisoblash jarayoni 10 – blokdan chiqayotganligini bildiradi. Agar ulagich chiziqlari turli xil sahifalarda joylashgan bloklarni ulash jarayonini ko'rsatadigan bo'lsa, unda sahifalararo ulagich belgilaridan foydalaniladi, hamda bu blok ulagichlari ichiga nafaqat blok raqamlari, balki sahifa raqamlari ham yoziladi.



Yuqoridagi sahifalararo ulagich tasvirlari hisoblash jarayoni 10-

	algoritmini tuzing : A, B, C sonlarning har biri musbat".
4	Uchta A, B, C butun sonlar berilgan. Jumlani rostlikka tekshirish algoritmini tuzing:A, B, C sonlarning hech bo'lmaganda bittasi musbat".
5	Uchta A, B, C butun sonlar berilgan. Jumlani Jumlani rostlikka tekshirish algoritmini tuzing:A, B, C sonlaridan faqat bittasi musbat son".
6	Musbat butun son berilgan. Jumlani rostlikka tekshirish algoritmini tuzing: "Berilgan son ikki xonali juft son".
7	Musbat butun son berilgan. Jumlani rostlikka tekshirish algoritmini tuzing: "Berilgan son uch xonali toq".
8	Jumlani rostlikka tekshirish algoritmini tuzing "Berilgan uchta butun sonlarning hech bo'lmaganda 2 tasi bir biriga teng".
9	Jumlani rostlikka tekshirish algoritmini tuzing: "Berilgan uchta butun sonlarning hech bo'lmaganda bir jufti o'zaro qarama qarshi".
10	Uch xonali son berilgan. Jumlani rostlikka tekshirish algoritmini tuzing: "Ushbu sonning barcha raqamlari xar xil".
11	n butun soni berilgan ($n > 0$). Quyidagi yig'indini hisoblovchi algoritm tuzilsin. $S = 1 + 1/2 + 1/3 + \dots + 1/n$
12	n butun soni berilgan ($n > 0$). Quyidagi yig'indini hisoblovchi algoritm tuzilsin. $s = n^2 + (n+1)^2 + (n+2)^2 + \dots + (2n)^2$
13	n butun soni berilgan ($n > 0$). Quyidagi yig'indini hisoblovchi algoritm tuzilsin. $S = 1 - 2 + 3 - \dots$ (n ta qo'shiluvchi, ishoralar almashib keladi.)
14	n butun soni berilgan ($n > 0$). Shu sonning kvadratini quyidagi formula asosida hisoblovchi algoritm tuzilsin. $n^2 = 1 + 3 + 5 + \dots + (2n - 1)$ har bir qo'shiluvchidan keyin natijani ekranga chiqarib boring. Natijda ekranda 1 dan n gacha bo'lgan sonlar kvadrati chiqariladi.
15	n butun soni va a haqiqiy soni berilgan ($n > 0$). a ning n - darajasini aniqlovchi algoritm tuzilsin. $a^n = a * a * \dots * a$;
16	n butun soni va a haqiqiy soni berilgan ($n > 0$). Bir sikldan foydalanib a ning 1 dan n gacha bo'lgan barcha darajalarini chiqaruvchi algoritm tuzilsin.
17	n butun soni va a haqiqiy soni berilgan ($n > 0$). Bir sikldan foydalanib quyidagi a ning 1 dan n gacha bo'lgan barcha darajalarini chiqaruvchi va yig'indini hisoblovchi algoritm tuzilsin. $1 + a + a^2 + a^3 + \dots + a^n$
18	n butun soni va a haqiqiy soni berilgan ($n > 0$). Bir sikldan foydalanib quyidagi a ning 1 dan n gacha bo'lgan barcha darajalarini chiqaruvchi va yig'indini hisoblovchi algoritm tuzilsin. $1 - a + a^2 - a^3 + \dots + (-1)^n a^n$
19	n butun soni berilgan ($n > 0$). Quyidagi ko'paytmani hisoblovchi algoritm tuzilsin. $S = 1.1 * 1.2 * 1.3 * \dots$ (n ta ko'payuvchi)
20	n va k butun son berilgan. k sonini n marta chiqarish algoritmini tuzing.

5-amaliy mashg'ulot. Chiziqli hisoblash jarayonlarini algoritmlash.

Ishdan maqsad: Berilgan masala uchun uni ishlanish algoritmini, uning blok-sxemasini tuzish usullarini o'rganish.

Vazifa.

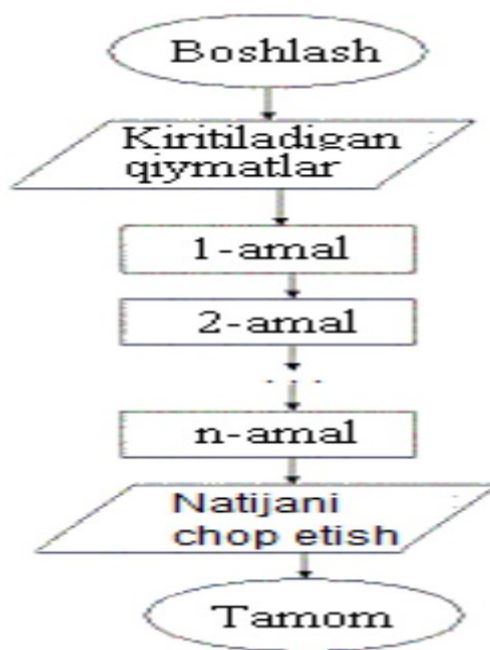
- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Chiziqli algoritmlar bilan tanishish.

1 . Berilgan nazariy materiallarni o'rganish.

Masala yechimini cheklangan qadamlar natijasida hosil qiladigan, oldindan tayinlangan va aniq belgilangan qoidalar yoki buyruqlar ketma-ketligi algoritm deyiladi. Soddaroq qilib aytganda, algoritm bu - oldimizga qo'yilgan masalani yechish uchun zarur bo'lgan amallar ketma-ketligidir. Algoritm tuzish - bu dasturlashdir, algoritmni tuzuvchilar esa dasturchilardir.

Har qanday murakkab algoritmni ham uch asosiy struktura yordamida tasvirlash mumkin. Bular ketma-ketlik, ayri va takrorlash strukturalaridir. Ushbu strukturalar asosida chiziqli, tarmoqlanuvchi va takrorlanuvchi hisoblash jarayonlarining algoritmlarini tuzish mumkin. Umuman olganda, algoritmlarni shartli ravishda quyidagi turlarga ajratish mumkin:

- *chiziqli* algoritmlar;
- *tarmoqlanuvchi* algoritmlar;
- *takrorlanuvchi* algoritmlar;
- *ichma-ich joylashgan takrorlanuvchi* algoritmlar;
- *rekurrent* algoritmlar;
- *takrorlanishlar soni oldindan no'malum* algoritmlar;
- *ketma-ket yaqinlashuvchi* algoritmlar.



5.1-rasm. Chiziqli algoritmlar blok-sxemasining umumiy tuzilishi

2 . Chiziqli algoritmlar bilan tanishish.

Faqat ketma-ket bajariladigan amallardan tashkil topgan algoritmlarga - chiziqli algoritmlar deyiladi. Bunday algoritmnı ifodalash uchun ketma-ketlik strukturasi ishlatiladi. Strukturada bajariladigan amal mos keluvchi shakl bilan ko'rsatiladi. Chiziqli algoritmlar blok-sxemasining umumiy tuzilishi 5.2.-rasmıda keltirilgan.

1-misol. $ax^2+bx+c=0$ kvadrat tenglamani yechish uchun quyidagi amallar ketma-ketligi zarur bo'ladi:

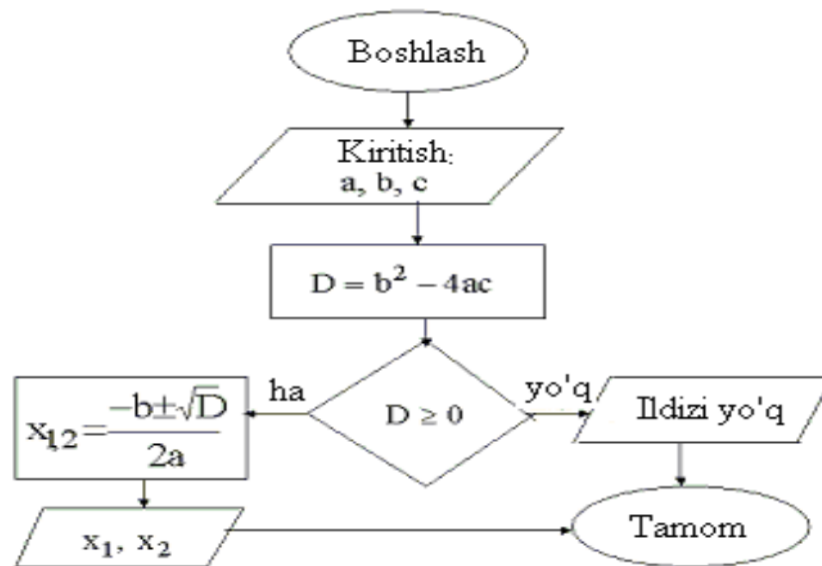
1. a, b, c koeffitsiyentlar berilgan bo'lsin.
2. Berilgan a, b, c koeffitsiyentlar yordamida diskreminant. $D=b^2-4ac$ hisoblanadi.
3. $D>0$ bo'lsa $X_{1,2} = \frac{-b \pm \sqrt{D}}{2a}$ hisoblanadi.
4. $D<0$ bo'lsa, haqiqiy yechimi yo'q.

Berilgan masala uchun algoritm.

Ushbu kvadrat tenglama quyidagicha yechiladi:

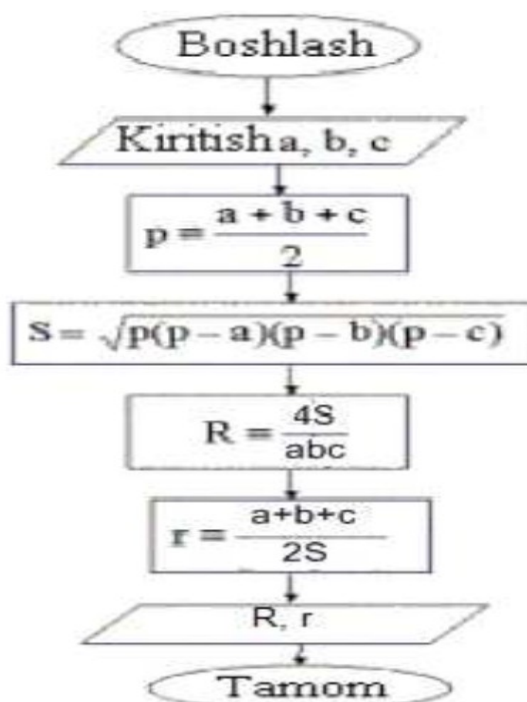
1. a, b, c – koeffitsientlar kiritiladi;
2. D – diskreminant, formula yordamida hisoblanadi;
3. $D>0$ bo'lsa, yechimlar aniqlanadi, aks holda $D=0$ bo'lsa, yechimi aniqlanadi, aks holda $D<0$ bo'lsa, berilgan tenglamaning yechimi yo'qligi aniqlanadi.

Berilgan masala uchun blok-sxema



5.2-rasm. Kvadrat tenglamani yechish blok-sxemasi

2-misol. Uchburchak tomonlarining uzunligi bilan berilgan. Uchburchakka ichki r va tashqi R chizilgan aylanalar radiuslarini hisoblang. Ichki chizilgan aylana radiusi $r = (a+b+c)/2S$, tashqi chizilgan aylana radiusi $R = 4S/abc$ formulalar orqali hisoblanadi. Bu yerda S - uchburchakning yuzi, a, b, c – uchburchak tomonlarining uzunliklari. Masala yechimining blok-sxemasi 5.3-rasmıda keltirilgan



5.3-rasm. Uchburchakka ichki va tashqi chizilgan aylanalar radiuslarini hisoblash blok-sxemasi

3-misol. Quyida keltirilgan munosabatni hisoblash algoritmini ko'rib chiqaylik. Jarayon amallarni ketma-ket bajarilishidan iborat.

$$z = x^2 + \sqrt{|\sin(x + y)|},$$

bu yerda

$$x = \cos(a - b), y = \ln(a_2 - x_2), a = 0.7, b = 2.1.$$

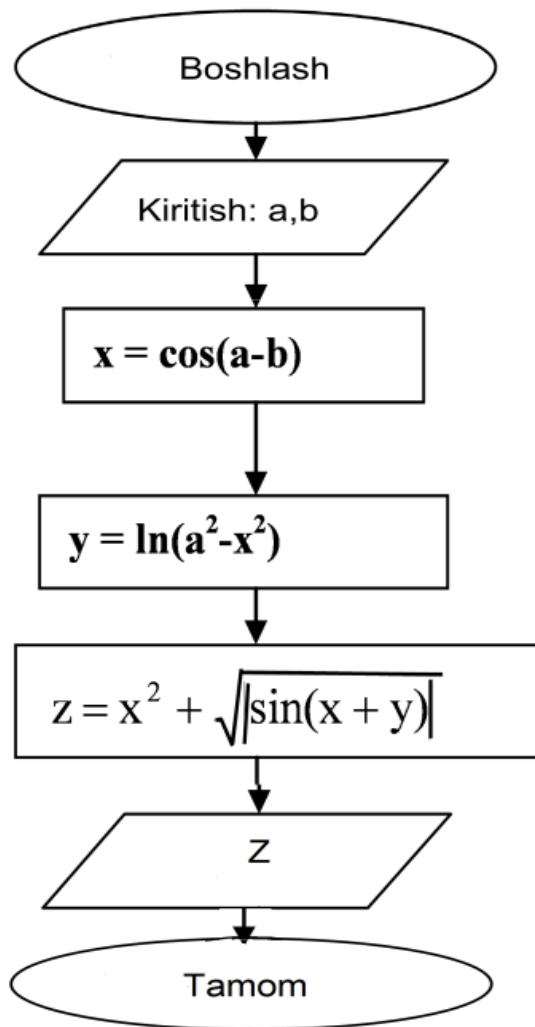
Bunda:

a, b - aniq qiymatga ega bo'lgan boshlang'ich ma'lumotlar;

x, y - oraliq ma'lumotlar;

z – natija

Masalani yechish jarayoni chiziqli hisoblanadi, chunki boshlang'ich ma'lumotlar kiritilgach, munosabatlarning qiymati dasturda joylashgan tartibda hisoblanadi, ya'ni dastlab x, so'ng - y qiymati va nihoyat z natija hisoblanadi. Mazkur jarayonning blok-sxemasi 5.3-rasmda keltirilgan.



5.4-rasm. Hisoblash blok sxemasi

Mustaqil bajarish uchun topshiriqlar.

1. Bir to'g'ri chiziqda yotmaydigan uchta nuqta koordinatlari orqali o'tuvchi aylanani yuzasini hisoblash algoritmini tuzing.
2. "Svetofordan (uchchiroqli) foydalanish" algoritmini tuzing.
3. Geron formulasi bo'yicha uchburchak yuzasini hisoblash algoritmining blok-sxemasini tuzing.
- 4.2. $y = x^2 + 2x^4 + b$ funksiyani $x=3$ qiymatida hisoblash algoritmining blok-sxemasini tuzing.
5. Quyidagi ifodani hisoblash algoritmini tuzing:
 $y = (\sin(a^2) + \cos^2(a - b \ln(/)) a + b) + c^2$ bu yerda
 $a = (p + q *) b - c$, $b = \sin(e^2 - (p + q^3))$, $c = \text{tg}(3p + q) - b$.
6. Quyidagi ifodani hisoblash algoritmini tuzing:
 $s = (\text{tg}(y^2) + (x - y^3 \ln(/)) x + y^4)$, bu yerda
 $x = (a + b)^2 - (y - c^3)$, $y = e^{(a+b)} - (c^2 - (a - b^3))$.
7. Quyidagi ifodalarni hisoblash algoritmini tuzing:
 $z = (a - b)^3 + \lg^2(2a + 3c) / (\text{tg}^2(c - \text{sina}))$ bu yerda
 $a = \sin((b - c^2))^3$, $b = \text{tg}(x + y^2)$, $c = \cos(e^{(x-y)} + 5\sin(b + 2))$.

8. Quyidagi ifodalarni hisoblash algoritmini tuzing:

$f = \lg(x^2 - y + z^3) + \cos^4(x + y)$ bu yerda

$$x = (\cos(a^3 - b^2) + \sin(e^{a+b})) / (a + \cos(y + 2)), \quad y = |a^2 - b|, \quad z = (a + |y - x|)^2$$

9. Haqiqiy a qiymat berilgan. Uchta ko'paytirish amali yordamida a^8 qiymatini hisoblash algoritmini tuzing.

10. Berilgan x uchun quyidagi ifodaning qiymatini hisoblash algoritmini tuzing:

$$A = |x - 2| / (2 \cdot x + \sqrt{x} - 2).$$

11. Berilgan ikki (x_1, y_1) va (x_2, y_2) nuqtalar orasidagi masofani hisoblash algoritmini tuzing.

12. Geron formulasi asosida uchlari koordinatalari bilan berilgan uchburchak yuzasini hisoblash algoritmini tuzing.

13. Diskreminant hisoblash asosida uchlari koordinatalari bilan berilgan uchburchak yuzasini hisoblash algoritmini tuzing.

14. Teng bo'laklarga bo'lish usuli yordamida $[a, b]$ oralig'ida $f(x) = 0$ tenglamaning yagona ildizini berilgan aniqlik bilan hisoblash algoritmini tuzing.

15. Berilgan 4 ta - a, b, c, d haqiqiy sonlar orasidan eng katta va eng kichik qiymatlar o'rtasidagi ayirmani aniqlang.

6-amaliy mashg'ulot. Tarmoqlanuvchi hisoblash jarayonini tashkil qilish.

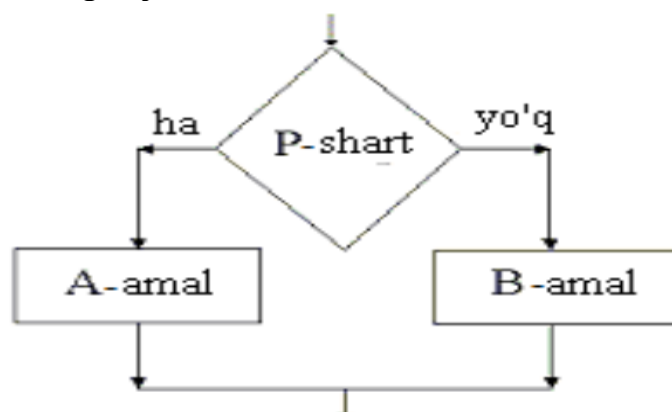
Ishdan maqsad: berilgan masala uchun uni ishlanish algoritmini va blok-sxemasini tuzish usullarini o'rganish.

Vazifa.

1. Berilgan nazariy materiallarni o'rganish.
2. Tarmoqlanuvchi algoritmlar bilan tanishish.

1. Berilgan nazariy materiallarni o'rganish.

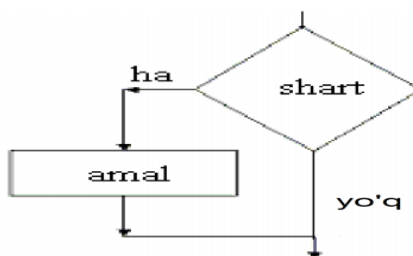
Agar hisoblash jarayoni biror-bir berilgan shartning bajarilishiga qarab turli tarmoqlar bo'yicha davom ettirilsa va hisoblash jarayonida har bir tarmoq faqat bir marta bajarilsa, bunday hisoblash jarayonlari tarmoqlanuvchi algoritmlar deyiladi. Tarmoqlanuvchi algoritmlarni tasvirlash uchun "ayri" tuzilmasi ishlatiladi. Tarmoqlanuvchi tuzilmasi berilgan shartning bajarilishiga qarab ko'rsatilgan tarmoqdan faqat bittasining bajarilishi ta'minlanadi.



6.1-rasm.Tarmoqlanishning umumiy ko‘rinishi

Berilgan R-*shart* romb figurasi ichida tasvirlanadi. Agar *shart* bajarilsa, "ha" tarmoq bo'yicha A-amal, aks holda (*shart* bajarilmasa) "yo'q" tarmoq bo'yicha B-amal bajariladi.

Ko'pgina masalalarni yechishda, *shart* asosida tarmoqlanuvchi algoritmnining ikki tarmog'idan biri, ya'ni «rost» yoki «yolg'on»ning bajarilishi yetarli bo'ladi. Bu holat tarmoqlanuvchi algoritmnining xususiy holi sifatida qisqartirilgan struktura deb atash mumkin. Qisqartirilgan struktura blok-sxemasi quyidagi ko'rinishga ega



6.2-rasm. Qisqartirilgan strukturaning umumiy ko‘rinishi

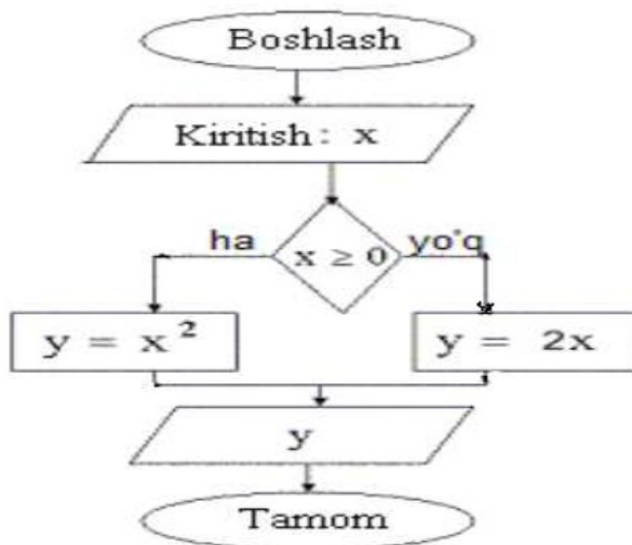
2 . Tarmoqlanuvchi algoritmlar bilan tanishish.

1-*misol*. Tarmoqlanuvchi algoritmgaga *misol* sifatida quyidagi sodda masalani keltiramiz:

$$y = \begin{cases} x^2, & \text{agar } x \geq 0 \\ 2x, & \text{aks holda.} \end{cases}$$

Natijaviy qiymat y berilgan x ning qiymatiga bog'liq holda bo'ladi: agar $x \geq 0$ shart *rost* bo'lsa, tarmoq bo'yicha $y = x^2$ munosabatning qiymati, aks holda, $y = 2 \cdot x$ munosabatning qiymati hisoblanadi. Bu masala bajarilishining so'z bilan ifodalangan algoritmi quyidagicha:

agar ($x \geq 0$) shart bajarilsa, u holda $u = x^2$, aks holda $u = 2 \cdot x$.
Masala yechimining blok-sxemasi 6.2-rasmida keltirilgan.



6.3-rasm. Interval ko‘rinishidagi funksiya qiymatini hisoblash blok-sxemasi

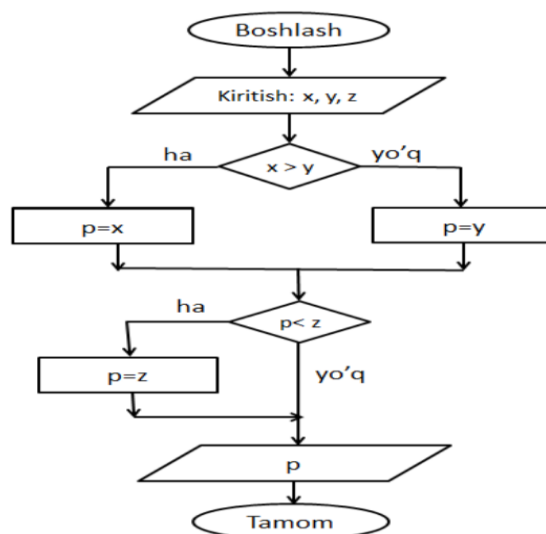
2-*misol*. Berilgan x , u , z sonlari ichidan eng kattasini aniqlang. Ushbu masalaga mos matematik modelni quyidagicha tasvirlash mumkin:

$$p = \max\{x, y, z\}.$$

Berilgan x, y, z sonlardan eng kattasi p deb belgilangan. Soʻzlar orqali ifodalangan algoritm asosida masala yechimini quydagicha tasvirlash mumkin:

- 1) kiritish (x, y, z);
- 2) agar ($x > y$) boʻlsa, u holda $p = x$, aks holda $p = y$;
- 3) agar ($p < z$) boʻlsa, u holda $p = z$;
- 4) muhrlash (p).

Keltirilgan algoritmga mos blok-sxema 6.4-rasmda tasvirlangan. Bu algoritmga, avval x va y oʻzaro solishtiriladi, katta qiymatligi ega r ga yuklanadi. Soʻngra x va y larning kattasi deb aniqlangan r va z oʻzaro solishtiriladi. Agar $r < z$ sharti bajarilsa, u holda eng katta qiymat $p = z$ deb olinadi, aks holda boshqarish navbatdagi amalga uzatiladi. Natijada p da uchta qiymatdan eng kattasi aniqlanadi.



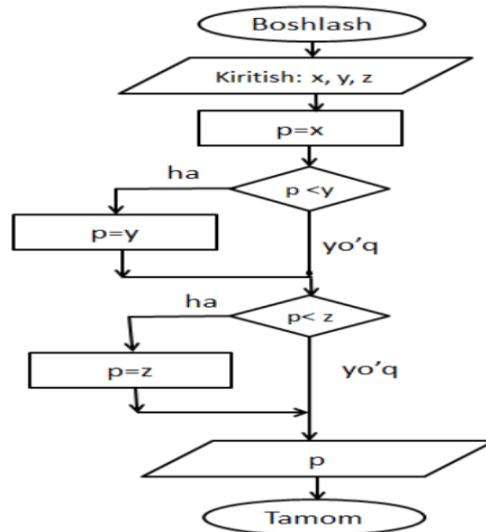
6.4-rasm. Berilgan x, y, z sonlar ichidan eng kattasini topish blok-sxemasi

Ushbu masalani yechish algoritmining yana bir usulini koʻrib chiqamiz.

- 1) kiritish (x, y, z);
- 2) $p = x$;
- 3) agar ($p < y$) boʻlsa, u holda $p = y$;
- 4) agar ($p < z$) boʻlsa, u holda $p = z$;
- 5) muhrlash (p).

Bu algoritmga mos blok-sxema 6.5-rasmda tasvirlangan.

Bu usulga asosan, avvalo sonlar ichida birinchisi eng kattasi deb faraz qilinadi, yaʼni $p = x$. Soʻngra har bir qadamda navbatdagi son – r ning qiymati bilan solishtiriladi va shart bajarilsa, u eng kattasi deb qabul qilinadi. Bu algoritmning afzalligi shundaki, uning asosida uchta va undan koʻp sonlar ichidan eng kattasini (kichigini) topishning qulay imkoniyati mavjud.



6.5-rasm. Hisoblash blok-sxemasi

Yuqorida keltirilgan soʻzlar asosida ifodalangan algoritm va blok-sxemadan koʻrinib turibdiki, amallar ketma-ketligining maʼlum qismi parametr i ga nisbatan n marta takrorlanadi.

3-misol. Quyidagi ifoda bilan berilgan munosabatni hisoblang [2, 52 b.]

$$Y = \begin{cases} b - x, & \text{agar } x > 0, \\ x + a, & \text{agar } x < 0, \\ a + b, & \text{aks holda .} \end{cases}$$

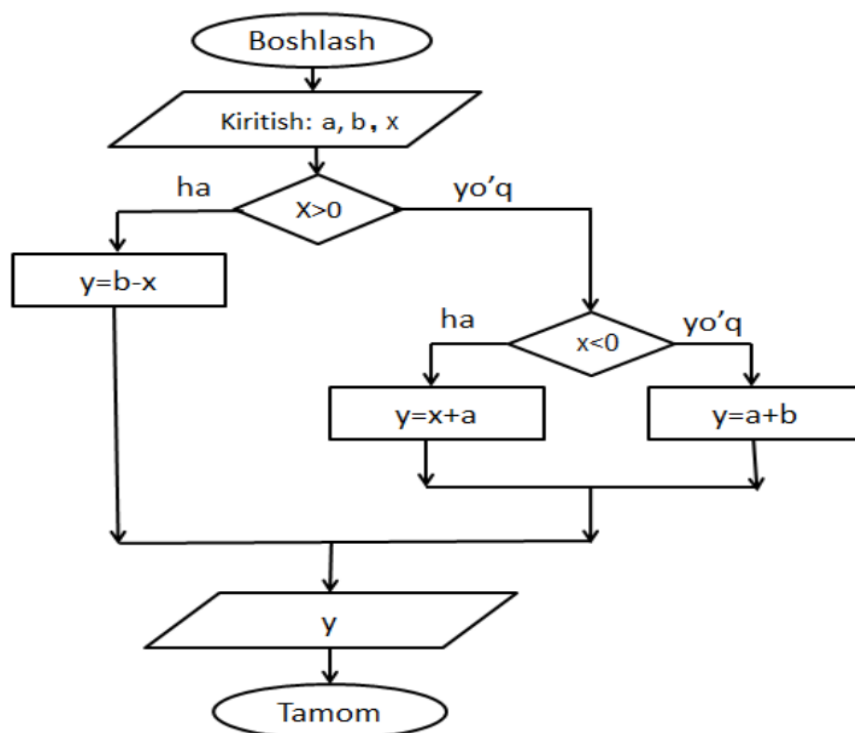
Bu misol natija x ning qiymatiga bogʻliq shart bilan berilgan va masala quyidagicha soʻzlar orqali ifodalangan algoritm asosida aniqlanadi:

agar $x > 0$ boʻlsa, u holda $u = b - x$ boʻladi, aks holda;
agar $x < 0$ boʻlsa, u holda $u = x + a$, aks holda $u = a + b$.

Avvalo, birinchi shart tekshiriladi va agar u bajarilsa, $y = b - x$ amal bajariladi, aks holda

$$Y = \begin{cases} x + a, & \text{agar } x < 0, \\ a + b, & \text{aks holda .} \end{cases}$$

munosabat hisoblanadi. Bu fikrlar quyidagi blok-sxemada oʻz aksini topgan (1.12-rasm).



6.6-rasm. Hisoblash blok-sxemasi

Topshiriqlar: berilgan masalalarning hisoblash blok-sxemasini tuzing.

1. Bir to'g'ri chiziqda yotmaydigan uchta nuqta koordinatlari orqali o'tuvchi aylanani yuzasini hisoblash algoritmini tuzing.
2. "Svetofordan (uchchiroqli) foydalanish" algoritmini tuzing.
3. Geron formulasi bo'yicha uchburchak yuzasini hisoblash algoritmining blok-sxemasini tuzing.
4. $y = x^2 + 2x^4 + b$ funksiyani $x = 3$ qiymatida hisoblash algoritmining blok-sxemasini tuzing.
5. Quyidagi ifodani hisoblash algoritmini tuzing:
 $y = (\sin(a^2) + \cos^2(a - b \ln(/)) a + b) + c^2$ bu yerda
 $a = (p + q^*) b - c$, $b = \sin(e^2 - (p + q^3))$, $c = \lg(3p + q) - b$.
6. Quyidagi ifodani hisoblash algoritmini tuzing:
 $s = (\lg(y^2) + (x - y^3 \ln(/)) x + y^4)$, bu yerda
 $x = (a + b)^2 - (y - c^3)$, $y = e^{(a+b)} - (c^2 - (a - b^3))$.
7. Quyidagi ifodalarni hisoblash algoritmini tuzing:
 $z = (a - b)^3 + \lg^2(2a + 3c) / (\lg^2(c - \sin a))$ bu yerda
 $a = \sin((b - c^2))^3$, $b = \lg(x + y^2)$, $c = \cos(e^{(x-y)} + 5\sin(b + 2))$.
8. Quyidagi ifodalarni hisoblash algoritmini tuzing:
 $f = \lg(x^2 - y + z^3) + \cos^4(x + y)$ bu yerda
 $x = (\cos(a^3 - b^2) + \sin(e^{a+b})) / (a + \cos(y + 2))$, $y = |a^2 - b|$, $z = (a + |y - x|)^2$
9. Haqiqiy a qiymat berilgan. Uchta ko'paytirish amali yordamida a^8 qiymatini hisoblash algoritmini tuzing.
10. Berilgan x uchun quyidagi ifodaning qiymatini hisoblash algoritmini tuzing:

$$A = |x - 2| / (2 \cdot x + \sqrt{x} - 2).$$

11. Berilgan ikki (x_1, y_1) va (x_2, y_2) nuqtalar orasidagi masofani hisoblash algoritmini tuzing.
12. Geron formulasi asosida uchlari koordinatalari bilan berilgan uchburchak yuzasini hisoblash algoritmini tuzing.
13. Diskriminant hisoblash asosida uchlari koordinatalari bilan berilgan uchburchak yuzasini hisoblash algoritmini tuzing.
14. Teng bo'laklarga bo'lish usuli yordamida $[a, b]$ oralig'ida $f(x) = 0$ tenglamaning yagona ildizini berilgan aniqlik bilan hisoblash algoritmini tuzing.
15. Berilgan 4 ta a, b, c, d haqiqiy sonlar orasidan eng katta va eng kichik qiymatlar o'rtasidagi ayirmani aniqlang.
16. 100 ballik shkala bo'yicha talabalarning to'plagan ballari klaviatura orqali kiritiladi. Ballar taqsimotining qabul qilingan usuliga ko'ra talabalarning ballar bo'yicha baholarini aniqlash dasturini tuzing.
17. Xizmatchining oylik maoshidan o'sib boruvchi shkala bo'yicha ushlab qolinadigan daromad solig'ini hisoblang.
18. Fibonachchi sonlar qatori: 1, 1, 2, 3, 5, ... dan dastlabki n - ta sonlarning yig'indisini aniqlang.
19. Aylananing yuzasi S va kvadratning yuzasi R berilgan. Kvadratning aylanaga sig'ishni yoki sig'masligi aniqlash algoritmini tuzing.
20. Berilgan uchta a, b, s sonlardan foydalanib tomonlarining uzunliklari shu sonlarga teng bo'lgan uchburchakning mavjudligini aniqlang va shunday uchburchakni yasash mumkin bo'lsa, uning yuzasini hisoblash algoritmini tuzing.

7-amaliy mashg'ulot. Takrorlanuvchi hisoblash jarayonini tashkil qilish.

Ishdan maqsad: berilgan masala uchun uni ishlash algoritmini va blok-sxemasini tuzish usullarini o'rganish va berilgan misolni ishlash.

Vazifa.

1. Berilgan nazariy materiallarni o'rganish.
2. Takrorlanuvchi algoritmlar bilan tanishish.

1. Berilgan nazariy materiallarni o'rganish.

Masala yechimini cheklangan qadamlar natijasida hosil qiladigan, oldindan tayinlangan va aniq belgilangan qoidalar yoki buyruqlar ketma-ketligi algoritm deyiladi. Soddaroq qilib aytganda, algoritm bu - oldimizga qo'yilgan masalani yechish uchun zarur bo'lgan amallar ketma-ketligidir. Algoritm tuzish - bu dasturlashdir, algoritmnı tuzuvchilar esa dasturchilardir.

Qo'yilgan biror masalani kompyuterda yechish uchun, avval uning matematik modelini, keyin esa yechish algoritmi va dasturini tuzish kerak bo'ladi. Ushbu uchlikda algoritm bloki muhim ahamiyatga ega. Endi algoritm tushunchasining ta'rifi va xossalarini bayon qilamiz. Masala yechimini cheklangan qadamlar natijasida hosil qiladigan, oldindan tayinlangan va aniq

belgilangan qoidalar yoki buyruqlar ketma-ketligi **algoritm** deyiladi. Soddaroq qilib aytganda, algoritm bu - oldimizga qo'yilgan masalani yechish uchun zarur bo'lgan amallar ketma-ketligidir. Algoritm tuzish - bu dasturlashdir, algoritmni tuzuvchilar esa dasturchilardir.

Agar biror masalani yechish uchun zarur bo'lgan amallar ketma-ketligining ma'lum bir qismi biror parametrga bog'liq holda ko'p marta qayta bajarilsa, bunday jarayon takrorlanuvchi algoritm deyiladi. Takrorlanuvchi algoritmlarga misol sifatida odatda qatorlarning yig'indisi yoki ko'paytmasini hisoblash jarayonlarini qarash mumkin.

2 . Takrorlanuvchi algoritmlar bilan tanishish.

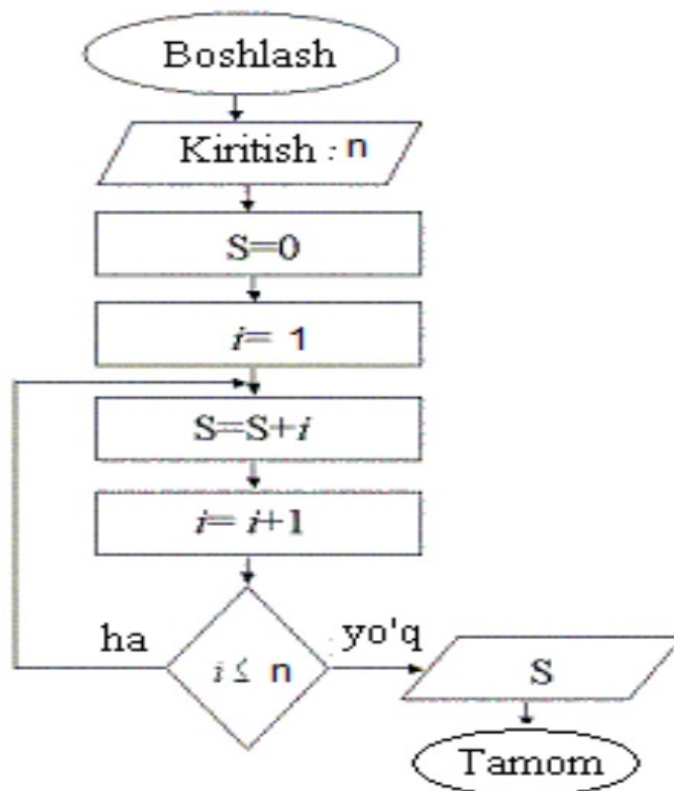
1-misol. Birdan n gacha bo'lgan natural sonlarning yig'indisini hisoblash algoritmini tuzaylik. Masalaning matematik modeli quyidagicha:

$$S = 1 + 2 + 3 + \dots + n = \sum_{i=1}^n i$$

Bu yig'indini hisoblash uchun, avvalo, natiga boshlangich qiymatini $S=0$ va indeksning boshlangich qiymatini $i = 1$ deb olamiz va joriy amallar $S=S+i$ va $i = i + 1$ hisoblanadi. Bu erda birinchi va ikkinchi qadamlar uchun yig'indi hisoblandi va keyingi qadamda i parametr yana bittaga orttiriladi va navbatdagi qiymat avvalgi yig'indi S ga qo'shiladi. Mazkur jarayon shu tartibda indeksning joriy qiymati $i \leq n$ sharti bajarilmaguncha davom ettiriladi va natijada, izlangan yig'indiga ega bo'lamiz. Ushbu fikrlarni quyidagi so'zlar orqali ifodalangan algoritm bilan ifodalash mumkin:

- 1) kiritish (n);
- 2) $S=0$ - natijaning boshlang'ich qiymati;
- 3) $i = 1$ - indeksning boshlang'ich qiymati;
- 4) $S = S + i$ - natijaning joriy qiymatini hisoblang;
- 5) $i = i + 1$ - indeksning joriy qiymatini hisoblang;
- 6) agar ($i \leq n$) sharti tekshirilsin va u bajarilsa =(4) ;
- 7) muhrlash (S).

Bu jarayonga mos keladigan blok-sxemaning ko'rinishi 7.1-rasmda tasvirlangan.



7.1-rasm. 1 dan n-gacha bo'lgan sonlar yig'indisini hisoblash blok-sxemasi

Yuqorida keltirilgan so'zlar asosida ifodalangan algoritmi va blok-sxemadan ko'rinib turibdiki, amallar ketma-ketligining ma'lum qismi parametr i ga nisbatan n marta takrorlanadi.

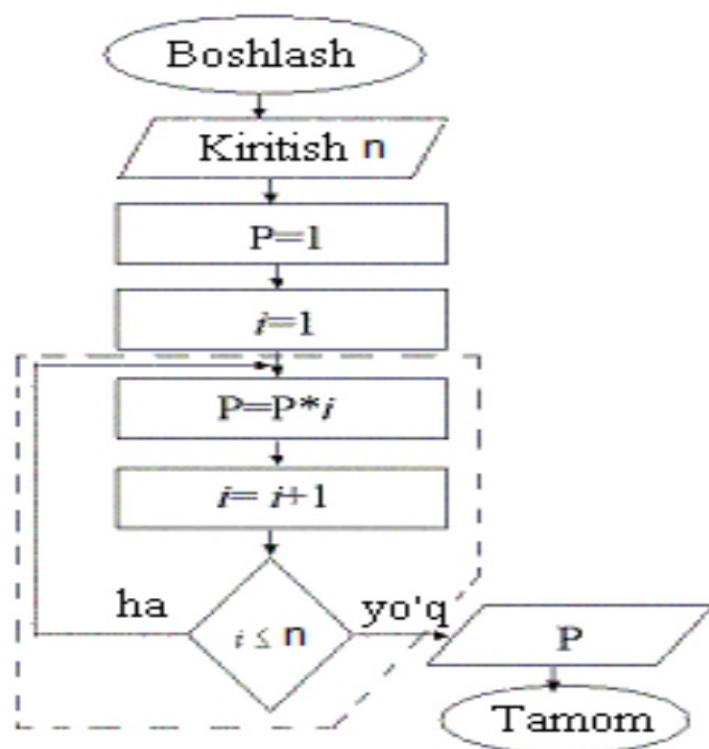
2-misol. Quyidagi ko'paytmani hisoblash algoritmi va blok-sxemasini tuzaylik: $P = 1 \cdot 2 \cdot 3 \cdots n = n!$ (odatda, 1 dan n gacha bo'lgan natural sonlarning ko'paytmasi $n!$ ko'rinishda belgilanadi va "en" faktorial deb ataladi: $P=n!$

(jarayonning matematik modeli: $P = \prod_{i=1}^n i$) [2, 57-58 b.].

Ko'paytmani hosil qilish algoritmi ham yig'indini hosil qilish algoritmiga o'xshash, faqat ko'paytmani hosil qilish uchun, avvalo, $i = 1$ da $P = 1$ deb olinadi, so'ngra $i = i + 1$ da $P = P \cdot i$ munosabatlar hisoblanadi. Keyingi qadamda i parametrning qiymati yana bittaga orttiriladi va navbatdagi qiymat avvalgi hosil bo'lgan ko'paytma - P ga ko'paytiriladi. Bu jarayon shu tartibda to $i \leq n$ sharti bajarilmaguncha davom ettiriladi va natijaviy ko'paytmaning qiymatiga ega bo'lamiz. Quyidagi so'zlar orqali ifodalangan algoritmda bu fikrlar o'z aksini topgan:

- 1) kiritish (n);
- 2) $P = 1$ - natijaning boshlang'ich qiymati;
- 3) $i = 1$ - indeksning boshlang'ich qiymati;
- 4) $P = P \cdot i$ - natijaning joriy qiymatini hisoblash;
- 5) $i = i + 1$ - indeksning joriy qiymatini hisoblash;
- 6) agar ($i \leq n$) shart bajarilsa, u holda =(4);
- 7) muhrlash (P).

Bu algoritmgaga mos blok-sxema 7.2-rasmida keltirilgan.



7.2-rasm. 1 dan n gacha bo'lgan sonlar ko'paytmasini hisoblash blok-sxemasi.

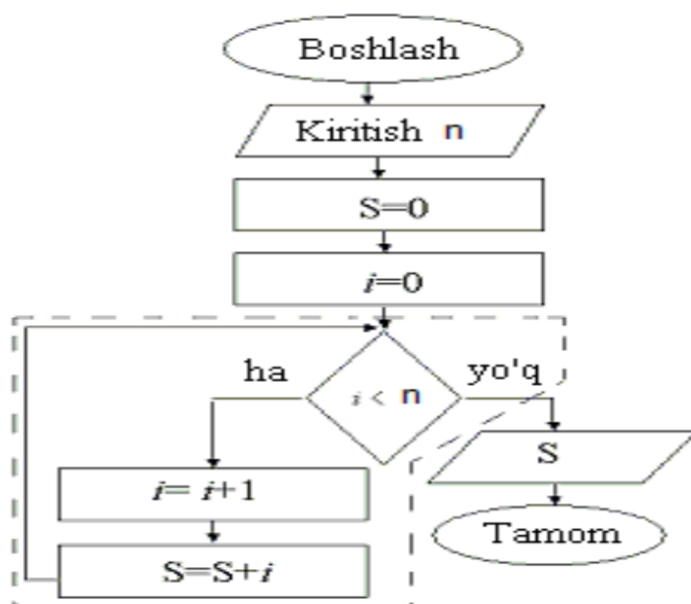
Yuqorida ko'rilgan yig'indi va ko'paytmalarning blok-sxemalaridagi takrorlanuvchi qismlariga (punktir chiziqlar ichiga olingan) 7.2 rasmdagi sharti keyin berilgan takrorlanuvchi struktura mos kelishini ko'rish mumkin.

3-misol. Yuqoridagi blok-sxemalarda shartni oldin tekshiriladigan holatda

chizish mumkin edi. Masalan, $S = \sum_{i=1}^n i$ yig'indini hisoblash algoritmi tadqiqi keltiriladi. Bu masalani yechishda algoritmning takrorlanuvchi qismiga quyidagi sharti oldin berilgan takrorlanuvchi strukturaning mos kelishini ko'rish mumkin.

- 1) kiritish (n);
- 2) $S=0$;
- 3) $i = 0$;
- 4) agar $(i > n) = (8)$;
- 5) $i = i + 1$;
- 6) $S = S + i$;
- 7) shartsiz o'tish=(4);
- 8) muhrlash (S).

Bu algoritmgaga mos blok-sxema 7.3- rasmda keltirilgan.

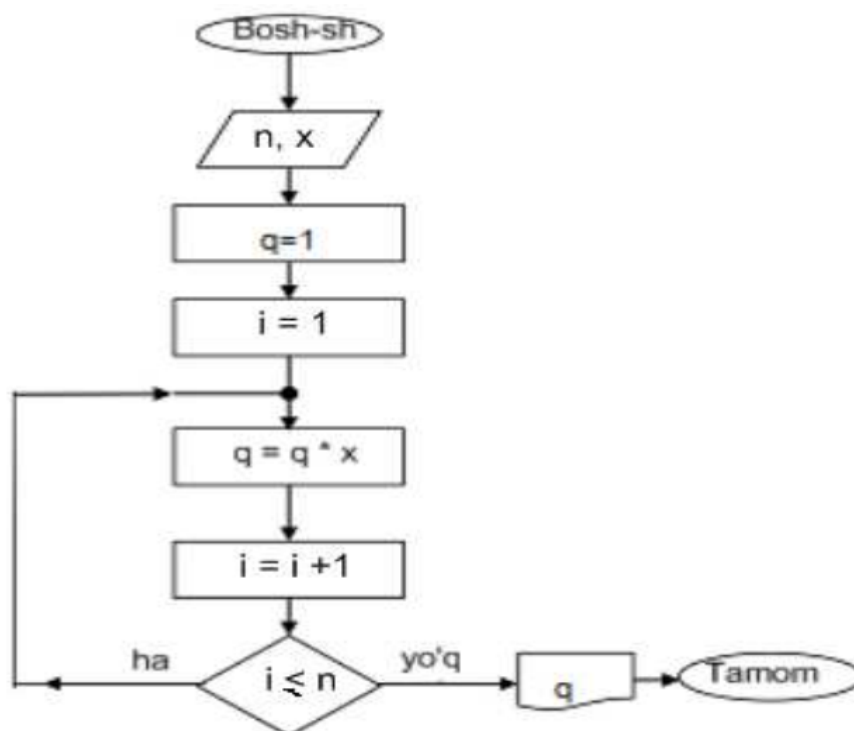


7.3-rasm. 1 dan n gacha bo'lgan sonlar yig'indisini hisoblash blok-sxemasi.

4-misol. Haqiqiy x sonining n chi darajasini hisoblash masalasi ko'riladi. Uning matematik modeli: $q=x^n$ ko'rinishga ega. Takrorlanuvchi jarayonni tashkil etish quyidagidan farqli, yuqoridagilar bilan bir xil:

- ko'paytirish jarayoni uchun boshlang'ich qiymat berilishi: $q = 1$;
- joriy natijani hisoblash: $q = q * x$ ifoda bo'yicha amalga oshiriladi.

Shunday qilib, x ning n chi darajasini hisoblash uchun takrorlanuvchi jarayonni tashkil etish blok-sxemasi 7.4-rasmida keltirilgan.



7.4-rasm. Hisoblash blok-sxemasi

5-misol. Quyidagi munosabatni hisoblash kerak bo'lsin :

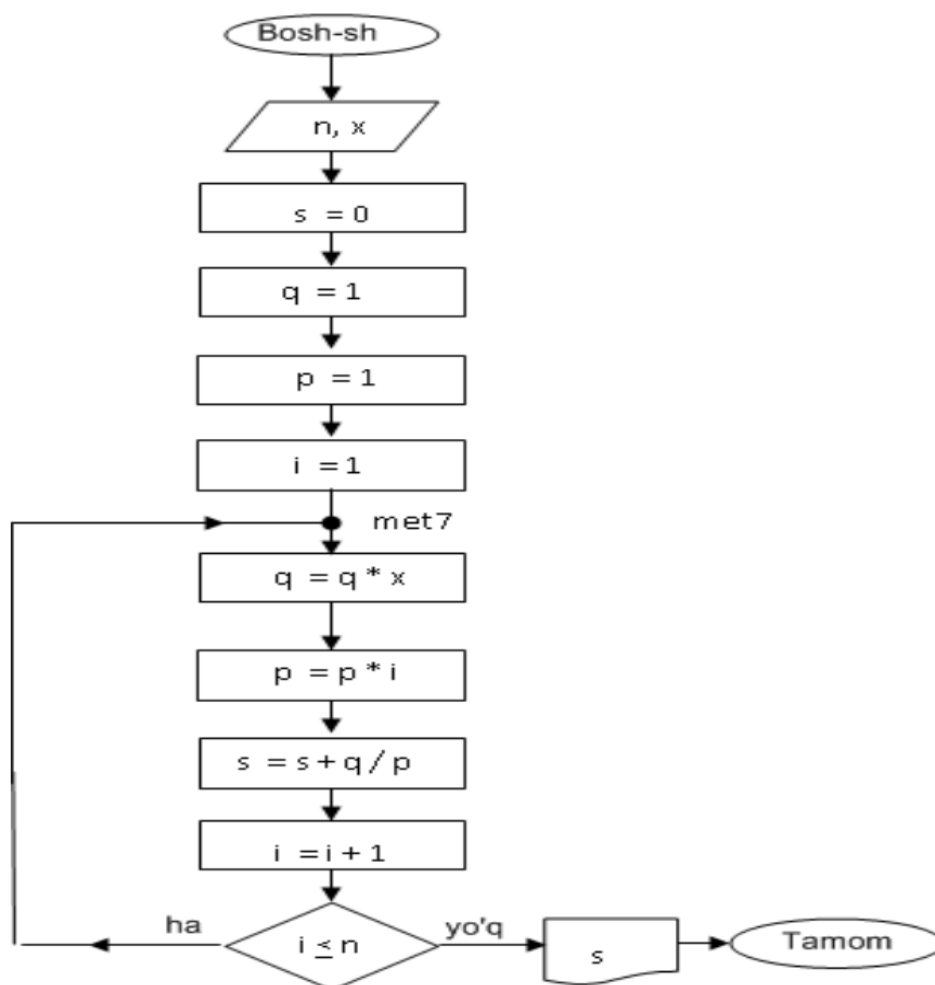
$$S = \sum_{i=1}^n \frac{x^i}{i!}.$$

Munosabatni ochib, quyidagi ko‘rinishda yozish mumkin:

$$s = x^1/1! + x^2/2! + \dots + x^n/n!$$

Masalani yechish algoritmidagi boshlang‘ich qiymat sifatida $s=0$ ni olamiz, chunki ifodada yig‘indi belgisi mavjud. Yig‘indi belgisi ostidagi munosabat kasr sonni anglatadi: suratda - x^i , mahrajda - $i!$. Ularning har biri uchun boshlang‘ich va joriy munosabatlar shakllantiriladi:

	surat	mahraj	natija
boshlang‘ich munosabat	$q = 1$	$p = 1$	$s=0$
joriy munosabat	$q = q * x$	$p = p * i$	$s = s + q / p$



7.5-rasm. Hisoblash blok-sxemasi

Bu jarayonni shakllantirish uchun i indeks-parametri ishlatiladi. Indeks-parametrni boshqarish amallari quyidagicha:

- a) $i = 1$ – parametrning boshlang'ich qiymati,
 b) $i = i + 1$ – parametrning orttirmasi (orttirma $h=1$),
 c) $i \leq n$ – jarayon yakunlanish sharti.

Bunga muvofiq, masalani yechish blok-sxemasi 7.5-rasmdagi ko'rinishga ega bo'ladi.

Topshiriqlar: berilgan masalalarga mos blok-sxema tuzing.

1. Bir to'g'ri chiziqda yotmaydigan uchta nuqta koordinatlari orqali o'tuvchi aylanani yuzasini hisoblash algoritmini tuzing.
2. "Svetofordan (uchchiroqli) foydalanish" algoritmini tuzing.
3. Geron formulasi bo'yicha uchburchak yuzasini hisoblash algoritmining 1.2 da keltirilgan masalaga oid blok-sxemasini tuzing.
4. $y=x^2+2x^4+b$ funksiyani $x=3$ qiymatida hisoblash algoritmining blok-sxemasini tuzing.
5. Quyidagi ifodani hisoblash algoritmini tuzing:
 $y = (\sin(a^2) + \cos^2(a - b \ln(/)) a + b) + c^2$ bu yerda
 $a = (p + q *) b - c$, $b = \sin(e^2 - (p + q^3))$, $c = \operatorname{tg}(3p + q) - b$.
6. Quyidagi ifodani hisoblash algoritmini tuzing:
 $s = (\operatorname{tg}(y^2) + (x - y^3 \ln(/)) x + y^4)$, bu yerda
 $x = (a + b)^2 - (y - c^3)$, $y = e^{(a+b)} - (c^2 - (a - b^3))$.
7. Quyidagi ifodalarni hisoblash algoritmini tuzing:
 $z = (a - b)^3 + \lg^2(2a + 3c) / (\operatorname{tg}^2(c - \sin a))$ bu yerda
 $a = \sin((b - c^2))^3$, $b = \operatorname{tg}(x + y^2)$, $c = \cos(e^{(x-y)} + 5\sin(b + 2))$.
8. Quyidagi ifodalarni hisoblash algoritmini tuzing:
 $f = \lg(x^2 - y + z^3) + \cos^4(x + y)$ bu yerda
 $x = (\cos(a^3 - b^2) + \sin(e^{a+b})) / (a + \cos(y + 2))$, $y = |a^2 - b|$, $z = (a + |y - x|)^2$
9. Haqiqiy a qiymat berilgan. Uchta ko'paytirish amali yordamida a^8 qiymatini hisoblash algoritmini tuzing.
10. Berilgan x uchun quyidagi ifodaning qiymatini hisoblash algoritmini tuzing:
 $A = |x - 2| / (2 - x + \sqrt{x - 2})$.
11. Berilgan ikki $(x1, y1)$ va $(x2, y2)$ nuqtalar orasidagi masofani hisoblash algoritmini tuzing.
12. Geron formulasi asosida uchlari koordinatalari bilan berilgan uchburchak yuzasini hisoblash algoritmini tuzing.
13. Diskriminant hisoblash asosida uchlari koordinatalari bilan berilgan uchburchak yuzasini hisoblash algoritmini tuzing.
14. Teng bo'laklarga bo'lish usuli yordamida $[a, b]$ oralig'ida $f(x) = 0$ tenglamaning yagona ildizini berilgan aniqlik bilan hisoblash algoritmini tuzing.
15. Berilgan 4 ta a, b, c, d haqiqiy sonlar orasidan eng katta va eng kichik qiymatlar o'rtasidagi ayirmani aniqlang.
16. 100 ballik shkala bo'yicha talabalarning to'plagan ballari klaviatura orqali kiritiladi. Ballar taqsimotining qabul qilingan usuliga ko'ra talabalarning ballar

bo'yicha baholarini aniqlash dasturini tuzing.

17.Xizmatchining oylik maoshidan o'sib boruvchi shkala bo'yicha ushlab qolinadigan daromad solig'ini hisoblang.

18.Fibonachchi sonlar qatori: 1,1,2,3,5,... dan dastlabki n – ta sonlarning yig'indisini aniqlang.

19.Aylananing yuzasi S va kvadratning yuzasi R berilgan. Kvadratning aylanaga sig'ishni yoki sig'masligi aniqlash algoritmini tuzing.

20.Berilgan - uchta - a , b , s sonlardan foydalanib tomonlarining uzunliklari shu sonlarga teng bo'lgan uchburchakning mavjudligini aniqlang va shunday uchburchakni yasash mumkin bo'lsa, uning yuzasini hisoblash algoritmini tuzing.

8-amaliy mashg'ulot. Asosiy ma'lumot turlari.

Ishdan maqsad: mavzuga doir nazariy ma'lumotlarni o'rganib, berilgan topshiriqlarni bajarish.

Vazifa.

- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Mavzu bo'yicha testlarni bajarish va savollarga javob berish.

1 . Berilgan nazariy materiallarni o'rganish.

Keyingi yillarda amaliy dasturchilarga juda ko'p integratsion dastur tuzish muhitlari taklif etilayapti. Bu muhitlar u yoki bu imkoniyatlari bilan bir-biridan farq qiladi. Aksariyat dasturlashtirish muhitlarining fundamental asosi C++ tiliga borib taqaladi.

C++ tili tarixi.

Birinchi elektron hisoblash mashinalari paydo bo'lishi bilan dasturlash tillari evolyutsiyasi boshlanadi. Dastlabki kompyuterlar ikkinchi jahon urushi vaqtida artilleriya snaryadlarining harakat traektoriyasini hisob-kitob qilish maqsadida qurilgan edi. Oldin dasturchilar eng sodda mashina tilini o'zida ifodalovchi kompyuter komandalari bilan ishlaganlar. Bu komandalar nol va birlardan tashkil topgan uzun qatorlardan iborat bo'lar edi. Keyinchalik, insonlar uchun tushunarli bo'lgan mashina komandalarini o'zida saqlovchi (masalan, ADD va MOV komandalari) assembler tili yaratildi. SHu vaqtlarda BASIC va COBOL singari yuqori sathli tillar ham paydo bo'ldiki, bu tillar tufayli so'z va gaplarning mantiqiy konstruksiyasidan foydalanib dasturlash imkoniyati yaratildi. Bu komandalarni mashina tiliga interpretatorlar va kompilyatorlar ko'chirar edi. Interpretator dasturni o'qish jarayonida uning komandalarini ketma - ket mashina tiliga o'tkazadi. Kompilyator esa yaxlit programma kodini biror bir oraliq forma - ob'ekt fayliga o'tkazadi. Bu bosqich kompilyatsiya bosqichi deyiladi. Bundan so'ng kompilyator ob'ektlili faylni bajariluvchi faylga aylantiradigan kompanovka dasturini chaqiradi.

Interpretatorlar bilan ishlash osonroq, chunki dastur komandalari qanday ketma - ketlikda yozilgan bo'lsa shu tarzda bajariladi. Bu esa dastur bajarilishini nazorat qilishni osonlashtiradi. Kompilyator esa kompilyatsiya va kompanovka

kabi qo'shimcha bosqichlardan iborat bo'lganligi uchun ulardan hosil bo'ladigan bajariluvchi faylni tahlil qilish va o'zgartirish imkoniyati mavjud emas. Faqatgina kompilyatsiya qilingan fayl tezroq bajariladi, chunki bundagi komandalar kompilyatsiya jarayonida mashina tiliga o'tkazilgan bo'ladi.

C++ kabi kompilyatsiya qiluvchi dasturlash tillarini yana bir afzalligi hosil bo'lgan dastur kompyuterda kompilyatorsiz ham bajarilaveradi. Interpretatsiya qiluvchi tillarda esa tayyor dasturni ishlatish uchun albatta mos interpretator dasturi talab qilinadi.

Ayrim tillarda (masalan, VISUAL BASIC) interpretator rolini dinamik bibliotekalar bajaradi. Java tilining interpretatori esa virtual mashinadir (Virtual Machine, yoki VM). Virtual mashinalar sifatida odatda brouzer (Internet Explorer yoki Netscape) lar qo'llaniladi.

Ko'p yillar davomida dasturlarning asosiy imkoniyati uning qisqaligi va tez bajarilishi bilan belgilanib kelinardi. Dasturni kichikroq qilishga intilish kompyuter xotirasini juda qimmatligi bilan bog'liq bo'lsa, uning tez bajarilishiga qiziqish protsessor vaqtining qimmatbaholigiga bog'liq edi. Lekin kompyuterlarning narxi tushishi bilan dastur imkoniyatini baholash mezonlari o'zgardi. Hozirgi kunda dasturchining ish vaqti biznesda ishlatiladigan ko'pgina kompyuterlarning narxidan yuqori. Hozirda professional tarzda yozilgan va oson ekspluatatsiya qilinadigan dasturlarga talab oshib bormoqda. Ekspluatatsiyaning oddiyligi, konkret masalani yechish bilan bog'liq bo'lgan talabni ozroq o'zgarishiga, dasturni ortiqcha chiqimlarsiz oson moslashtirish bilan izohlanadi.

Dasturlar

Dastur so'zi ham komandalarning alohida blokini (berilgan kodini) aniqlovchi so'z, ham yaxlit holdagi bajariluvchi dasturiy mahsulotni belgilovchi so'z sifatida ishlatiladi.

Bu ikki xillilik o'quvchini chalg'itishi mumkin. SHuning uchun unga aniqlik kiritamiz. Demak dasturni yo dasturchi tomonidan yoziladigan komandalar to'plami, yoki amallar bajaradigan kompyuter mahsuloti sifatida tushinamiz.

Dasturchilar oldida turgan masalalar

Vaqt o'tishi bilan dasturchilar oldiga quyilgan masalalar o'zgarib boryapti. Bundan yigirma yil oldin dasturlar katta hajmdagi ma'lumotlarni qayta ishlash uchun tuzilar edi. Bunda dasturni yozuvchi ham, uning foydalanuvchisi ham kompyuter sohasidagi bilimlar bo'yicha professional bo'lishi talab etilardi. Hozirda esa ko'pgina o'zgarishlar ro'y berdi. Kompyuter bilan ko'proq uning apparat va dasturiy ta'minoti haqida tushunchalarga ega bo'lmagan kishilar ishlashyapti. Kompyuter odamlar tomonidan uni chuqur o'rganish vositasi emas, ko'proq o'zlarining oldilariga qo'yilgan, o'zlarining ishlariga tegishli bo'lgan muammolarini yechish instrumenti bo'lib qoldi.

Foydalanuvchilarning ushbu yangi avlodini dasturlar bilan ishlashlarini osonlashtirilishi bilan bu dasturlarning o'zini murakkabligi darajasi oshadi. Zamonaviy dasturlar - foydalanuvchi bilan do'stona munosabatni yuqori darajada tashkil qiladigan ko'p sondagi oynalar, menyu, muloqot oynalari va vizual

grafikaviy muhitlardan tarkib topgan interfeysga ega bo'lishi lozim.

Dasturlashga talabni o'zgarishi nafaqat tillarning o'zgarishiga balki uni yozish texnologiyasini ham o'zgarishiga olib keldi. Dasturlash evolyutsiyasi tarixida ko'pgina bosqichlar bo'lishiga qaramay biz bu kursimizda protsedurali dasturlashdan ob'ektlarga mo'ljallangan dasturlashga o'tishni qaraymiz.

Protseduraviy, strukturaviy va ob'ektlarga mo'ljallangan dasturlash

SHu vaqtgacha dasturlar berilgan ma'lumotlar ustida biror bir amal bajaruvchi protseduralar ketma-ketligidan iborat edi. Protsedura yoki funktsiya ham o'zida aniqlangan ketma-ket bajariluvchi komandalar to'plamidan iboratdir. Bunda berilgan ma'lumotlarga murojaatlar protseduralarga ajratilgan holda amalga oshiriladi.

Strukturaviy dasturlashning asosiy g'oyasi «bo'lakla va hukmronlik qil» printsiptiga butunlay mos keladi. Kompyuter dasturini masalalar to'plamidan iborat deb qaraymiz. Oddiy tavsiflash uchun murakkab bo'lgan ixtiyoriy masalani bir nechta nisbatan kichikroq bo'lgan tarkibiy masalalarga ajratamiz va bo'linishni toki masalalar tushunish uchun yetarli darajada oddiy bo'lguncha davom ettiramiz.

Misol sifatida kompaniya xizmatchilarining o'rtacha ish haqini hisoblashni olamiz. Bu masala sodda emas. Uni qator qism masalalarga bo'lamiz:

1. Har bir xizmatchining oylik maoshi qanchaligini aniqlaymiz.
2. Kompaniyaning xodimlari sonini aniqlaymiz.
3. Barcha ish haqlarini yig'amiz.
4. Hosil bo'lgan yig'indini kompaniya xodimlari soniga bo'lamiz.

Xodimlarning oylik maoshlari yig'indisini hisoblash jarayonini ham bir necha bosqichlarga ajratish mumkin.

1. Har bir xodim haqidagi yozuvni o'qiymiz.
2. Ish xaqi to'g'risidagi ma'lumotni olamiz.
3. Ish haqi qiymatini yig'indiga qo'shamiz.
4. Keyingi xodim haqidagi yozuvni o'qiymiz.

O'z navbatida, har bir xodim haqidagi yozuvni o'qish jarayonini ham nisbatan kichikroq qism operatsiyalarga ajratish mumkin:

1. Xizmatchi faylini ochamiz.
2. Kerakli yozuvga o'tamiz.
3. Ma'lumotlarni diskdan o'qiymiz.

Strukturaviy dasturlash murakkab masalalarni yechishda yetarlicha muvofaqiyatli uslub bo'lib qoldi. Lekin, 1980 – yillar oxirlarida Strukturaviy dasturlashning ham ayrim kamchiliklari ko'zga tashlandi.

Birinchidan, berilgan ma'lumotlar (masalan, xodimlar haqidagi yozuv) va ular ustidagi amallar (izlash, tahrirlash) bajarilishini bir butun tarzda tashkil etilishidek tabiiy jarayon realizatsiya qilinmagan edi. Aksincha, protseduraviy dasturlash berilganlar strukturasi bu ma'lumotlar ustida amallar bajaradigan funktsiyalarga ajratgan edi.

Ikkinchidan, dasturchilar doimiy tarzda eski muammolarning yangi yechimlarini ixtiro qilar edilar. Bu situatsiya ko'pincha velosipedni qaytam ixtiro qilish ham deb aytiladi. Ko'plab dasturlarda takrorlanuvchi bloklarni ko'p

martalab qo'llash imkoniyatiga bo'lgan hohish tabiiydir. Buni radio ishlab chiqaruvchi tomonidan priyomnikni yig'ishga o'xshatish mumkin. Konstruktor har safar diod va tranzistorni ixtiro qilmaydi. U oddiygina – oldin tayyorlangan radio detallaridan foydalanadi xolos. Dasturiy ta'minotni ishlab chiquvchilar uchun esa bunday imkoniyat ko'p yillar mobaynida yo'q edi.

Amaliyotga do'stona foydalanuvchi interfeyslari, ramkali oyna, menyu va ekranlarni tadbiq etilishi dasturlashda yangi uslubni keltirib chiqardi. Dasturlarni ketma-ket boshidan oxirigacha emas, balki uning alohida bloklari bajarilishi talab qilinadigan bo'ldi. Biror bir aniqlangan hodisa yuz berganda dastur unga mos shaklda ta'sir ko'rsatishi lozim. Masalan, bir knopka bosilganda faqatgina unga birlashtirilgan amallar bajariladi. Bunday uslubda dasturlar ancha interaktiv bo'lishi lozim. Buni ularni ishlab chiqishda hisobga olish lozim.

Ob'ektga mo'ljallangan dasturlash bu talablarga to'la javob beradi. Bunda dasturiy komponentlarni ko'p martalab qo'llash va berilganlarni manipulyatsiya qiluvchi metodlar bilan birlashtirish imkoniyati mavjud.

Ob'ektga mo'ljallangan dasturlashning asosiy maqsadi berilganlar va ular ustida amal bajaruvchi protseduralarni yagona ob'ekt deb qarashdan iboratdir.

2 . Mavzu bo'yicha testlarni bajarish va savollarga javob berish.

Topshiriq: berilgan testlarni bajarish va savollarga javob yozish.

1.Axborot hajmining eng kichik o'lchov birligi nima?

- A) Bit B) Son S) Belgi D) Bayt

2.Algoritm so'zi qaysi olimning nomi bilan bog'liq?

- A) Beruniy B) Al-Xorazmiy S) Ulug'bek D) Ibn Sino

3. Kompyuter nima?

- A) Chop qilish qurilmasi B) Faqat yozishga mo'ljallangan qurilma
C) Faqat o'qish uchun mo'ljallangan qurilma
D) Ma'lumotlar ustida turli amallar bajaruvchi kichik hajmdagi elektron hisoblash mashinasi.

4. Informatika fani nimani o'rgatadi?

- A) Axborotlarni o'lchashning usulini o'rgatadi
B) Axborotlarni uzatish usullarini o'rgatadi
C) Axborotni qayta ishlash usullarini o'rgatadi.
D) Axborotlarni qabul qilish, saqlash, qayta ishlash va uzatish usullarini o'rgatadi.

5. Qanday chop qilish qurilmalari bor?

- A) Matritsali, purkagichli, lazerli. B) Ignali, lazerli C) Lazerli,
D) Purkagichli, lazerli

6.Algoritm so'zi qaysi olim nomidan olingan.

- A) Al-Farg'oniy B) A.Navoiy C) Al Xorazmiy D) Pifagor

7.Windowsning ish stolida nimalar mavjud?

- A) Yorliqlar, fayllar B) Timsollar, buyruqlar C) Yorliqlar, timsollar,
Korzina, "Moy kompyuter", masalalar paneli D) Masalalar paneli

8. "Alt" + "F4" klaviatura tugmachalari birgalikda bosilganda qanday amal bajariladi?

- A) dastur yuklanadi. B) dasturdan chiqiladi. C) Fayldan chiqiladi.
D) Yangi fayl ko'riladi.

9. Qatorlar orasidagi intervallarni belgilash *Format* menyusining qaysi bandi yordamida bajariladi?

- A) Шрифт. B) Абзац. C) Табуляция. D) Стилъ.

10. Kiritilayotgan ma'lumotlarni kompyuter xotirasiga olish uchun qaysi tugmachalardan foydalaniladi?

- A) F1. B) F4. C) Shift + F12. D) F5.

Savollar:

1. Yuqori darajadagi dasturlash tillari nima?
2. Kompilyatsiya jarayonini necha bosqichdan tashkil topadi?
3. Qanday belgilar bilan tugagan barcha belgilar ketma-ketligi izoh hisoblanadi?
4. C++ tilining kalit so'zlariga qaysilar kiradi?
5. Protsessor registrlarini belgilash uchun qaysi so'zlar ishlatiladi?
6. Kompilyatsiya jarayoni nima?
7. Kiritilgan qiymatni o'zgaruvchi turiga mos kelishini qanday tekshirish mumkin?
8. Identifikator nima?
9. Sintaksis – qanday tilning qoidalar?
10. C++ instruksiyalari qanday belgi bilan tugallanishi zarur?

9-amaliy mashg'ulot. Zamonaviy ob'ektga mo'ljallangan algoritmik tillar.

Ishdan maqsad: mavzuga doir nazariy ma'lumotlarni o'rganib, berilgan topshiriqlarni bajarish.

Vazifa.

1. Berilgan nazariy materiallarni o'rganish.
2. Mavzu bo'yicha testlarni bajarish va savollarga javob berish.

1. Berilgan nazariy materiallarni o'rganish.

Ob'ektga yo'naltirilgan dasturlash atamasi dastlab 60-yillarning o'rtalarida, Simula-67 dasturlash tilida paydo bo'lgan. Ammo, bu til FORTRAN, ALGOL, PL/1 kabi tillarga nisbatan o'zining imkoniyatlari past bo'lganligi hamda shu davrda yechish talab qilingan masalalar uchun yetarli vositalarni taklif qila olmagan uchun dasturchilar orasida keng ommalasha olmadi. Keyinchalik ob'ektga yo'naltirilgan dasturlash Smalltalk, Object Pascal, C++, Ada, hozirda esa C# kabi yuqori darajali dasturlash tillarining topologiyasi asosida rivojlandi va ommalashdi.

Ob'ektga yo'naltirilgan dasturlash – bu dasturlashning shunday yangi yo'nalishiki, dasturiy tizimda o'zaro aloqada bo'lgan ob'ektlar majmuasi sifatida qaraladi va xar bir ob'ektni ma'lum bir klassga mansub hamda xar bir klass qandaydir shajarani hosil qiladi deb hisoblanadi. Alohida olingan klass ma'lumotlar to'plami va ular ustida bajariladigan amallarning to'plami sifatida qaraladi. Bu klassning elementlariga faqat shu klassda aniqlangan amallar orqali

murojaat qilish mumkin. Dasturdagi ma'lumotlar va ular ustida bajariladigan amallar o'rtasidagi o'zaro bog'liqlik an'anaviy dasturlash tillariga nisbatan dasturiy tizimlarning ishonchliligini ta'minlaydi. Ob'ektga yo'naltirilgan dasturlashning eng asosiy tushunchasi ob'ekt va klass hisoblanadi.

Ob'ekt. Oldimizda turgan olma mevasini dasturlash terminologiyasi orqali ko'raylik. An'anaviy dasturlashda biz uni qismlarga bo'lib o'rganganmiz: S-olma po'stining yuzi, J – olma mevasidagi sharbat hajmi, F – po'st orasidagi meva og'irligi, D-urug'lar soni va x.k. Endi ana shu olmaga rassom ko'zi bilan qaraylik. Olmaning rasmi bu meva emas, balki olma mevasining yassi tekisliklikdagi tasviridir. Uni har biri alohida va bir-biriga bog'liq bo'lmagan biror ma'lumotlar segmentida turgan bir nechta ma'lumotlardan iborat bo'lgan ko'rinishda abstraksiyalash mumkin emas. Olmaning komponentalari doimo birgalikda va bu komponentalar orasidagi o'zaro aloqalarni saqlagan holda qaraladi.

Ob'ekt - biz yashayotgan olamdagi biror elementga hos bo'lgan barcha ma'lumot va hulqlarni, ya'ni shu element ustida bajarish mumkin bo'lgan xarakatlarni ifodalaydi hamda ma'lumotlarning tugal abstraksiyasidan iborat bo'ladi. Bu ma'lumot va hulqlar ob'ektga yo'naltirilgan dasturlash atamasida mos ravishda hususiyat va metod deb ataladi. Hususiyatni ob'ektning maydoni deb ham yuritiladi. Masalan, shashka ob'ekti rang, vertikal maydondagi o'rni, gorizontal maydondagi o'rni kabi maydonlarga, surish, urish, «damka» ga chiqish, shashka taxtasidan chetga olish kabi metodlarga ega bo'ladi. Maydon va metodlar birgalikda ob'ektning a'zolari deyiladi. Ob'ektlarning strukturasi ularning o'zaro aloqasini ifodalaydi. Ob'ektlar o'zlarining barcha xarakteristika va hulqlarining o'ziga hos tomonlarini birgalikda saqlaydi.

Klass. Har bir ob'ekt qandaydir bitta klassga taaluqli bo'ladi. Klass – bu murakkab struktura bo'lib, o'z ichiga ma'lumotlarni, protsedura va funksiyalarni ifodalashdan tashqari, klasslarning vakili bo'lmish ob'ektlar ustida bajarilishi mumkin bo'lgan amallarni ham oladi. Klassdagi ma'lumotlar maydonlar, protsedura va funksiyalar esa metodlar deb ataladi. Klassning strukturasi tushunchasi ham muhim hisoblanadi. U tizim ichidagi vositalar boyligini namoyon qiladi. Ma'lumki, biror yaproqdagi fotosintez jarayonini o'rganish uchun shu yaproqdagi bitta xujayrani ko'rish yetarli, chunki, qolgan xujayralar ham o'zini ana shu o'rganilgan xujayra kabi tutadi. Biz xam biror klass, tip yoki bo'linga mansub bo'lgan ob'ektni ko'rar ekanmiz, faraz qilishimiz mumkinki, uning hulqi ham shu tipdagi boshqa ob'ektlarniki kabi bo'ladi.

Ob'ektga yo'naltirilgan dasturlash bizning hulqimizga o'xshab ketadigan dasturlash usulini ifodalaydi. U dasturlash tillarini ishlab chiqishdagi yangiliklarning tabiiy evolyusiyasi hisoblanadi. U barcha avvalgi dasturlash tillariga qaraganda strukturaliroq, ma'lumotlarni abstraktlashga urinishda abstraktliroq va modulliroqdir.

Ob'ektga yo'naltirilgan dasturlash o'zining uchta asosiy hususiyatlari bilan xarakterlanadi:

1. Inkapsulyasiya – yozuvlarni shu yozuvlarning maydonlari ustida amallar bajarish uchun mo'ljallangan funksiyalar va protseduralar bilan birgalikda olib boradi. Inkapsulyasiya yashirish prinsipiga ham ega bo'ladi. Buning ma'nosi

shuki, ob'ekt ustida bajarish mumkin bo'lgan xarakatlarni amalga oshirish vositalari shu ob'ektdan foydalanayotgan dasturchilar ko'zidan yashirib qo'yiladi. Dasturchi faqat shu ob'ektning ayrim metod va maydonlari bilangina ishlay oladi. Boshqacha aytganda, barcha maydon va metodlar ichki va tashqi guruhlariga bo'linadi. Ob'ektning ichki a'zolari dasturchiga «ko'rinmaydi» va ob'ektning hulqi va imkoniyatlarini aniqlaydi, tashqi a'zolari esa dasturchiga «ko'rinib turadi» hamda ob'ektni boshqarish imkonini beradi. Ob'ektning dasturchiga ko'rinib, ob'ektni boshqarishga yordam beradigan metod va maydonlari (hususiyatlari) ob'ektning interfeysi deb ataladi. Dasturchi ob'ekt bilan ishlash uchun uning interfeysini bilishi kifoya.

Masalan, yengil avtomobilni boshqarishni o'rganish uchun uning matorining ishlash prinsipi, g'ildiraklarning burilishini, tormoz mexanizmini o'rganishning hojati yo'q, rulni burash, pedal yoki uzatmalar qutisi richagini bosishni bilish yetarli.

2. Vorislik – yangi ob'ektni aniqlashga ehtiyoj paydo bo'lganda oldindan ma'lum bo'lgan ob'ektlardan foydalanish. Ob'ektlar o'zlarini yaratishda ishtirok etgan ota ob'ektlarning xarakteristika va hulqlarini meros qilib olishlari mumkin. OYD konsepsiyasi yangi klasslarni mavjud klasslarga yangi maydonlar, hususiyatlar va metodlarni qo'shish orqali yaratish imkonini ham beradi. Yangi klasslarni tashkil qilishning bunday usuli yuzaga keltirish deb ataladi. Bu holda yuzaga kelgan yangi klass o'zining bazaviy ota klassiga hos bo'lgan hususiyat va metodlarini meros oladi.

Masalan hayotdan misol qilib, hasharotlar klassini olish mumkin. U ikki guruhga bo'linadi: qanotlilar va qanotsizlar. Qanotli hasharotlarga kapalaklar, pashshalari, parvonalar va x.k. lar kiradi. Shuning uchun, pashshalarni qaytadan qanotli deb ta'riflashning hojati yo'q, u bu hususiyatni ota klassi bo'lgan qanotli hasharotlar klassidan meros qilib oladi.

3. Polimorfizm – xarakatga biror nom berish hamda undan birgalikda ob'ektlar shajarasining quyi va yuqori qismlarida foydalanish. Bunda shajaraning har bir ob'ekti bu xarakatni o'zi uchun hos bo'lgan usul bilan bajaradi. Polimorfizm — bu turli klasslarga kirgan metodlar uchun bir hil nomlardan foydalanish imkoniyatidir. Polimorfizm konsepsiyasida ob'ektga nisbatan metod qo'llanganida aynan ob'ektning klassiga mos keluvchi metoddan foydalanishni ta'minlaydi.

Object Pascal dasturlash tili ob'ektga yo'naltirilgan dasturlashga hos bo'lgan barcha vositalarni bera oladi: strukturalilik, modullik, katta abstraktlilik. Bu xarakteristikalarining hammasi xizmat ko'rsatish uchun juda ham sodda, osongina boshqa vaziyatlarga moslashtirila oladigan, kuchliroq strukturaga ega bo'lgan kodlarda o'z aksini topadi.

Ob'ektga yo'naltirilgan dasturlash uzoq yillar davomida an'anaviy, ya'ni standart hisoblangan dasturlashga nisbatan hos bo'lgan tasavvurlarni bir chetga qo'yishni talab qiladi. Natijada ob'ektga yo'naltirilgan dasturlash juda ham sodda, ko'rgazmaliligi yuqori bo'lib, dasturiy ta'minot yaratishdagi ko'plab muammolarni hal qilishning juda ajoyib vositasiga aylanadi.

Yuqorida aytilgan fikrlarni hisobga olsak, ob'ektga yo'naltirilgan dasturlash

texnologiyalari quyidagicha masalalarni hal qilishi mumkin:

1. An'anaviy dasturlash tillarida mavjud bo'lgan kamchiliklarni bartaraf qilish;
2. An'anaviy dasturlash tillari yordamida yechib bo'lmaydigan yoki juda katta qiyinchiliklar bilan yechilishi mumkin bo'lgan masalalarni hal qilish;
3. Qayta ishlash mumkin bo'lgan ma'lumotlar va ularning tiplari doirasi an'anaviy dasturlash tillariga nisbatan ancha keng;
4. Foydalanuvchilar uchun qulay bo'lgan muloqot interfeysini yaratish;
5. Kiritilayotgan va chiqarilayotgan turli tipdagi ma'lumotlarni nazorat qilish;
6. Yangi tipdagi ma'lumotlar, klasslar va modullarni osongina tashkil etish va ma'lumotlarni nazorat qilish;
7. Multimedia va animatsion vositalaridan foydalanib, turli darajadagi tovushli va xarakatli effektlarni hosil qilish va qayta ishlash;
8. Ma'lumotlar bazasi va undagi ma'lumotlar ustida amallarni bajarish, SQL so'rovnomalari yordamida ma'lumotlarni qidirib topish kabi masalalar juda osonlik bilan hal qilish;
9. OLE konteyneri yordamida WINDOWS muhiti uchun mo'ljallangan ilovalardagi ob'ektlar bilan ishlash;
10. Foydalanuvchilar uchun yaratilgan dasturiy ta'minotdan foydalanish uchun yordamchi ma'lumotnomalar tizimini yaratish;
11. Dasturiy ta'minotni boshqa kompyuterlarga ko'chirish uchun o'rnatuvchi diskarni yaratish;
12. Dastur matnini tashkil qilishda yuzaga kelishi mumkin bo'lgan xatoliklar bilan ishlash masalasini hal qilish va x.k.

Ob'ekt ob'ektga yo'naltirilgan dasturlash(OYD) texnologiyasining eng asosiy kalit tushunchasidir. Atrofqa qarang, haqiqiy hayotdagi bir necha ob'ektlarni ko'rishingiz mumkin: stol, uy, it, mushuk, televizor va h.k.

Ularning barchasining albatta xususiyatlari va bajaradigan vazifalari (funktsiyalari) bor. Masalan, Mushuk xususiyatlari: rangi, qorni to'qligi, yoshi, jinsi; funktsiyalari: ovqat yeyishi, myovlashi, yurishi, sichqon tutishi. Mashina, xususiyatlari: tezligi, rangi, nomi, narxi; funktsiyalari: yurishi, to'xtashi, oyna artgichlarining ishlashi, eshiklarning ochilib yopilishi v.h.k. Bu kabi hayotiy misollarning xususiyatlari va funktsiyalarini aniqlash OYD nuqtai nazaridan fikrlashning eng zo'r ko'rinishidir.

Bir daqiqaga to'xtang va hozirda atrofingizdagi biror narsalarni analiz qiling. Har bir ob'ekt uchun o'zingizdan so'rang: "Bu ob'ektning qanday xususiyatlari bor?" , "Qanday vazifalarni bajaradi?" kabi. Va kuzatish natijalaringizni yozib oling, sezgan bo'lsangiz tuziladigan ro'yxat ob'ektning murakkabligiga qarab ko'payib boradi. Kompyuter indikatorining 2 ta *xususiyati* bor o'chiq va yoniq: *funktsiyalari* esa yonish va o'chish. Bu barcha kuzatishlar OYD dunyosiga o'tkazish mumkin.

Dasturlashdagi ob'ekt.

Dasturlashdagi ob'ekt(bundan keyin oddiygina *ob'ekt* deb ketiladi) ham

haqiqiy hayotdagi ob'ektlarga o'xshash: Ular ham qandaydir xususiyatlar va bajaradigan funktsiyalardan iborat bo'ladi. Ob'ektning xususiyatlari har xil dasturiy *o'zgaruvchilardan* iborat bo'ladi va ularning o'zgartirish uchun qandaydir *funktsiyalar* bajariladi. Bunday *funktsiyalar* bilan o'zgaruvchilarning holatini berkitish mumkin ya'ni aynan o'sha o'zgaruvchini tashqaridan o'zgartirish uchun albatta maxsus funktsiyadan foydalanish kerak bo'ladi. Bu jarayon "**Enkapsulatsiya**" deb atalib, OYDning eng muxim tushunchalaridan biridir. Hech e'tibor berganmisiz dorilarda ham shu termin ishalitladi ya'ni kapsula(ustidan maxsus modda bilan o'ralgan dorilar), buni misolni **Enkapsulatsiya** jarayoni esda yaxshi qolishi va tushunarli bo'lishi uchun keltirdim.

Mashinani tasavvur qiling, uni dasturlash obyekt sifatida modellashtiramiz: Uning o'zgaruvchilari(hozirgi tezligi, qolgan benzini, va h.k) va uning funktsiyalari(to'xtatish, tezlikni oshirish, rulni burish, va h.k.). Bu yerda uning bakidagi benzini yurishi tufayli kamayib boradi demak uning qiymatining o'zgarishi 0 dan bakning sig'imigacha bo'ladi, yoki uning tezligi ham shu kabi aynan qaysidir funktsiyalarning amalga oshirilishi orqali u ham 0 dan maksimal tezligigacha o'zgarishi mumkin. Bulardan tashqari mashinaning ba'zi xususiyatlari borki ular o'zgarishligi mumkin, masalan, rangi.

Demak, ko'rinib turiptiki mashina ham o'z navbatida bir necha mayda obyektlardan iborat bo'ladi. Va albatta ularni kodda yozganda ham alohida ob'ekt sifatida ifodalash kerak bu orqali nimalarga erishish mumkin:

Qismlilik: Har bir ob'ektga tegishli bo'lgan kodlar alohida-alohida, boshqa obyektlarga bog'liq bo'lmagan holda boshqarish imkoniyatiga ega bo'lamiz. Bu hammasi emas, tasavvur qiling mashina ob'ektini ifodalovchi kodni bo'lmasdan faqat bitta faylda ifodaladik; bu esa murakkabligiga qarab yuzlab hatto minglab qatorli kod bo'lishi mumkin. Undan biror narsani topib-o'zgartrish ancha mashaqqat bo'ladi.

Qayta foydalanish: Yana boshqa plyus tarafi biz bo'laklagan mashinaning detallarini boshqa ob'ektlarda ham ishlatishimiz mumkin. Masalan, 2 xil mashina ularning shunday qismlari borki aynan bir xil, ana o'shalar uchun ikki marta alohida kod yozmasdan, bitta yozganimizni qayta ishlatishimiz mumkin.

Uzib-ulanuvchanligi: buni tushunish uchun yuqoridagi misoldan foydlanamiz, aytaylik, mashinaning biror qismi ishlamayapti, xo'sh nima qilinadi? yoki ishlab turgan boshqasiga almashtiramiz, yoki tuzatamiz. Mashinaning biror bolti buzilsa uni boshqa ishlab turgani bilan almashtirasiz yoki tuzatamiz lekin mashinani almashtirmaymiz.

2 . Mavzu bo'yicha testlarni bajarish va savollarga javob berish.

1. Saxifalab pastga yurish tugmasi qaysi?
A) PageUp. B) Home. C) End. D) Page Down
2. Xat boshiga joy tashlash uchun qaysi tugmadan foydalaniladi?
A) End B) Esc C) Delete D) Tab.
3. Format buyruqlar to'plami nima vazifani bajaradi?
A)matn, formula va grafiklarni formatlovchi buyruqlar to'plami;

- B) hisoblash jarayonini boshqaruvchi buyruqlar to'plami;
 C) xujjatga turli ob'ektlarni joylashtirish uchun xizmat qiluvchi buyruqlar to'plami
 D) matn, formula va grafiklarni formatlovchi buyruqlar to'plami;
4. DELETE tugmasining vazifasi nima?
 A) Matn boshiga borish B) Abzats boshiga borish
 C) Ko'rsatkichdan oldingi (chap tomondagi) belgini o'chirish
 D) Ko'rsatkich turgan belgini va undan o'ng tomondagi belgilarni o'chirish.
5. End tugmasining vazifasi ?
 A) Sahifalab quyiga yurish B) Xat boshiga borish
 C) Sahifalab yuqoriga yurish D) qator oxiriga borish
6. Qator boshiga borish uchun qaysi yozuvli tugmacha kerak bo'ladi?
 A) Home B) End C) Page Up D) Esc
7. "Pusk" tugmasining vazifasi nimadan iborat?
 A) OS ni ishga tushirishdan. B) Windows ning barcha dasturlarini ishga tushirishdan.
 C) Word ni ishga tushirishdan. D) Excel ni ishga tushirishdan.
8. WORD taxrirlagichi ishga tushganda, kompyuterning o'zi faylga qanday nom beradi?
 A) ПАПКА 1 B) Документ 1 C) Текст1 D) WORD 1
9. Matnda kursor qanday ko'rinishda bo'ladi?
 A) Ko'rsatkich B) Gorizontal chiziqcha C) Vertikal chiziqcha
 D) qum soat
10. Kursorning joyini o'zgartirish uchun nimadan foydalaniladi?
 A) Sichqoncha. B) Asboblarni paneli C) Menyu satri
 D) Holat satri

Savollar:

1. Ob'ektga yo'naltirilgan dasturlash atamasi dastlab qachondan boshlab ishlatila boshlagan?
2. Ob'ekt nima?
3. Qanday dasturlash tillarini bilasiz?
4. Ob'ektga yo'naltirilgan dasturlashning nechta xususiyati bor?

10-amaliy mashg'ulot. Dasturlash muhiti bilan tanishish, ilk dasturni taxlil qilish, xatolar, muammoni yechish.

Ishdan maqsad: mavzuga doir nazariy ma'lumotlarni o'rganib, berilgan topshiriqlarni bajarish.

Vazifa.

1. Berilgan nazariy materiallarni o'rganish.
2. Mavzu bo'yicha testlarni bajarish.

1. Berilgan nazariy materiallarni o'rganish.

Dasturlash tillari sun'iy tillar hisoblanadi, ularda sintaksis va semantik qoidalar qat'iy aniqlangan bo'ladi. SHu sababli Dasturlash tillari tabiiy tillardan

farqli ravishda jummalarni ko'pmazmunli va erkin talqin etishga yo'l qo'ymaydi. Buning asosiy sababi tildagi xar bir ko'rsatma mashina tilidagi aniq bir buyruqqa o'tadi.

Sintaksis-bu qoidalar to'plami bo'lib, Dasturlash tilida ruxsat etilgan belgilarning ketma-ketligi va asosiy ichki tuzilishni aniqlaydi. Semantika-til birliklariga (so'z, so'z birikmalari, jumllariga) beriladigan qiymatlardir.

EHMlar endi yuzaga kelgan paytda Dastur tuzishda, faqat mashina tillarida, ya'ni sonlar yordamida EHM bajarishi kerak bo'lgan amallarning kodlarida kiritilgan. Bu holda mashina uchun tushinarli sanoq, sistemasi sifatida 2 lik, 6 lik, 8 lik sanoq sistemalari bo'lgan. Dastur mazkur sanoq sistemasidagi sonlar vositasida kiritilgan.

YUqori bosqichli Dasturlashda, mashina tillariga qaraganda mashinaga moslashgan (yo'naltirilgan) belgili kodlardagi tillar hisoblanadi. Belgilar kodlashtirilgan tillarning asosiy tamoyillari shundaki, unda mashina kodlari ularga mos belgilar bilan belgilanadi, hamda xotirani avtomatik taqsimlash va xatolarni tashhis qilish kiritilgan. Bunday mashina moslashgan til - ASSEMBLER tili nomini oldi.

EHM faqat mashina tilini-buyruqlar, operand va sonlarning ikkilik sanok, sistemasidagi ko'rinishini «tushinadi». SHu sababli, Assembler tilida yozilgan Dasturlar uchun ularni mashina tiliga o'tkazuvchi «tarjimon» kerak bo'ladi. Assembler tilidagi Dasturni mashina tiliga *o'tkazuvchi* maxsus Dastur translyator deyiladi, ayrim hollarda uni assembler deb ham atashadi. Mashinaga moslashgan tillarning asosiy kamchiligi bir turdagi mashina uchun tuzilgan Dastur boshqa mashinalarda bajarilmaydi, ya'ni bunday tillar mashinaning apparat tuzilishiga bog'liq qilib yaratiladi. Masalan, Pravets mashinasi uchun tuzilgan Dastur IBM yoki YAMANA turidagi mashinalar uchun o'tmaydi va aksincha.

Dasturlash tillarining keyingi bosqichida protsedurali tillar joylashadi. Bu tillaring asosiy xususiyati shundaki, ularning sintaksis va semantikasi konkret EHM (protessor) buyruqlari tarkibiga bog'liq, emas. Tuzilgan Dasturni konkret EHM ga bog'lashni translyator amalga oshiradi.

Dasturning boshlang'ich matni Operativ xotiraga kiritilgandan keyin u translyasiya qilinadi. Natijada, aynan shu mashina «tushinadigan» buyruqlar ketma-ketligi hosil bo'ladi va ularni bajarish asosida EHM masalani echadi. Dasturni tarjimasi va uni bajarish jarayoni ikki usulda amalga oshirilishi mumkin: Birinchi usul-komplilyator deb nomlanuvchi usulda Dasturni EHMda bajarilishi-tarjima jaryoni to'liq, tugagandan keyin amalga oshiriladi. Bunda tarjima Dastursini operativ xotirada saqlab turishga hojat yuq, shu sababli xotirani tejashga erishiladi.

Ikkinchi usul – interpretatsiya - Dasturdagi ayrim operatorlar ular tarjima qilingan zahoti bajariladi, shundan keyin navbatdagi operator tarjima qilinadi, bajariladi va xokazo. Bu rejimda xotirada interpretator Dastursi ham bo'lishi kerak, natijada qo'shimcha xotira ishlatiladi va Dasturning ishlashi kompilyasiya qilinganga nisbatan sekin ishlaydi. Interpretatorning afzallik tomonlari Dasturlarni

sozlash paytida ko‘rinadi, ya’ni yo‘l qo‘yilgan xatolar tuzatilishi mumkin. Kompilyatorda bunday imkoniyat yo‘q.

Protsedurali tillarning umumiy xususiyatlarini ko‘raylik. Bu tillarda Dasturdagi yozuvlar umumiy qabul qilingan matematik yozuvlarga juda yaqin va tushunish uchun engildir. Masalan, 7 va 5 sonlaring yig‘indisini hisoblash amalini Dasturlash tillarida yozilishini ko‘raylik:

mashina tilida

Buyruq	Operand adresi
0011 1111	1101 0001
0000 0101	1101 0010
0000 0110	1101 00111
0000 0111	1101 0100
1000 0000	1101 0101

2) mashinaga moslashgan tilda

MOV A, 7

MOV V, 5

ADD V

protsedurali tillarda $A=7+5$.

Ko‘rinib turibdiki, protsedurali tillardagi yozuv tabiiy yozuvga yaqin. Birinchilar qatorida yuzaga kelgan protsedurali tillar ichida FORTRAN (FORMula TRANslation - formulalarni boshqa ko‘rinishga aylantirish). O‘zining sodda tuzilishi, samarali translyasiyalanishi bu tilni nafaqat shu kungacha saqlanib qolishiga sabab bo‘ldi, balki uning ilmiy tadkqdqot, injener-texnik masalalarini echishda eng samarali til sifatida rivojlanib, keng tarqalishiga sabab bo‘ldi. Fortran tili Beysik tilining asosiga qo‘yildi. Beysik so‘zi inglizcha «VASIC-Veginners All-purpose Symbolic Instruction Sode» ma’noni anglatadi. Bu jumla quyidagicha tarjima qilinadi: boshlovchilar uchun ko‘pmaqsadli belgili buyruqlar tili.

Beysik tili mutaxassis bo‘lmagan odamlarni Dasturlash texnologiyalariga jalb qilish imkoniyatini berdi. Beysik tilida Dastur operatorlarining kamligi,

tuzilishining soddaligi uni Dasturlashga o'rgatish tili sifatida keng tarqalishiga sabab bo'ldi. Bu til asosan maktablarda o'rgatiladi.

Algol-60 tili Fortranga nisbatan takomillashgan til, unda tuzilgan Dasturlar moslashuvchanlik va yuqori ishonchlilik xususiyatlari bilan xarakterlanadi. Prolog va Lisp tillari sun'iy tafakkur sohasidagi masalalarni echishga moslashgan. Bu tillar belgili berilganlarni qayta-ishlash orqali matematika va mantiqiy masalalarni yechishga eng qulay tillardir.

1971 yilda paydo bo'lgan Paskal tili Dasturlash texnologiyalariga strukturali Dasturlash g'oyasini olib kirdi. Natijada bitta masala bir nechta bo'laklarga bo'lingan holda alohida yechilishi va oxirida bitta Dasturga jamlanish imkoniyati yuzaga keldi.

Ci tili zamonaviy kompyuterlar xususiyatlarini o'zida aks ettirgan va samarali Dasturlar yaratishda Assemblerga murojaat qilmaydigan til sifatida namoyon bo'ldi. Bu til UNIX operatsion sistemasida yozilgan. Ayni paytda Dasturlash texnologiyasining istiqbolli yo'nalishlaridan biri ob'ektga yo'naltirilgan Dasturlash tillaridir (ob'ektno-orientirovannoe programmirovanie). Ob'ektlar ko'p ishlatiladigan Dastur modullaridir. O'z navbatida ob'ektlar ikkita qismdan tashkil topadi: metodlar va o'zgaruvchilar. Metodlar protsedura va funksiyalar to'plami bo'lib, ob'ekt ishlash algoritmini aniqlaydi. O'zgaruvchilar turli xil qiymatlarni qabul qiladi.

Ob'ektga moslashgan Dasturlashning asosiy mohiyati shundaki, soddal ob'ektlardan vorislik orqali murakkab ob'ektlar shajarasini yaratish mumkin. Bunda biror masalani echish deganda masala algoritimga mos ob'ektlar shajarasini qurish tushiniladi.

Ob'ektga yo'naltirilgan tillarning yaqqol namunasi - bu Delphi vizual Dasturlash muhitidir. Bu muhit Turbo Paskal tili asosida yaratilgan bo'lib, unda Dastur yaratish muhitidagi mavjud komponentalarni loyihalash asosida bo'ladi. SHuning uchun ham bosh Dastur Rgoject deyiladi. Delphining yana bir muhim xususiyati vizual Dasturlashdir, ya'ni Dastur yaratilayotgan paytda tuzuvchi hosil bo'ladigan Dastur shaklini ko'rib turadi va grafik interfeys yordamida komponentalar shaklini o'zi ma'qul topgan joyiga qo'yishi mumkin. Visual Basic tili ham ob'ektga moslashgan til hisobalanadi va Windowsning ofis Dasturlarida makroslar yozish uchun til sifatida ishlatiladi. Internet uchun Dasturlar yaratish uchun NTML (Nureg Text Markur Language) 1989 yilda yaratilgan bo'lib, WWW sahifalarini yaratish uchun asosiy til bo'lib qolmoqda. Bu til yordamida sahifa poligrafik usulda yaratishilishi mumkin, unda matn, audio, video, animatsiya va boshqa ma'lumotlar joylashishi mumkin. Java tili kompyuter tarmoqlarida amal qiladigan Dasturlarni yaratishga mo'ljallangan, ya'ni Java tilida yozilgan Dasturlar global tarmoqlarda dinamik reklamalarni (animatsiya, teletayp lentarlari) global tarmoqda joylashtirish uchun ishlatiladi, Java tilining afzalligi shundaki, u turli operatsion sistemalarda ham birdek amal qilaveradi. Java Dastursi bir paytda Windows, UNIX va Macintosh operatsion sistemalarida ishlashi mumkin.

Quyidagi jadvalda Dasturlash tillari xaqida ma'lumotlar keltirilgan.

Til	YAratilgan yili	Mualliflar	Tashkilot, firma
Ada, Ada	1979-80	Jean Ichbian	Cii-oneywell(Fransiya)
Algol, Algol	1960		International Commitee
ARL, APL	1961-1962	Kenneth Iverson, Adin Falkoff	IBM
DELPHI, DELFI	1995		Borland
VASIS, Beysik	1964-1965	JohnKemeny, Thomas Kurtz	Dartmouth Colleje
C, Ci	1972-1973	Dennis Ritchie	Bell Laboratories
C++, Ci	1980	Bjarne Strostrup	Bell Laboratories
SOVOL, Kobol	1959-1961	Grace Murray Hopper	
Fort	1971	Charles H.Moore	
FORTTRAN, FORTRAN	1950-1958	John Backus	IBM
HTML	1989	Tim Berners-Li	CERN, Jeneva
LISP, LISP	1956-1960	John MCCarthy	
LOGO, LOGO	1968-70	Seymour Papert	Massachusetts Institute of Techn.
Pascal, Paskal	1967-1971	Niklaus Wirth	Federal Institute of Technology (SHveysariya)
PL1, PL1	1964-1966		
PROLOG, PROLOG	1978	Alan Kalmeroe	

SIMULA, SIMULA	1967	Ole-Yoxan Dal, Kristen Nigaard	Norvegiya XM
Java, Djava	1995	Djeys Gosling	Sun Microsystems

2 . Mavzu bo'yicha testlarni bajarish.

1. Menyu satridan so'ng qanday satr joylashgan?

- A) Asboblar paneli B) Sarlavha satri C) Holat satri
D) CHizg'ich

2. CHizg'ichlar qaysi menyu orqali o'rnatiladi va o'chiriladi?

- A) Вид ва Правка B) Правка ва Вид C) Вид ва линейка
D) Файл ва Вид

3. Asboblar panelidagi birinchi tugma qanday tugma?

- A) Saqlash tugmasi B) CHop etish tugmasi C) Yangi fayl hosil qilish tugmasi.
D) SHrift turi

4. Qanday dasturlash tillarini bilasiz?

- A) Word, Excel, Paint B) Delphi , Paskal, C++ C) Windows, NC
D) Matlab, Mathcad, Avtacad

5. Windows operatsion tizimining asosiy menyusiga qanday chiqiladi?

- A) «Мой компьютер » papkasidan; B) «Пуск» tugmachasidan;
C) «Мои документы» papkasidan; D) Masalalar panelidan

6. Skanerning vazifasi nima?

- A) qog'ozdagi ma'lumotlarni kompyuterga tasviriy ko'chirish.
B) Qog'ozdagi matnlarni chop qilish.
C) Diskdagi fayllarni kompyuterga o'tkazish.
D) Ma'lumotlarni magnit lentalariga yozish.

7. Qaysi printerda tasvir nuqtalardan zarbli usul bilan shakllanadi?

- A) Matrisali printer B) purkagichli printer C) lazerli printer

8. Qanday chop qilish qurilmalari bor?

- A) Rangli, planshetli, purkagichli B) Matrisali printer , purkagichli printer ,
lazerli printer
C) oq-qora, rangli D) matrisali

9. Birinchi avlod EHMLarning asosiy element bazasini nimalar tashkil etadi.

- A) Elektron lampalar B) yarim o'tkazgichli tranzistorlar
C) chiplar D) integral sxemalar

10. Monitorlar necha xil ko'rinishda bo'ladi?

- A) 5 B) 3 C) 4 D) 2

11-amaliy mashg'ulot: C++ algoritmik tili va unda dastur yaratish.

Ishdan maqsad: C++ algoritmik tilida sodda dasturlar tuzish.

Vazifa.

- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Mavzu bo'yicha sodda dasturlar tuzish.

1 . Berilgan nazariy materiallarni o'rganish.

Ma'lumotlarning haqiqiy son turi. Ma'lumotlarning haqiqiy son turi olti qismdan iborat bo'lishi mumkin: butun qism, nuqta, kasr qism, e yoki E belgisi, o'nlik daraja, F yoki f suffikslari.

Masalan : 66. .0 .12 3.14F 1.12e-12

Ma'lumolarning uzun xaqiqiy son turi :

Oxiriga L yoki l suffikslari qo'yilgan xaqiqiy son.

Masalan: 2E+6L;

Mantiqiy konstanta. Mantiqiy konstantalar true(rost) va false(yolg'on) qiymatlardan iborat. Ichki ko'rinishi false – 0, ixtiyoriy boshqa qiymat true deb qaraladi.

Satrlı konstanta. Satrlı konstantalar C++ tili konstantalariga kirmaydi, balki leksemalari aloxida tipi hisoblanadi. SHuning uchun adabiyotda satrlı konstantalar satrlı leksemalar deb ham ataladi.

Satrlı konstanta bu ikkilik qavslarga olingan ixtiyoriy simvollar ketma ketligidir. Misol uchun "Men satrlı konstantaman".

Satrlar orasiga eskeyp simvollar ham kirishi mumkin. Bu simvollar oldiga \ belgisi qo'yiladi. Misol uchun:

"\n Bu satr \n uch qatorga \n joylashadi".

Satr simvolları xotirada ketma ket joylashtiriladi va har bir satrlı konstanta oxiriga avtomatik ravishda kompilyator tomonidan '\0' simvoli qo'shiladi. SHunday satrning xotiradagi xajmi simvollar sonı+1 baytga tengdir.

Ketma-ket kelgan va bo'shliq, tabulyatsiya yoki satr oxiri (" ") belgisi bilan ajratilgan satrlar kompilyatsiya davrida bitta satrga aylantiriladi. Misol uchun:

"Salom" "Toshkent "

satrlari bitta satr deb qaraladi.

"Salom Toshkent"

Bu qoidaga bir necha qatorga yozilgan satrlar ham bo'ysinadi. Misol uchun :

"O'zbekistonga "

"baxor "

"keldi"

qatorlari bitta qatorga mos:

"O'zbekistonga baxor keldi"

Agar satrda '\ ' belgisi uchrasa va bu belgidan so'ng to '\n' satr oxiri belgisigacha bo'shlik belgisi kelsa bu bo'shlik belgilari '\ ' va '\n' belgisi bilan birga satrdan o'chiriladi. Satrning o'zi keyingi satrda kelgan satr bilan qo'shiladi.

Nomlangan o'zgarmaslar. Belgili o'zgarmas – bu nomga ega bo'lgan o'zgarmasdir. C++ tilida belgili o'zgarmasni aniqlashning ikki usuli mavjud:

#define direktivasi yordamida o'zgarmasni aniqlash.

const kalitli so'zi orqali o'zgarmasni aniqlash.

An'anaviy usul hisoblangan #define direktivasi orqali o'zgarmasni aniqlashni quyidagi misolda ko'rishimiz mumkin.

```
#define StudentsPerClass 15
```

Bu holda StudentsPerClass o'zgarmas hech qanday tipga tegishli bo'lmaydi. Preprocessor StudentsPerClass so'ziga duch kelganida uni 15 literaliga almashtiradi.

C++ tilida #define direktivasidan tashqari o'zgarmasni aniqlashning nisbatan qulayroq bo'lgan yangi usuli ham mavjud:

```
const unsigned short int StudentsPerClass=15
```

Bu misolda ham belgili konstanta StudentsPerClass nomi bilan aniqlanayapti va unga unsigned short int tipi berilyapti. Bu usul bir qancha imkoniyatlarga ega bo'lib u sizning dasturingizni keyingi himoyasini yengillashtiradi. Bu o'zgarmasni oldingisidan eng muhim afzalligi uning tipga egaligidir.

Bu konstantalar qiymatlarini dasturda o'zgartirish mumkin emas. Konstantalar nomlari dasturchi tomonidan kiritilgan va xizmatchi so'zlardan farqli bo'lgan identifikatorlar bo'lishi mumkin. Odatda nom sifatida katta lotin harflari va ostiga chizish belgilari kombinatsiyasidan iborat identifikatorlar ishlatiladi. Nomlangan konstantalar quyidagi shaklda kiritiladi:

```
sonst tip konstanta_nomi=konstanta_qiymati.
```

Misol uchun:

```
const double EULER=2.718282;
```

```
const long M=999999999;
```

```
const R=765;
```

Sanovchi konstantalar enum xizmatchi so'zi yordamida kiritilib, int tipidagi sonlarga qulay so'zlarni mos qo'yish uchun ishlatiladi.

Misol uchun:

```
enum{one=1,two=2,three=3};
```

Agar son qiymatlari ko'rsatilmagan bo'lsa eng chapki so'zga 0 qiymati berilib kolganlariga tartib bo'yicha o'suvchi sonlar mos qo'yiladi:

```
enum{zero,one,two};
```

Bu misolda avtomatik ravishda konstantalar quyidagi qiymatlarni qabul qiladi:

```
zero=0, one=1, two=2;
```

Konstantalar aralash ko'rinishda kiritilishi ham mumkin:

```
enum(zero,one,for=4,five,seeks}.
```

Bu misolda avtomatik ravishda konstantalar quyidagi qiymatlarni qabul qiladi:

```
zero=0, one=1, for=4,five=5,seeks=6;
```

Sanoqli tiplarni hosil qilish uchun enum kalitli so'zi va undan keyin tip nomi hamda figurali qavs ichida vergullar bilan ajratilgan o'zgarmas qiymatlari ro'yxati ishlatiladi. Masalan, RANG nomli sanoqli tip deb e'lon qilaylik va uning uchun 5 ta QIZIL, KO'K, YASHIL, OQ, QORA qiymatlarini aniqlaylik.

```
enum RANG { QIZIL, KO'K, YASHIL, OQ, QORA };
```

Bunda ifoda ikkita ishni bajaradi:

RANG nomli yangi sanoqli tip hosil qiladi;

Quyidagi belgisi o'zgarishlarni aniqlaydi.

0 qiymat bilan QIZIL;

1 qiymat bilan KO'K;

2 qiymat bilan YASHIL va hokazo;

Matematik amallar. C++ tilida 5 ta asosiy matematik amallar qo'llaniladi: qo'shish (+), ayirish (-), ko'paytirish (*), butun songa bo'lish (/) va modul bo'yicha bo'lish (%) (qoldiqni olish). Amallar odatda unar ya'ni bitta operandga qo'llaniladigan amallarga va binar ya'ni ikki operandga qo'llaniladigan amallarga ajratiladi.

Binar amallar additiv ya'ni + qo'shish va – ayirish amallariga, hamda multiplikativ ya'ni * ko'paytirish, / bo'lish va % modul olish amallariga ajratiladi.

Butun songa bo'lish odatdagi bo'lishdan farq qiladi. Butun songa bo'lishdan hosil bo'lgan bo'linmaning faqatgina butun qismi olinadi.

Butun sonni butun songa bo'lganda natija butun songacha yaxlitlanadi. Misol uchun $20/3=6$; $(-20)/3=-6$; $20/(-3)=-6$.

Modul amali butun sonni butun songa bo'lishdan xosil bo'ladigan qoldiqqa tengdir. Agar modul amali musbat operandlarga qo'llanilsa, natija ham musbat bo'ladi, aks holda natija ishorasi kompilyatorga bog'liqdir.

Masalan, 21 sonini 4 ga bo'lsak 5 soni va 1 qoldiq hosil bo'ladi. 5 butun songa bo'lishni qiymati, 1 esa qoldiqni olish qiymati hisoblanadi.

Inkrement va dekrement. Dasturlarda o'zgaruvchiga 1 ni qo'shish va ayirish amallari juda ko'p hollarda uchraydi. C++ tilida qiymatni 1 ga oshirish inkrement, 1 ga kamaytirish esa dekrement deyiladi. Bu amallar uchun maxsus operatorlar mavjuddir.

Inkrement operatori (++) o'zgaruvchi qiymatini 1 ga oshiradi, dekrement operatori (—) esa o'zgaruvchi qiymatini 1 ga kamaytiradi. Masalan, s o'zgaruvchisiga 1 qiymatni qo'shmoqchi bo'lsak quyidagi ifodani yozishimiz lozim.

C++ //s o'zgaruvchi qiymatini 1 ga oshirdik.

Bu ifodani quyidagicha yozishimiz mumkin edi.

$s=s+1$;

Bu ifoda o'z navbatida quyidagi ifodaga teng kuchli:

$s+=1$;

Amallar ustivorligi. Murakkab ifodalarda qaysi amal birinchi navbatda bajariladi, qo'shishmi yoki ko'paytirishmi? Masalan:

$x=5+3*8$;

ifodada agarda birinchi qo'shish bajarilsa natija 64 ga, agarda ko'paytirish birinchi bajarilsa natija 29 ga teng bo'ladi.

Murakkab qiymat berish amali. C++ tilida murakkab qiymat berish amali mavjud bo'lib, umumiy ko'rinishi quyidagichadir:

O'zgaruvchi_nomi amal= ifoda;

Bu yerda amal quyidagi amallardan biri $*$, $/$, $\%$, $+$, $-$, $\&$, $^$, $|$, $<<$, $>>$.

Misol uchun:

$x+=4$ ifoda $x=x+4$ ifodaga ekvivalentdir;

$x*=a$ ifoda $x=x*a$ ifodaga ekvivalentdir;

$x/=a+b$ ifoda $x=x/(a+b)$ ifodaga ekvivalentdir;

$x>>=4$ ifoda $x=x>>4$ ifodaga ekvivalentdir;

Imlo belgilari amal sifatida. C++ tilida ba'zi bir imlo belgilari ham amal sifatida ishlatilishi mumkin. Bu belgilardan oddiy () va kvadrat [] qavslardir. Oddiy qavslar binar amal deb qaralib ifodalarda yoki funktsiyaga murojaat qilishda foydalaniladi. Funktsiyaga murojaat qilish quyidagi shaklda amlga oshiriladi:

<funktsiya nomi> (<argumentlar ro'yxati>). Misol uchun $\sin(x)$ yoki $\max(a,b)$.

Murakkab ifodalarni tuzishda ichki qavslardan foydalaniladi.

Quyidagi mantiq asosida yuqori darajada tashkil qilingan signalizatsiya sistemasini tasavvur qiling. Agarda eshikda signalizatsiya o'rnatilgan bo'lsa VA kun vaqti kech soat olti VA bugun bayram YoKI dam olish kuni BO'LMASA politsiya chaqirilsin. Barcha shartlarni tekshirish uchun C++ tilining uchta mantiqiy operatori ishlatiladi. Ular jadvalda keltirilgan

Amal	Belgi	Misol
VA	&&	1ifoda && 2ifoda
YoKI		1ifoda 2ifoda
INKOR	!	!ifoda

2 . Mavzu bo'yicha sodda dasturlar tuzish.

1-misol. Nolga teng bo'lmagan ikkita son berilgan. Ularning yig'indisini, ko'paytmasini va har birining kvadratini hisoblovchi dastur tuzing.

1-misol uchun dastur ko'rinishi:

```
#include <iostream.h>
int main()
{
    int a,b,d,f,k,s;
    cout <<"a="; cin >>a;
    cout <<"b="; cin >>b;
    d=a+b;
    f=a*b;
    k=a*a;
```

```

s=b*b;
cout << d << endl;
cout << f << endl;
cout << k << endl;
cout << s << endl;
return (0);
}

```

2-misol. 10 dan 1 gacha bo'lgan sonlarni chiqaruvchi dastur tuzing.

2-misol uchun dastur ko'rinishi:

```

#include <iostream.h>
int main()
{
    for (int i = 10; i >= 1; i--)
        cout << i << " ";
    cout << endl;

    return (0);
}

```

3-misol. 2 ta sondan kattasini topuvchi dastur tuzing.

3-misol uchun dastur ko'rinishi:

```

#include <iostream.h>
int main()
{
    int a,b,max;
    cout << "a="; cin >> a;
    cout << "b="; cin >> b;
    max=(a>b) ? a:b;
    cout << max << endl;

    return (0);
}

```

Topshiriq: berilgan masalalar uchun sodda dasturlar tuzing.

1. Tomonlari a va b bo'lgan to'g'ri to'rtburchakning diagonalini topuvchi dastur tuzing. $d^2 = a^2 + b^2$
2. Aylananing radiusi r berilgan. Uning uzunligini hisoblovchi dastur tuzing:
 $L = 2\pi r$, $\pi = 3,14$
3. x –ning qiymati berilganda $y = 3x^6 - 6x^2 - 7$ funksiyaning qiymatini aniqlovchi dastur tuzing.
4. Temperatura T_F Faringeytda berilgan. Temperatura qiymatini T_C gradus selsiyga o'tkazuvchi dastur tuzing. $T_C = (T_F - 32) * 5/9$

5. Tomoni va balandligi berilgan rombning yuzini topuvchi dastur tuzing. $S = ah$
6. Asoslari berilgan trapetsiyaning o'rta chizig'ini topuvchi dasturni tuzing. $l = \frac{a+b}{2}$
7. Sonlar o'qida ikki nuqta orasidagi masofani aniqlovchi dastur tuzing.
8. 1 dan 15 gacha bo'lgan sonlarni chiqaruvchi dastur tuzing.
9. 2 ta sondan kattasini topuvchi dastur tuzing.
10. Temperatura T_C gradus selsiyda berilgan. Temperatura qiymatini T_F
11. Faringeytga o'tkazuvchi dastur tuzing. $T_F = (T_C - 32) * 5/9$

12-amaliy mashg'ulot. C++ algoritmik tilning tarmoqlanish operatorlar.

Ishdan maqsad: C++ algoritmik tilning tarmoqlanish operatorlari bilan ishlash.

Vazifa.

1. Berilgan nazariy materiallarni o'rganish.
2. Mavzu bo'yicha sodda dasturlar tuzish.

1. Berilgan nazariy materiallarni o'rganish.

If va if else operatorlari, ?: amali.

Dastur bajarilishining birorta qadamida qandaydir shartni tekshirish natijasiga ko'ra boshqaruvni dasturning u yoki bu bo'lagiga uzatish mumkin (Tarmoqlanuvchi algoritm). Tarmoqlanishni amalga oshirish uchun shartli operatorlardan foydalaniladi.

If operatori. *If* qandaydir shartni rostlikka tekshirish natijasiga ko'ra dasturda tarmoqlanishni amalga oshiradi:

If (<shart>) <operator>;

Bu yerda *<shart>* har qanday ifoda bo'lishi mumkin, odatda u taqqoslash amali bo'ladi.

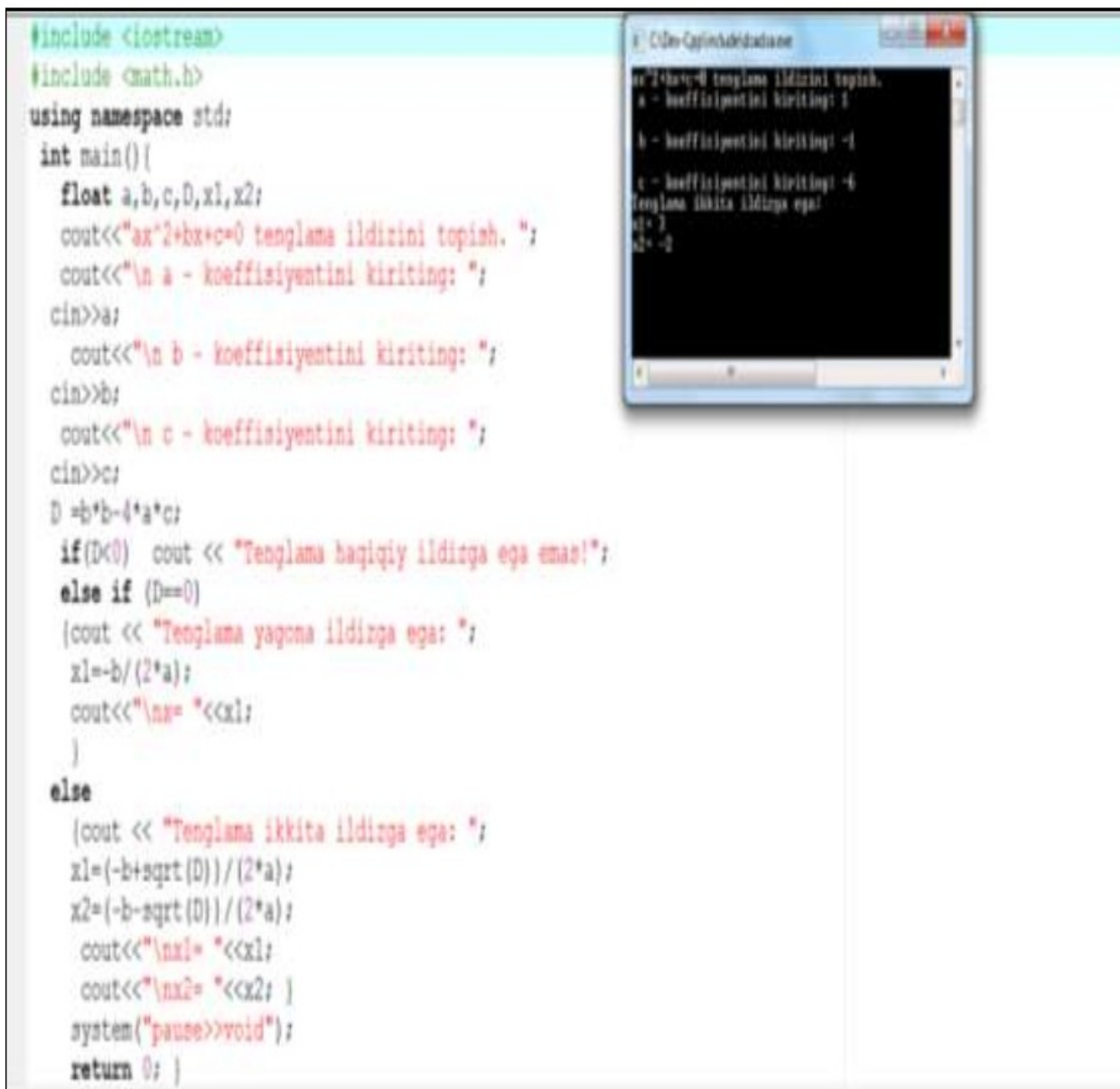
Agar shart 0 qiymatidan farqli yoki rost (true) bo'lsa, *<operator>* bajariladi, aks holda, ya'ni shart 0 yoki yolg'on (false) bo'lsa, hech qanday amal bajarilmaydi va boshqaruv *if* operatoridan keyingi operatorga o'tadi (agar u mavjud bo'lsa).

C++ tilining qurilmalari operatorlarni blok ko'rinishida tashkil qilishga imkon beradi. Blok - '{' va '}' belgi oralig'iga olingan operatorlar ketma-ketligi bo'lib, u kompilyator tomonidan yaxlit bir operator deb qabul qilinadi. Blok ichida e'lon operatorlari ham bo'lishi mumkin va ularda e'lon qilingan o'zgaruvchilar faqat shu blok ichida ko'rinadi (amal qiladi), blokdan tashqarida ko'rinmaydi. Blokdan keyin ';' belgisi qo'yilmasligi mumkin, lekin blok ichidagi har bir ifoda ';' belgisi bilan yakunlanishi shart. Shart operatorining **if - else** ko'rinishi quyidagicha:

if (<shart-ifoda>) <operator1>; else <operator2>;

Bu yerda *<shart - ifoda>* 0 qiymatidan farqli yoki true bo'lsa, *<operator1>*, aks holda *<operator2>* bajariladi. Misol tariqasida diskreminantni

hisoblash usuli yordamida $ax^2 + bx + c = 0$ ko'rinishidagi kvadrat tenglama ildizlarini topish masalasini ko'raylik:



12.1-rasm.Kvadrat tenglamani yechish dasturi

Dastur bajarilganda, birinchi navbatda tenglama koeffitsiyentlari - a,b,c o'zgaruvchilar qiymatlari kiritiladi, keyin diskriminant – D o'zgaruvchi qiymati hisoblanadi. Keyin D qiymatining manfiy ekanligi tekshiriladi. Agar shart o'rinli bo'lsa, yaxlit operator sifatida keluvchi '{' va '}' belgilari orasidagi operatorlar bajariladi va ekranga "Tenglama haqiqiy ildizlarga ega emas" xabari chiqadi va dastur o'z ishini tugatadi ("return 0;" operatorini bajarish orqali). Diskriminant noldan kichik bo'lmasa, navbatdagi shart operatori uni nolga tengligini tekshiradi. Agar shart o'rinli bo'lsa, keyingi qatorlardagi operatorlar bloki bajariladi - ekranga "Tenglama yagona ildizga ega:" xabari, hamda x1 o'zgaruvchi qiymati chop etiladi va dastur shu yerda o'z ishini tugatadi, aks holda, ya'ni D qiymati noldan katta holati uchun else kalit so'zidan keyingi operatorlar bloki bajariladi va ekranga "Tenglama ikkita ildizga ega: " xabari, hamda x1 va x2

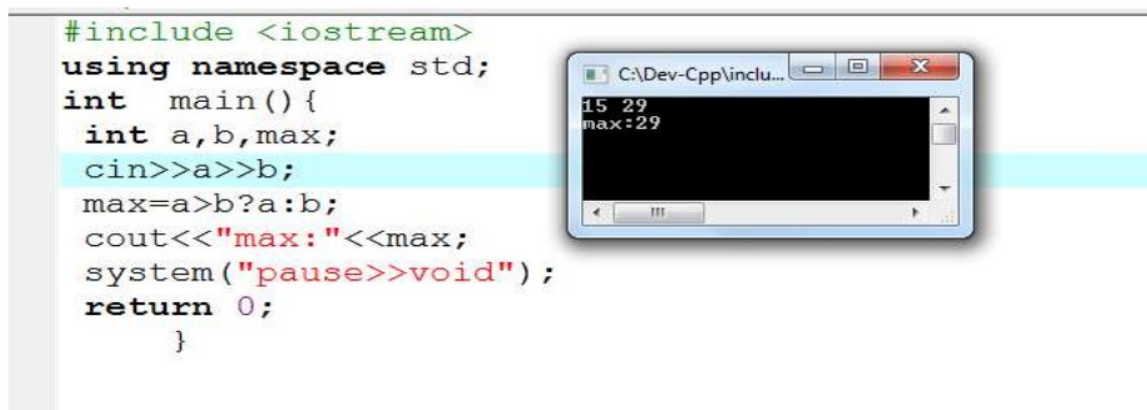
o'zgaruvchilar qiymatlari chop etiladi. Shu bilan shart operatoridan chiqiladi va asosiy funksiyaning return ko'rsatmasini bajarish orqali dastur o'z ishini tugatadi.

?: shart amali. Agar tekshirilayotgan shart nisbatan sodda bo'lsa, shart amalining “?:” ko'rinishini ishlatish mumkin:

$\langle \text{shart ifoda} \rangle ? \langle \text{ifoda1} \rangle : \langle \text{ifoda2} \rangle ;$

Shart amali *if* shart operatoriga o'xshash holda ishlaydi: agar $\langle \text{shart ifoda} \rangle$ 0 qiymatidan farqli yoki *true* bo'lsa, $\langle \text{ifoda1} \rangle$, holda $\langle \text{ifoda2} \rangle$ bajariladi. Odatda ifodalar qiymatlari birorta o'zgaruvchiga o'zlashtiriladi.

Misol:



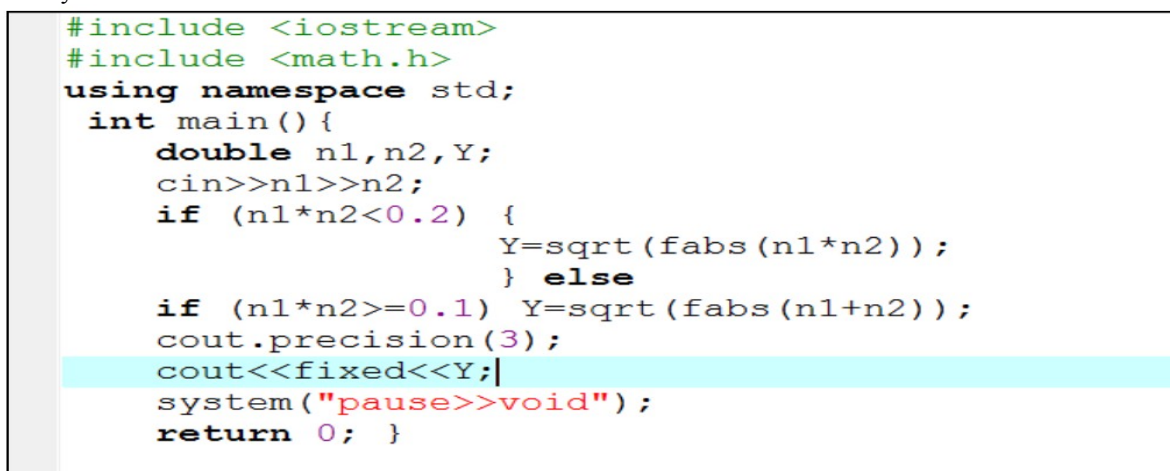
12.2-rasm

Berilgan masala uchun algoritm.

Yuqorida berilgan masalani hisoblash uchun birinchi navbatda uni matematik ko'rinishdan C++ dasturlash muhitiga o'tkazishda *if* shart operatoridan foydalanamiz.

$$Y = \begin{cases} \sqrt{|n_1 \cdot n_2|} & \text{agar } n_1 n_2 < 0.2 \\ \sqrt{|n_1 + n_2|} & \text{agar } n_1 n_2 \geq 0.1 \end{cases}$$

Asosiy dastur matni:

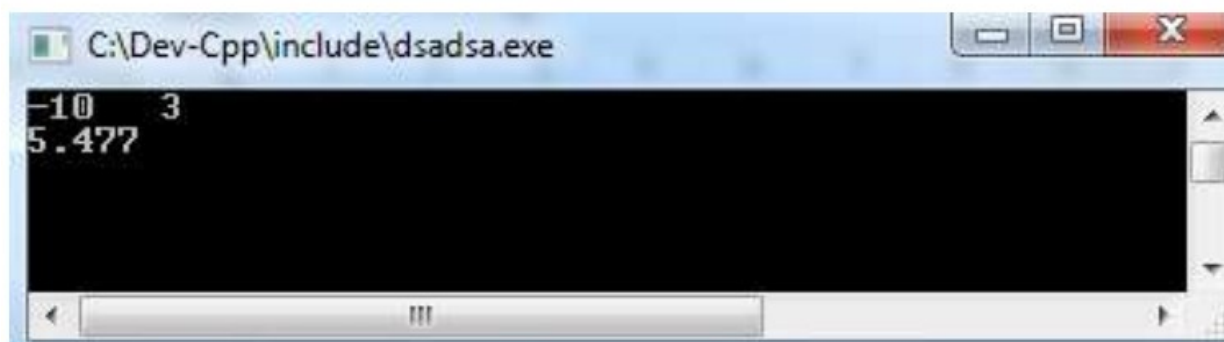


12.3-rasm

Berilgan misol ishlanish natijalari tahlili.

n_1 va n_2 haqiqiy sonlari kiritilgandan so'ng ularning ko'paytmasi *if* shart operatori yordamida tekshirilib agarda $n_1 n_2 < 0.2$ shart bajarilsa $Y = \sqrt{|n_1 \cdot n_2|}$ qiymat, aks holda, $n_1 n_2 \geq 0.1$ shart bajarilsa $Y = \sqrt{|n_1 + n_2|}$ qiymatni o'zlashtiradi va natijaga 10^{-3} aniqlikda chiqaradi.

Natija.



12.4-rasm

C++ tilida oddiy dastur. C++ tilida tuzilgan dastur ob'ektlar, funksiyalar, o'zgaruvchilar va boshqa elementlardan tashkil topadi.

Topshiriqlar: Quyidagi variantlarda berilgan tarmoqlanuvchi algoritmlar dasturini tuzing.

N _o	Variantlar	N _o	Kirish	Chiqish
1	$Y = \begin{cases} m^2n + 1 - c, & \text{agar } n + 1 > 0 \\ (m + n)^2 + cm^2, & \text{agar } n + 1 \leq 0 \end{cases}$	1	$m=1$ $n=2$ $c=3$	$Y=0.000$
		2	$m=2$ $n=3$ $c=7$	$Y=6.000$
2	$Y = \begin{cases} \frac{1}{\sqrt{9+x^2}}, & \text{agar } a < 5 \\ b \cdot \sin a, & \text{agar } a \geq 5 \end{cases} \text{ bu yerda } a = d^2 + \frac{cd}{c^2-d^2};$	1	$b=1$ $c=2$ $d=3$ $x=4$	$Y=0.999$
		2	$b=2$ $c=3$ $d=7$ $x=-6$	$Y=-1.952$
3	$Z = \begin{cases} 7x^2 - 3ab - 5ab, & \text{agar } a \geq 0; \\ 15a - 7b, & \text{agar } a < 0 \end{cases};$	1	$a=1$ $b=2$ $x=3$	$Z=47.000$
		2	$a=-2$ $b=9$ $x=7$	$Z=-93.000$
4	$Y = \begin{cases} \frac{a^2+b^2}{c} + \sqrt{a^2+x}, & \text{agar } x \geq 0 \\ \frac{\sin x + b}{a-b}, & \text{agar } x < 0 \end{cases} \text{ bu yerda } x = \frac{a^2-b^2}{c^2-b};$	1	$a=1$ $b=2$ $c=3$	$Y=-1.090$
		2	$a=2$ $b=3$ $c=7$	$Y=-2.170$
5	$Y = \begin{cases} (nm^2 + d)^2, & \text{agar } d > 1 \\ \frac{d}{n^2+m^2}, & \text{agar } d \leq 1 \end{cases};$	1	$m=1$ $n=2$ $d=3$	$Y=25.000$
		2	$m=4$ $n=7$ $d=0.2$	$Y=0.003$
6	$Z = \begin{cases} \frac{ax^2}{b-1}, & \text{agar } a < 9 \\ (a+1)^3 + cx^3, & \text{agar } a \geq 9 \end{cases};$	1	$a=1$ $b=2$ $c=3$ $x=4$	$Z=16.000$
		2	$a=12$ $b=3$	$Z=7300.000$

			$c=7$ $x=9$	
7	$X = \begin{cases} \frac{a^3}{3+a}, \text{ agar } a > 0 \\ \sqrt{\left \frac{a^2+2}{1+a}\right }, \text{ agar } a \leq 0 \end{cases}$ bunda $a = \frac{b^2-c^2}{d}$;	1	$b=4$ $c=-2$ $d=8$	$X=0.750$
		2	$b=4$ $c=-3$ $d=7$	$X=0.250$
8	$Z = \begin{cases} \sqrt{\left \frac{x}{x+1}\right } - \sqrt{x}, \text{ agar } x > 3 \\ \left(\frac{\ln x }{x}\right)^3, \text{ agar } x \leq 3 \end{cases}$	1	$x=2$	$Z=0.042$
		2	$x=3.7$	$Z=1.066$
9	$K = \begin{cases} xe^x \sin x, \text{ agar } x \geq 0 \\ \frac{1}{3} \ln^3 x , \text{ agar } x < 0 \end{cases}$;	1	$x=3$	$K=8.503$
		2	$x=-7$	$K=2.456$
10	$Y = \begin{cases} \ln(x + \sqrt{x^2+9}), \text{ agar } x \geq 0 \\ tg^3 x, \text{ agar } x < 0 \end{cases}$;	1	$x=3$	$Y=1.980$
		2	$x=-7.2$	$Y=0.308$
11	$Y = \begin{cases} \frac{a}{a^2+x}, \text{ agar } a > 5 \\ \frac{1}{2a^3+\sin a}, \text{ agar } a \leq 5 \end{cases}$;	1	$a=4$ $x=9$	$Y=0.008$
		2	$a=7$ $x=1$	$Y=0.140$
12	$Y = \begin{cases} \sqrt{k}, \text{ agar } \sin k \leq 0.2 \\ \frac{1}{\sqrt{k}}, \text{ agar } \sin k > 0.2 \end{cases}$	1	$k=7$	$Y=0.378$
		2	$k=1.4$	$Y=0.845$
13	$Y = \begin{cases} \sin^2(2x) - \cos^2 x, \text{ agar } x > 0 \\ \frac{1}{x^2-\sqrt{x}}, \text{ agar } x \leq 0.3 \end{cases}$;	1	$x=0.5$	$Y=-0.062$
		2	$x=0.7$	$Y=0.386$
14	$P = \begin{cases} \frac{1}{2}(3x^2 - 1), \text{ agar } x > 0.4 \\ \frac{1}{2}(5x^3 - 3x), \text{ agar } x \leq 0.4 \end{cases}$;	1	$x=7.3$	$P=79.435$
		2	$x=0.12$	$P=-0.176$
15	$Y = \begin{cases} x^2 + 4, \text{ agar } x < 10 \\ x^3 - 7, \text{ agar } x \geq 10 \end{cases}$ bunda $x = \frac{a^2-b}{c}$;	1	$a=1$ $b=2$ $c=3$	$Y=4.11$
		2	$a=-2$ $b=3$ $c=7$	$Y=4.02$
16	$Y = \begin{cases} \sqrt{15a^2 + 21b^2}, \text{ agar } a > b \\ \sqrt{15b^2 + 21a^2}, \text{ agar } a \leq b \end{cases}$;	1	$a=7$ $b=3$	$Y=30.397$
		2	$a=1.2$ $b=3.8$	$Y=15.711$
17	$Y = \begin{cases} \ln 2x - 3z^2 , \text{ agar } x < 5z \\ \ln 2x^2 - 3z , \text{ agar } x > 5z \end{cases}$;	1	$x=-10$ $z=3$	$Y=3.850$

		2	$x=6.3$ $z=-7$	$Y=4.609$
18	$P = \begin{cases} \sin(5k + 3m), & \text{agar } k > m \\ \cos(5k + 3m), & \text{agar } k \leq m \end{cases}$	1	$k=5$ $m=-6$	$Y=0.754$
		2	$k=-4.2$ $m=3.7$	$Y=0.458$
19	$Y = \begin{cases} \sqrt{2k_1 - 7k_2}, & \text{agar } k_1 k_2 < 1 \\ \sqrt{2k_1 + 7k_2}, & \text{agar } k_1 k_2 \geq 1 \end{cases}$	1	$k_1 = 0.02$ $k_2 = 4$	$Y=5.288$
		2	$k_1 = 9$ $k_2 = 7$	$Y=8.185$
20	$Y = \begin{cases} \frac{4r+3m}{r^2+m^2}, & \text{agar } r > m + \frac{1}{2} \\ r-m , & \text{agar } r \leq m + \frac{1}{2} \end{cases}$	1	$m=3$ $r=5$	$Y=0.853$
		2	$m=6.5$ $r=7$	$Y=0.500$

13-amaliy mashg'ulot. C++ algoritmik tilning takrorlanish operatorlar.

Ishdan maqsad: C++ algoritmik tilning takrorlanish operatorlari bilan ishlash.

Vazifa.

- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Mavzu bo'yicha sodda dasturlar tuzish.

1 . Berilgan nazariy materiallarni o'rganish.

Sikllarni tashkil etish. Har qanday dasturning strukturasi tarmoqlanish va sikllar to'plamining kombinatsiyasidan iborat bo'ladi. Qator masalalarni yechish uchun ko'pincha bitta amalni bir necha marotaba bajarish talab qilinadi. Amaliyotda bu rekursiyalar va iterativ algoritmlar yordamida amalga oshiriladi. Iterativ jarayonlar – bu operatsiyalar ketma-ketligini zaruriy sonda takrorlanishidir.

while operatori orqali sikllarni tashkil etish. while operatori yordamida sikllarni tashkil etishda operatsiyalar ketma-ketligi siklning davom etish sharti «to'g'ri» bo'lsa yoki 0 dan farqli bo'lsagina uning navbatdagi operatsiyalari amalga oshiriladi.

while operatori quyidagi umumiy ko'rinishga egadir:

while(ifoda) Operator

Agar dasturda while (1); satr qo'yilsa bu dastur xech qachon tugamaydi.

For, while va **do while** takrorlash operatorlari, **break, continue, goto** operatorlari.

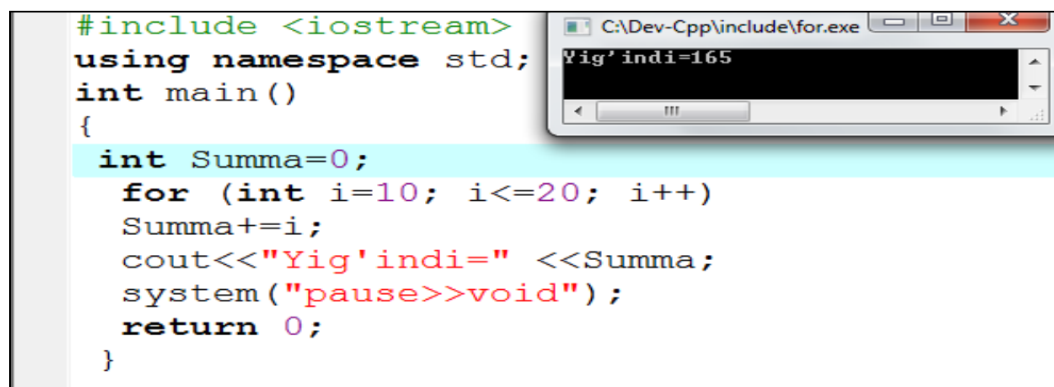
Dastur bajarilishini boshqarishning boshqa bir kuchli mexanizmlaridan biri - takrorlash operatorlari hisoblanadi. Takrorlash operatori «takrorlash sharti» deb nomlanuvchi ifodaning rost qiymatida dasturning ma'lum bir qismidagi operatorlarni (takrorlash tanasini) ko'p marta takror ravishda bajaradi (itarativ jarayon). Takrorlash shartini tekshirish takrorlash tanasidagi operatorlarni bajarishdan oldin tekshirilishi mumkin (**for, while** takrorlashlari) yoki

takrorlash tanasidagi operatorlari bir marta bajarilgandan keyin tekshirilishi mumkin (*do-while*). Takrorlash operatorlari ichma-ich joylashgan bo'lishi mumkin.

for takrorlash operatori

for takrorlash operatorining sintaksisi qo'yidagi ko'rinishga ega:
for (<ifoda1>; <ifoda2>;<ifoda3>) <operator yoki blok>;
Bu operator o'z ishini <ifoda1> ifodasini bajarishdan boshlaydi. Keyin takrorlash qadamlari boshlanadi. Har bir qadamda <ifoda2> bajariladi, agar natija 0 qiymatidan farqli yoki *true* bo'lsa, takrorlash tanasi - <operator yoki blok> bajariladi va oxirida <ifoda3> bajariladi. Agar <ifoda2> qiymati 0 (*false*) bo'lsa, takrorlash jarayoni to'xtaydi va boshqaruv takrorlash operatoridan keyingi operatorga o'tadi. Shuni qayd qilish kerakki, <ifoda2> ifodasi vergul bilan ajratilgan bir nechta ifodalar birlashmasidan iborat bo'lishi mumkin, bu holda oxirgi ifoda qiymati takrorlash sharti hisoblanadi. Takrorlash tanasi sifatida bitta operator, jumladan bo'sh operator bo'lishi yoki operatorlar bloki kelishi mumkin.

Misol uchun 10 dan 20 gacha bo'lgan butun sonlar yig'indisini hisoblash masalasini ko'raylik.



```
#include <iostream>
using namespace std;
int main()
{
    int Summa=0;
    for (int i=10; i<=20; i++)
        Summa+=i;
    cout<<"Yig'indi=" <<Summa;
    system("pause">>void");
    return 0;
}
```

13.1-rasm. 10 dan 20 gacha bo'lgan butun sonlar yig'indisini hisoblovchi dastur tuzilishi

Dasturdagi takrorlash operatori o'z ishini i takrorlash parametriga (takrorlash sanagichiga) boshlang'ich qiymat 10 sonini berishdan boshlaydi va har bir takrorlash qadamidan (itirasiyadan) keyin qavs ichidagi uchinchi operator bajarilishi hisobiga uning qiymati bittaga oshadi. Har bir takrorlash qadamida takrorlash tanasidagi operator bajariladi, ya'ni Summa o'zgaruvchisiga i qiymati qo'shiladi. Takrorlash sanagichi i qiymati 21 bo'lganda "*i<=20*" takrorlash sharti *false* (0-qiymati) bo'ladi va takrorlash tugaydi. Natijada boshqaruv takrorlash operatoridan keyingi *cout* operatoriga o'tadi va ekranga yig'indi chop etiladi. Yuqorida keltirilgan misolga qarab takrorlash operatorlarining qavs ichidagi ifodalariga izoh berish mumkin:

<ifoda1> - takrorlash sanagichi vazifasini bajaruvchi o'zgaruvchiga boshlang'ich qiymat berishga xizmat qiladi va u takrorlash jarayoni boshida faqat bir marta hisoblanadi. Ifodada o'zgaruvchi e'loni uchralishi mumkin va bu o'zgaruvchi takrorlash operatori tanasida amal qiladi va takrorlash operatoridan

tashqarida «ko'rinmaydi».

<ifoda2> - takrorlashni bajarish yoki yo'qligini aniqlab beruvchi mantiqiy ifoda, agar shart rost bo'lsa, takrorlash davom etadi, aks holda yo'q. Agar bu ifoda bo'sh bo'lsa, shart doimo rost deb hisoblanadi;

<ifoda3> - odatda takrorlash sanagichining qiymatini oshirish (kamaytirish) uchun xizmat qiladi yoki unda takrorlash shartiga ta'sir qiluvchi boshqa amallar bo'lishi mumkin. Takrorlash operatorida qavs ichidagi ifodalar bo'lmasligi mumkin, lekin sintaksis ';' bo'lmasligiga ruxsat bermaydi. Shu sababli, eng sodda ko'rinishdagi takrorlash operatori quyidagicha bo'ladi:

for (;);cout <<"Cheksiz takrorlash...";

Agar takrorlash jarayonida bir nechta o'zgaruvchilarning qiymati sinxron ravishda o'zgarishi kerak bo'lsa, takrorlash ifodalarida zarur operatorlarni ';' bilan yozish orqali bunga erishish mumkin:

for(int i=10;j=2;i<=20;i++,j=i+10) {...};

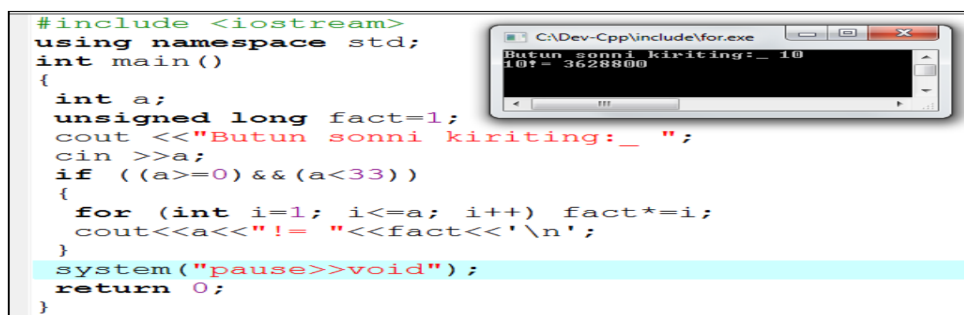
Takrorlash operatorining har bir qadamida j va i o'zgaruvchilarning qiymatlari mos ravishda o'zgarib boradi. for operatorida takrorlash tanasi bo'lmasligi ham mumkin. Masalan, dastur bajarilishini ma'lum bir muddatga «to'xtab» turish zarur bo'lsa, bunga takrorlashni hech qanday qo'shimcha ishlarni bajarmasdan amal qilishi orqali erishish mumkin:

```
#include <iostream.h>
int main()
{int delay;
...
for(delay=5000;delay>0;delay--) ;//bo'sh operator
...
return 0;}
```

Yuqorida keltirilgan 10 dan 20 gacha bo'lgan sonlar yig'indisini bo'sh tanali takrorlash operatori orqali hisoblash mumkin:

```
...
for (int i=10; i<=20; Summa+=i++);
...
```

Takrorlash operatori tanasi sifatida operatorlar bloki ishlatishini faktorialni hisoblash misolida ko'rsatish mumkin:



13.2-rasm. Takrorlash operatorining ishlatilishi

Dastur foydalanuvchi tomonidan 0 dan 33 gacha oraliqdagi son

kiritilganda amal qiladi, chunki 34! qiymati unsigned long uchun ajratilgan razryadlarga sig'maydi.

while takrorlash operatori

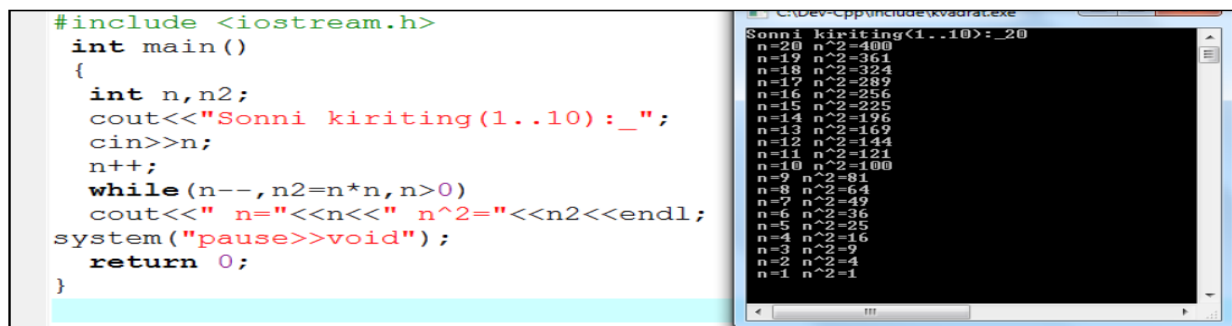
while takrorlash operatori, operator yoki blokni takrorlash sharti yolg'on (false yoki 0) bo'lguncha takror bajaradi. U quyidagi sintaksisga ega:

while (<ifoda>) <operator yoki blok>;

Agar <ifoda> rost qiymatli o'zgarmas ifoda bo'lsa, takrorlash cheksiz bo'ladi. Xuddi shunday, <ifoda> takrorlash boshlanishida rost bo'lib, uning qiymatiga takrorlash tanasidagi hisoblash ta'sir etmasa, ya'ni uning qiymati o'zgarmasa, takrorlash cheksiz bo'ladi. while takrorlash shartini oldindan tekshiruvchi takrorlash operatori hisoblanadi. Agar takrorlash boshida <ifoda> yolg'on bo'lsa, while operatori tarkibidagi <operator yoki blok> qismi bajarilmasdan cheklab o'tiladi. Ayrim hollarda <ifoda> qiymat berish operatori ko'rinishida kelishi mumkin. Bunda qiymat berish amali bajariladi va natija 0 bilan solishtiriladi.

Natija noldan farqli bo'lsa, takrorlash davom ettiriladi. Agar rost ifodaning qiymati noldan farqli o'zgarmas bo'lsa, cheksiz takrorlash ro'y beradi. Masalan:

while(1); // cheksiz takrorlash Xuddi **for** operatoridek, ',' yordamida <ifoda> da bir nechta amallar sinxron ravishda bajarish mumkin. Masalan, son va uning kvadratlarini chop qiladigan dasturda ushbu holat ko'rsatilgan:



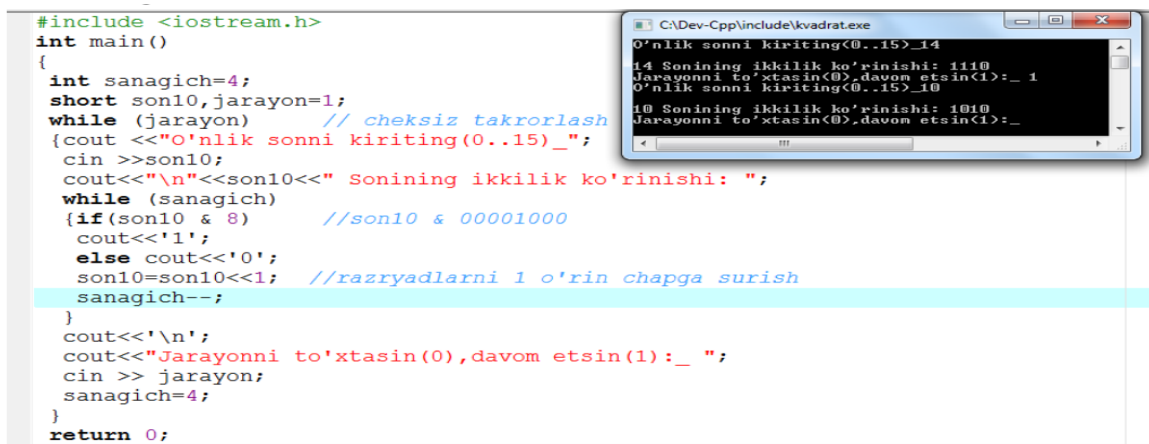
```
#include <iostream.h>
int main()
{
    int n,n2;
    cout<<"Sonni kiriting(1..10):_";
    cin>>n;
    n++;
    while(n-->0,n2=n*n,n>0)
    cout<<" n="<<n<<" n^2="<<n2<<endl;
    system("pause>>void");
    return 0;
}
```

The screenshot shows a C++ program in a text editor on the left and its execution output in a console window on the right. The program prompts the user to enter a number (1 to 10). The user enters 20. The program then prints a list of numbers from 1 to 20 and their squares. The output is as follows:

```
Sonni kiriting(1..10):_20
n=20 n^2=400
n=19 n^2=361
n=18 n^2=324
n=17 n^2=289
n=16 n^2=256
n=15 n^2=225
n=14 n^2=196
n=13 n^2=169
n=12 n^2=144
n=11 n^2=121
n=10 n^2=100
n=9 n^2=81
n=8 n^2=64
n=7 n^2=49
n=6 n^2=36
n=5 n^2=25
n=4 n^2=16
n=3 n^2=9
n=2 n^2=4
n=1 n^2=1
```

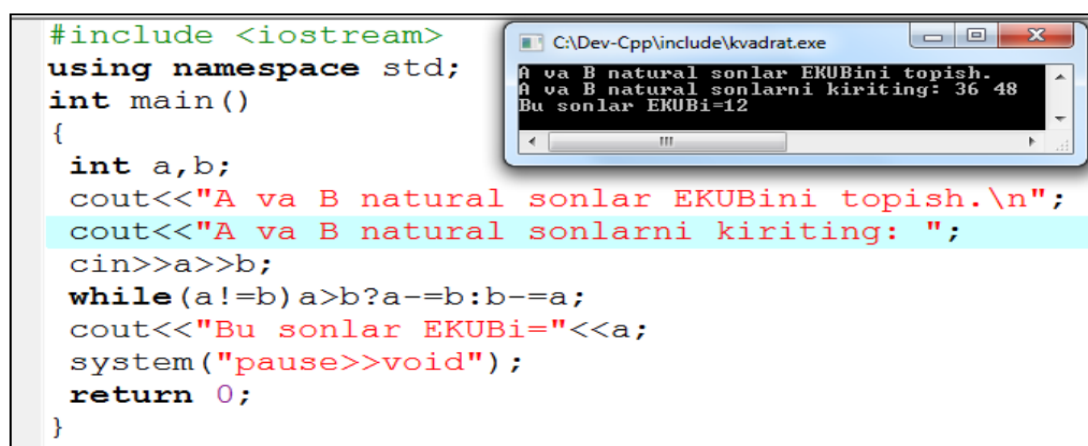
13.3-rasm. while operatorining ishlatilishi

Dasturdagi takrorlash operatori bajarilishida n soni 1 gacha kamayib boradi. Har bir qadamda n va uning kvadrati chop qilinadi. Shunga e'tibor berish kerakki, shart ifodasida operatorlarni yozilish ketma-ketligining ahamiyati bor, chunki eng oxirgi operator takrorlash sharti sifatida qaraladi va n qiymati 0 bo'lganda takrorlash tugaydi. Keyingi dasturda berilgan o'nlik sonning ikkilik ko'rinishini chop qilish masalasini yechishda while operatorini qo'llash ko'rsatilgan.



13.4-rasm. while operatorining ishlatilishi

Dasturda ichma-ich joylashgan takrorlash operatorlari ishlatilgan. Birinchisi, sonning ikkilik ko'rinishini chop qilish jarayonini davom ettirish sharti bo'yicha amal qiladi. Ichki joylashgan ikkinchi takrorlash operatoridagi amallar - har qanday, 0 dan 15 gacha bo'lgan sonlar to'rtta razryadli ikkilik son ko'rinishida bo'lishiga asoslangan. Unda kiritilgan sonning ichki, ikkilik ko'rinishida uchinchi razryadida 0 yoki 1 turganligi aniqlanadi ("son10&8"). Shart natijasi natija 1 (rost) bo'lsa, ekranga '1', aks holda '0' belgisi chop etiladi. Keyingi qadamda son razryadlari chapga bittaga suriladi va yana uchinchi razryaddagi raqam chop etiladi. Takrorlash sanagich qiymati 0 bo'lguncha ya'ni to'rt marta bajariladi va boshqaruv ichki takrorlash operatoridan chiqadi. *while* ta krorlash operatori yordamida samarali dastur kodi yozishga yana bir misol bu - ikkita natural sonlarning eng katta umumiy bo'luvchisini (EKUB) Evklid algoritmi bilan topish masalasini keltirishimiz mumkin:



13.5-rasm. Ichma-ich joylashgan takrorlash operatorining ishlatilishi

Butun turdagi a va b qiymatlari oqimdan o'qilgandan keyin toki ularning qiymatlari o'zaro teng bo'lmaguncha takrorlash jarayoni ro'y beradi. Takrorlashning har bir qadamida a va b sonlarning kattasidan kichigi ayriladi. Takrorlashdan keyingi ko'rsatma vositasida a

o'zgaruvchisining qiymati natija sifatida chop etiladi.

***do-while* takrorlash operatori**

do-while takrorlash operatori *while* operatoridan farqli ravishda oldin operator yoki blokni bajaradi, keyin takrorlash shartini tekshiradi. Bu qurilma takrorlash tanasini kamida bir marta bajarilishini ta'minlaydi. *do-while* takrorlash operatori quyidagi sintaksisga ega:

do <operator yoki blok>; *while* (<ifoda>);

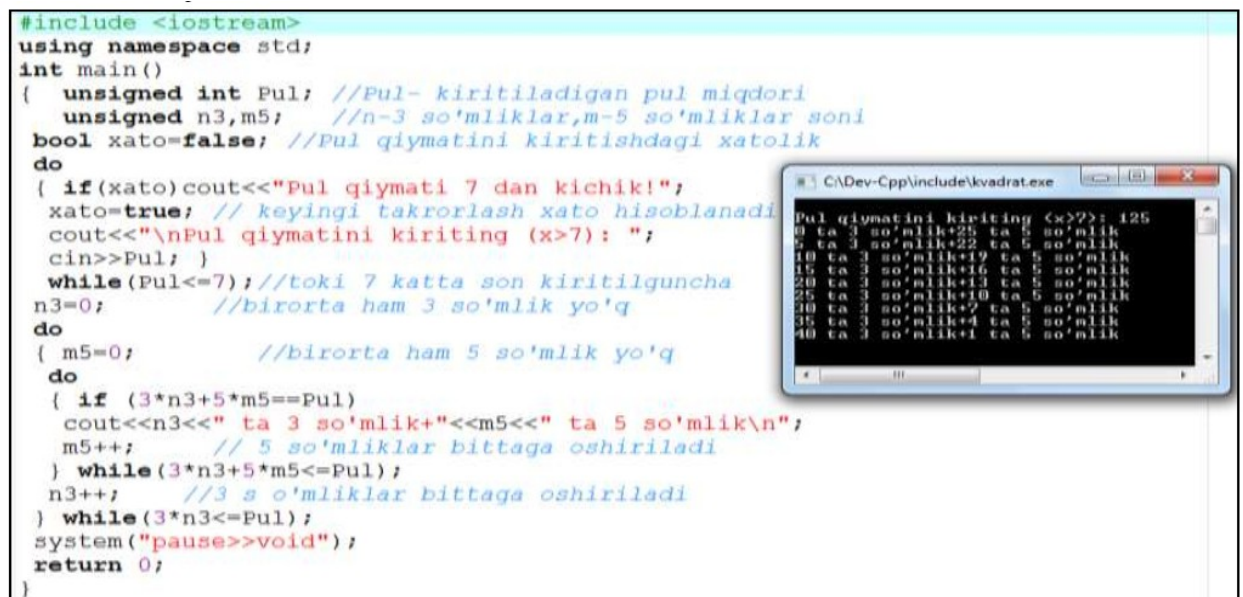
Bunday takrorlash operatorining keng qo'llaniladigan holatlari - bu takrorlashni boshlamasdan turib, takrorlash shartini tekshirishning iloji bo'lmagan holatlar hisoblanadi. Masalan, birorta jarayonni davom ettirish yoki to'xtatish haqidagi so'rovga javob olish va uni tekshirish zarur bo'lsin. Ko'rinib turibdiki, jarayonni boshlamasdan oldin bu so'rovni berishning ma'nosi yo'q. Hech bo'lmaganda takrorlash jarayonining bitta qadami amalga oshirilgan bo'lishi kerak:

```
#include <iostream.h>
int main()
{
    char javob;
    do
    {
        ... // dastur tanasi
        cout<<"Jarayonni to'xtatish (N):_ ";
        cin>>javob;
    } while(javob != 'N')
    return 0;
}
```

Dastur toki "Jarayonni to'xtatish (N):_ " so'roviga 'N' javobi kiritilmaguncha davom etadi. Bu operator ham cheksiz takrorlanishi mumkin:

do; while(1);

Masala. Har qanday 7 dan katta butun sondagi pul miqdorini 3 va 5 so'mliklarda berish mumkinligi isbotlansin. Qo'yilgan masala $p=3n+5m$ tenglamasi qanoatlantiruvchi m , n sonlar juftliklarini topish masalasidir (p -pul miqdori). Bu shartning bajarilishini m va n o'zgaruvchilarining mumkin bo'lgan qiymatlarining barcha kombinasiyalarida tekshirish zarur bo'ladi.



13.6-rasm. do while operatorining ishlatilishi

Dastur pul qiymatini kiritishni so'raydi (Pul o'zgaruvchisiga). Agar pul qiymati 7 sonidan kichik bo'lsa, bu haqda xabar beriladi va takror ravishda qiymat kiritish talab qilinadi. Pul qiymati 7 dan katta bo'lganda 3 va 5 so'mliklarning mumkin bo'lgan to'la kombinatsiyasi amalga oshirish uchun ichma-ich takrorlashlar amalga oshiriladi. Tashqi takrorlash n3 (3 so'mliklar miqdori) bo'yicha ichki takrorlash esa m5 (5 so'mliklar miqdori) bo'yicha, toki bu miqdordagi pullar qiymati Pul qiymatidan oshib ketmaguncha davom etadi. Ichki takrorlashda m5 o'zgaruvchisining har bir qiymatida « $3 \cdot n3 + 5 \cdot m5 == \text{Pul}$ » sharti tekshiriladi, agar u o'rinli bo'lsa, yechim variant sifatida n3 va m5 o'zgaruvchilar qiymatlari chop etiladi.

Break operatori. Takrorlash operatorlarining bajarilishida shunday holatlar yuzaga kelishi mumkinki, unda qaysidir qadamda takrorlashni yakuniga yetkazmasdan takrorlashdan chiqish zarurati bo'lishi mumkin.

Boshqacha aytganda, takrorlashni «uzish» kerak bo'lishi mumkin. Bunda break operatoridan foydalaniladi. break operatorini takrorlash operatori tanasining ixtiyoriy (zarur) joylariga qo'yish orqali shu joylardan takrorlashdan chiqishni amalga oshirish mumkin. E'tibor beradigan bo'lsak, switch-case operatorining tub mohiyatiga ham break operatorini qo'llash orqali erishilgan. Ichma - ich joylashgan takrorlash va switch operatorlarida break operatori faqat o'zi joylashgan blokdan chiqish imkoniyatini beradi.

Quyidagi programmada ikkita ichma-ich joylashgan takrorlash operatoridan foydalangan holda foydalanuvchi tomonidan kiritilgan qandaydir sonni 3 va 7 sonlariga nisbatan qanday oraliqqa tushishi aniqlanadi. Tashqi takrorlashda "Son kiriting (0- to'xtash):_" so'rovi beriladi va javob javob_son o'zgaruvchisiga o'qiladi. Agar son noldan farqli bo'lsa, ichki takrorlash operatorida bu sonning qandaydir oraliqqa tushishi aniqlanib, shu haqida xabar beriladi va ichki takrorlash operatoridan chiqiladi. Tashqi takrorlashdagi so'rovga javob tariqasida

0 kiritilsa, programma o'z ishini tugatadi.

```
#include <iostream.h >
int main ()
{
    int javob_son=0;
    do
    {
        while (javob_son)
        {
            if (javob_son<3)
            {cout<<"3 kichik ! "; break;}
            if ( 3<=javob_son && javob_son <=7)
            {cout<<"3 va 7 oralig'ida ! "; break;}
            if (javob_son >7 )
            {cout<<" 7 dan katta ! "; break;}
        }
        cout<<"\nSon kiriting ( 0 – to'xtash) :_";

        cin >> javob_son;
    }
    while (javob_son !=0)
    return 0;
}
```

Amaliyotda break operatoridan cheksiz takrorlashdan chiqishda foydalaniladi.

```
for ( ; ; )
{
    // 1 – shart
    if (....)
    {
        ....
        break;
    }
    // 2-shart
    if ( ... )
    {
        ....
        break;
    }
    ...
}
```

Bu misolda cheksiz for takrorlashidan 1 yoki 2 - shart bajarilganda chiqiladi.

Masala. Ishorasiz butun sonlar ketma-ketligi 0 qiymati bilan tugaydi,

0 ketma-ketlik hadi hisoblanmaydi. Ketma-ketlikni kamaymaydigan holda tartiblangan yoki yo'qligi aniqlansin.

```
#include <iostream.h>
int main()
{
    unsigned int Ai_1=0,Ai;
    cout<<"Sonlar ketma-ketligini kiriting"
    cout<<(0-tugash alomati):\n ";
    cin>>Ai;        // ketma-ketlikning birinchi hadi
    while (Ai)
    {
        Ai_1=Ai;
        cin>>Ai;        // navbatdagi had
        if (Ai_1>Ai) break;
    }
    if (Ai_1)
    {
        cout<<"Ketma-ketlik tartiblangan";
        if (!Ai) cout<<" emas!";
        else cout<<"!";
    }
    else cout<<"Ketma-ketlik bo'sh!";
    return 0;
}
```

Dastur ishga tushganda, boshda ketma-ketlikning birinchi hadi alohida o'qib olinadi (Ai o'zgaruvchisiga). Keyin Ai qiymati nolga teng bo'lmaguncha takrorlash operatori amal qiladi. Takrorlash tanasida Ai qiymati oldingi qiymat sifatida Ai_1 o'zgaruvchisida eslab qolinadi va navbatdagi had Ai o'zgaruvchisiga o'qiladi. Agar oldingi had navbatdagi haddan katta bo'lsa, break operatori yordamida takrorlash jarayoni uziladi va boshqaruv takrorlashdan keyingi shart operatoriga o'tadi. Bu yerdagi shart operatorlari mazmuni quyidagicha: agar Ai_1 noldan farqli bo'lsa, ketma-ketlikning kamida bitta hadi kiritilgan bo'ladi (ketma-ketlik mavjud) va oxirgi kiritilgan had tekshiriladi. O'z navbatida agar Ai noldan farqli bo'lsa, bu holat hadlar o'rtasida kamaymaslik sharti bajarilmaganligi sababli hadlarni kiritish jarayoni uzilganini bildiradi va bu haqda xabar chop etiladi. Aks holda ketma-ketlikni kamaymaydigan holda tartiblangan bo'ladi.

continue operatori

continue operatori xuddi break operatoridek takrorlash operatori tanasini bajarishni to'xtatadi, lekin takrorlashdan chiqib ketmasdan keyingi qadamiga «sakrab» o'tishini tayinlaydi. continue operatorini qo'llanishiga misol tariqasida 2 va 50 sonlar oralig'idagi tub sonlarni topadigan programma matnini keltiramiz.

```

#include <iostream.h>
int main()
{
    bool bulinadi=false;
    for (int i=2; i<50; i++)
    {
        for (int j=2; j<i/2; j++)
        {
            if (i%j) continue;
            bulinadi=true;
            break;
        }
    }
    // break bajarilganda boshqaruv o'tadigan joy

    if ( bulinadi) cout <<i<<" ";
    }
    return 0;
}

```

Keltirilgan programmada qo'yilgan masala ichma-ich joylashgan ikkita takrorlash operatorlari yordamida yechilgan. Birinchi takrorlash operatori 2 dan 50 gacha sonlarni hosil qilishga xizmat qiladi. Ichki takrorlash esa har bir hosil qilinayotgan sonni 2 sonidan toki shu sonning yarmigacha bo'lgan sonlarga bo'lib, qoldig'ini tekshiradi, agar qoldiq 0 sonidan farqli bo'lsa, navbatdagi songa bo'lish davom etadi, aks holda bulinadi o'zgaruvchisiga true qiymat berib, ichki takrorlash uziladi (son o'zining yarmigacha bo'lgan qandaydir songa bo'linar ekan, demak u tub emas va keyingi sonlarga bo'lib tekshirishga hojat yo'q). Ichki j bo'yicha takrorlashdan chiqqandan keyin bulinadi qiymati false bo'lsa (!bulinadi), i soni tub bo'ladi va u chop qilinadi.

goto operatori va nishonlar

Nishon - bu davomida ikkita nuqta (':') qo'yilgan identifikator. Nishon bilan qandaydir operator belgilanadi va keyinchalik, programmaning boshqa bir qismidan unga shartsiz o'tish amalga oshiriladi. Nishon bilan har qanday operator belgilanishi mumkin, shu jumladan e'lon operatori va bo'sh operatori ham. Nishon faqat funksiyalar ichida amal qiladi. Nishonga shartsiz o'tish goto operatori yordamida bajariladi. goto operatori orqali faqat uning o'zi joylashgan funksiya ichidagi operatorlarga o'tish mumkin. goto operatorining sintaksisi quyidagicha:

```
goto <nishon>;
```

Ayrim hollarda, goto operatorining «sakrab o'tishi» hisobiga xatoliklar yuzaga kelishi mumkin. Masalan,

```

int i=0;
i++; if (i) goto m;
int j;

```

m: j+=i;

operatorlarining bajarilishi xatolikka olib keladi, chunki *j* e'lon qilinmay qoladi. Shartsiz o'tish operatori programmani tuzishdagi kuchli va shu bilan birgalikda xavfli vositalardan biri hisoblanadi. Kuchliligi shundaki, uning yordamida algoritmning «boshi berk» joylaridan chiqib ketish mumkin. Ikkinchi tomondan, bloklarning ichiga o'tish, masalan, takrorlash operatorlarini ichiga «sakrab» kirish kutilmagan holatlarni yuzaga keltirishi mumkin. Shu sababli, imkon qadar goto operatoridan foydalanmaslik kerak, ishlatilgan taqdirda ham quyidagi qoidaga amal qilish zarur: «blok ichiga, if...else va switch operatorlari ichiga, hamda takrorlash operatorlari tanasiga tashqaridan kirish mumkin emas».

Garchi, nishon yordamida programmaning ixtiyoriy joyiga o'tish mumkin bo'lsa ham, boshlang'ich qiymat berish e'lonlaridan sakrab o'tish man etiladi, lekin bloklardan sakrab o'tish mumkin.

Masalan:

```
....
goto B;      \\ xato
float x=0.0;
goto B;      \\ to'g'ri
{ int n=10; x=n*x+x; }
B: cout << "x=" << x;
```

....

Xususan, nishon yordamida ichki blokdan tashqi blokka va tashqi blokdan ichki blokka o'tishga C++ tili ruxsat beradi:

```
{....
    goto ABC:
    ....
    { int i=15;
    ....
    ABC:
    ....
    goto XYZ;
    int y=10;
    ...
    XYZ:
    ....

    goto KLM;
    ...}
    ...
    int k=0;
    ...
    KLM:
```

```
...
}
```

Lekin, yuqorida keltirilgan misoldagi barcha o'tishlar mazmunan xato hisoblanadi. Quyidagi programmada ikkita natural sonlar eng katta umumiy bo'luvchini (EKUB) topish masalasidagi takrorlash jarayonini nishon va goto operatori vositasida amalga oshirish ko'rsatilgan:

```
int main()
{
    int a,b;
    cout<<"A va B natural sonlar EKUBini
topish.\n";
    cout<<"A va B natural sonlarni kiriting: "
    cin>>a>>b;
    nishon:
    if(a==b)
    {
        cout<<"Bu sonlar EKUBi: "<<a;
        return 0;
    }
    a>b?a-=b:b-=a;
    goto nishon;
}
```

dasturdagi nishon bilan belgilangan operatorlarda a va b sonlarni tengligi tekshiriladi. Agar ular teng bo'lsa, ixtiyoriy bittasi, masalan a soni EKUB bo'ladi va funksiyadan chiqiladi. Aks holda, bu sonlarning kattasidan kichigi ayiriladi va goto orqali ularning tengligi tekshiriladi. Takrorlash jarayoni a va b sonlar o'zaro teng bo'lguncha davom etadi.

Shuni qayd etish kerakki, bu masalani takrorlash operatorlari yordamida bajarish ancha samarali hisoblanadi.

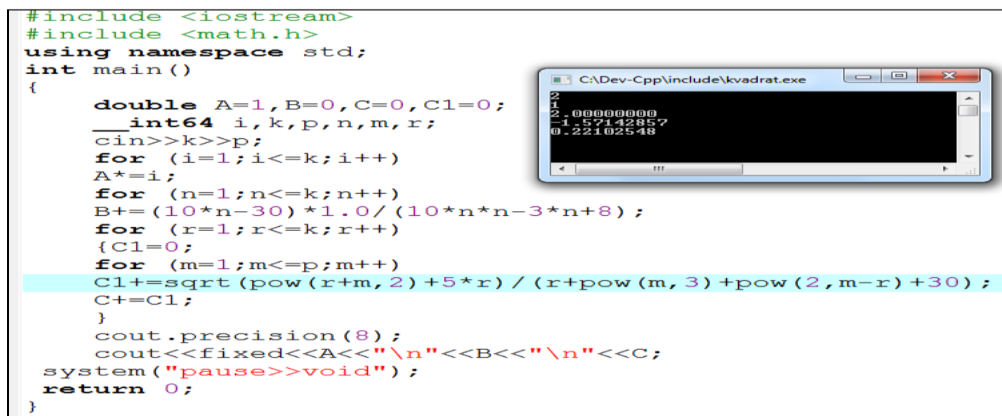
Berilgan masala uchun algoritm.

Berilgan masalada uchta A, B va C misollar mavjud. A misol: k! hisoblash, ya'ni $1 \cdot 2 \cdot \dots \cdot k$ ko'paytma natijasini olish lozim. Bu yerda $A=1$ haqiqiy o'zgaruvchi olamiz va k ta takrorlash natijasida 1 dan k gacha bo'lgan sonlarni A ga ko'paytiramiz.

B misol: $B=0$ haqiqiy o'zgaruvchi olamiz. $\frac{10n-30}{10n^2-3n+8}$ ifodaning natijasini $n=1$ dan toki $n=k$ gacha almashtirib hisoblaymiz va har birining natijasini B ga qo'shamiz.

C misol: Ushbu misol ichma-ich sikl bo'lganligi sababli $C=0$ va $C1=0$ haqiqiy qiymatli o'zgaruvchi olamiz. Tashqi siklning har bir holati uchun ichki sikl hisoblanib C1 ga qo'shiladi va natija C ga qo'shiladi.

Asosiy dastur matni.

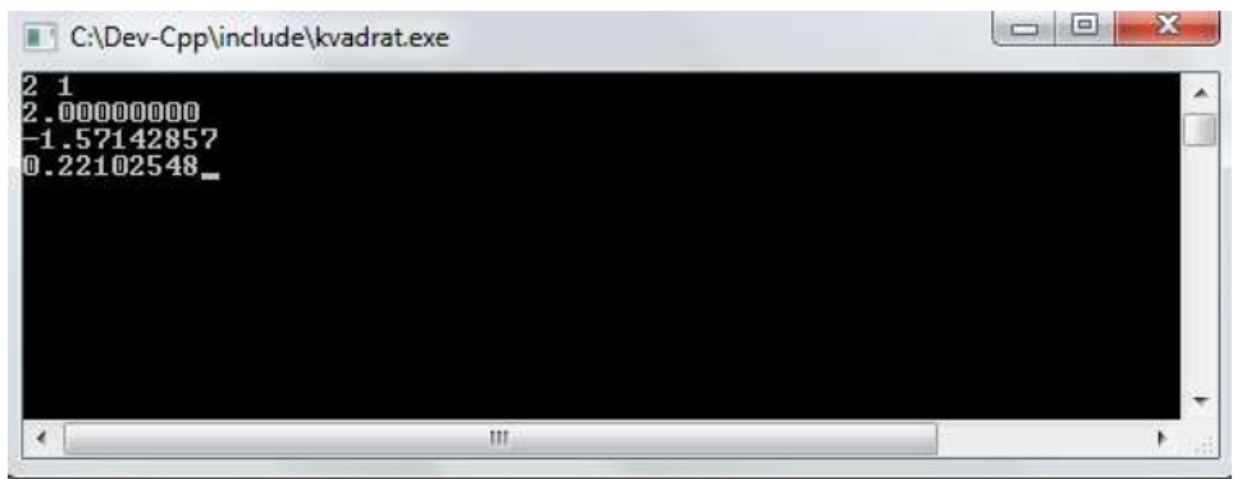


13.7-rasm. ichma-ich siklar ishlatilishi

Berilgan misol ishlanish natijalari tahlili.

$k!$ faktorialni hisoblash *for* ($i=1; i \leq k; i++$) $A*=i$; bu yerda $i=1$ siklning initsializatsiya qismi, ya'ni i o'zgaruvchi 1 qiymatini o'zlashtiradi. $i \leq k$ siklning shart qismi, ushbu shart bajarilmaguncha davom etadi. $i++$ siklning qadamlar qismi, ya'ni har safar A ga ko'paytirilib bir birlikka oshiriladi. Ikkinchi va uchinchi misolni hisoblashda ham sikldan huddi shu uslubda foydalanamiz.

Natija



2 . Mavzu bo'yicha sodda dasturlar tuzish.

Topshiriqlar. Quyidagi variantlarda berilgan takrorlanuvchi algoritmlar dasturini tuzing.

№	Variantlar	№	Kirish	Chiqish
1	$a) \sum_{n=1}^m \frac{1}{n^3}; \quad b) \sum_{R=1}^l \frac{R^3}{R^4 + 3R^2 + e^{-R}};$ $c) \prod_{R=1}^l \prod_{i=1}^m \frac{R^i + 1}{R^4 + 3^i - \cdot R + e^{-R}};$	1	$m=2$ $l=1$	a)1.12500000 b)0.22894405 c)0.08832821
		2	$m=5$ $l=8$	a)1.18566204 b)1.56011635 c)0.00000000
2	$a) \sum_{n=1}^k \frac{2}{n^3(n+1)}; \quad b) \sum_{R=1}^y \frac{R^2 + R-2 }{\ln R + 3R};$ $c) \sum_{R=1}^k \sum_{m=1}^p \frac{R^2 + R-2 }{\ln R + 3R};$	1	$k=2$ $p=1$	a)1.08333333 b)0.66666667 c)2.55180630
		2	$k=5$ $p=8$	a)1.11076852 b)12.34348563 c)44.68083491
3	$a) \sum_{n=1}^k \frac{3}{(2n+1)^3}; \quad b) \sum_{R=1}^p \frac{R+1}{R + e^{-R} + 1};$ $c) \prod_{R=1}^k \sum_{i=1}^p \frac{R+3}{R^3 + 3R + i^3};$	1	$k=2$ $p=1$	a)0.13511111 b)0.90524345 c)0.26666667
		2	$k=5$ $p=8$	a)0.15022664 b)169.36763033 c)0.06609173
4	$a) \sum_{R=1}^k \frac{4}{R(R+1)}; \quad b) \sum_{R=1}^p \frac{R^{R+1}}{2^{R+1} + (R+1)^4};$ $c) \sum_{R=1}^k \prod_{i=1}^p \frac{(R+1)^i + 4}{(-1)^R + 3(-1)^i + i^R};$	1	$k=2$ $p=1$	a)2.66666667 b)0.05000000 c)-9.00000000
		2	$k=5$ $p=8$	a)3.33333333 b)20424.87608993 c)-204508918.09011152
5	$a) \sum_{m=1}^k \frac{5}{m^2 + m + 4}; \quad b) \sum_{R=1}^p \frac{(100-R)^2}{\lg R + 5^{-R}};$ $c) \sum_{i=1}^k \sum_{R=1}^p \frac{(-1)^i \cos(i+R) + 5}{5i + 7^{-R} + i^{-R}};$	1	$k=2$ $p=1$	a)1.33333333 b)49005.00000000 c)1.25847755
		2	$k=5$ $p=8$	a)2.00122549 b)155695.30188719 c)16.88785843
6	$a) \prod_{n=1}^k \frac{n+6}{n^2 + 4n + 1}; \quad b) \sum_{i=1}^p \frac{i+6}{i^4 + 27i + 7};$	1	$k=2$ $p=1$	a)0.71794872 b)0.20000000 c)15.21598260

	$c) \prod_{R=1}^k \prod_{i=1}^p (-1)^i \frac{\sqrt{5i^4 + e^{-R} + 6}}{\cos(i+1)^3 - R^{-i}};$	2	$k=5$ $p=4$	a)0.02128306 b)0.38410472 c)5851732838696673300000 000000.00000000
7	$a) k! + 7; \quad b) \sum_{i=1}^p \frac{(-1)^i \cdot 7^{-i}}{1+i+i^2};$ $c) \sum_{i=1}^k \sum_{R=1}^p \frac{(-1)^i \cos(i+R) + 5}{5i + 7^{-R} + i^{-R}};$	1	$k=2$ $p=1$	a)9.0000000 b)-0.0476190 c)1.2584776
		2	$k=5$ $p=8$	a)127.00000000 b)-0.0449098 c)16.8878584
8	$a) (k+1)!; \quad b) \sum_{n=1}^p \frac{10n-8}{10n^2-3n+8};$ $c) \sum_{i=1}^k \sum_{m=1}^p \left[\frac{i^m + 4m + e^m}{m^i} \right];$	1	$k=2$ $p=1$	a)6.00000000 b)0.13333333 c)16.43656366
		2	$k=5$ $p=8$	a)720.00000000 b)1.43814142 c)942.46222819
9	$a) \prod_{n=1}^k \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-2}}}; \quad b) \sum_{i=1}^p (2i + 5i + 9);$ $c) \sum_{R=1}^k \prod_{m=1}^p \sqrt{\left \frac{R + m^3 + e^{-m} + 9}{\log_2 R + (mR)^3} \right };$	1	$k=2$ $p=1$	a)16.47419724 b)16.00000000 c)4.54389514
		2	$k=5$ $p=8$	a)9644244424.99799920 b)324.00000000 c)6.96548965
10	$a) \prod_{n=1}^k \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-n}}}; \quad b) \prod_{n=1}^p \frac{n^2 + 3n + 10}{n^2 + 7n + 91};$ $c) \sum_{R=1}^k \sum_{m=1}^p \frac{\sqrt{tg(R+m)^2 + 10R}}{R + m^{-R} + e^{m-R}};$	1	$k=2$ $p=1$	a)16.47419724 b)3.02633015 c)2.42622166
		2	$k=5$ $p=8$	a)9644244424.99799920 b)13252651.75703296 c)28.57163344
11	$a) \prod_{n=1}^k \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-n}}}; \quad b) \prod_{n=1}^p \frac{1}{n^4 + 1};$ $c) \prod_{i=1}^k \sum_{m=1}^p \left[e^{\sqrt{i^2+m-i}} + \frac{i^2 + 11}{m^4 + i^{-m}} \right];$	1	$k=2$ $p=1$	a)16.7419724 b)0.50000000 c)136.46058442
		2	$k=5$ $p=8$	a)9644244424.99799920 b)0.00000000 c)1293947622309.52320000
12	$a) \prod_{i=1}^k \frac{i^4 + i^2 + 12}{\sqrt{i^3 + e^{-i}}}; \quad b) \sum_{R=1}^p \frac{R + 1.125}{R^5 + 5R + 1.2}$ $c) \sum_{m=1}^k \prod_{n=1}^p \sqrt{\left \frac{m^3 - n^2 + 3.4}{m^{-n} - n^{-m} + 12} \right };$	1	$k=2$ $p=1$	a)134.29701962 b)0.29513889 c)1.48326269
		2	$k=5$ $p=8$	a)5535312.16043921 b)0.39191141 c)5141.72021009
13	$a) \prod_{n=1}^k \frac{13}{n^3 + 5n + 7}; \quad b) \sum_{m=1}^p \frac{(-1)^m \sqrt{m}}{2^{-2m}};$	1	$k=2$ $p=1$	a)0.52000000 b)-4.00000000 c)-0.00258075
		2	$k=5$	a)0.00163191

	$c) \prod_{i=1}^k \prod_{m=1}^p \operatorname{tg} \frac{i^{-m} - i^{-3} - i^2 + 1.3}{m^{-i} + m^{-6} + im + 13}$		$p=8$	$b)150178.96310968$ $c)0.00000000$
14	$a) \sum_{R=1}^k \frac{R^2 + 14}{\sqrt[3]{3^{-R} + R^3}}; \quad b) \prod_{n=1}^p \frac{n + 1.23}{n + \frac{1}{n}};$	1	$k=2$ $p=1$	$a)19.31060301$ $b)1.11500000$ $c)-0.78146109$
	$c) \sum_{i=1}^k \sum_{m=1}^p \lg \frac{\sqrt{m^2 + e^{m-i}}}{i^2 + 2^{m-1}};$	2	$k=5$ $p=8$	$a)30.97176927$ $b)4.13609486$ $c)-28.76105925$
15	$a) \prod_{i=1}^k \frac{ i - 15 + i^3}{\ln i + 7i}; \quad b) \sum_{R=1}^p \frac{(-1)^R \cdot (R + 1)}{R^3 + R^2 + 1}$	1	$k=2$ $p=1$	$a)3.06265223$ $b)-0.66666667$ $c)0.03451403$
	$c) \prod_{n=1}^k \sum_{m=1}^p (-1) \frac{m \ln(n + 5) + 1.5}{2^{m+3} + (n + 3)^{-m} + nm};$	2	$k=5$ $p=8$	$a)50.86914472$ $b)-0.49910258$ $c)-0.07829932$
16	$a) \sum_{i=-22}^k \frac{\sqrt{ i - 2i^3 } + 16}{\ln i + 3 + 1.6};$	1	$k=2$ $p=1$	$a)462.95921847$ $b)-0.75299133$ $c)1.00000000$
	$b) \sum_{n=1}^p (-1)^n \frac{n + 1.258974}{2n^4 + 1};$ $c) \prod_{R=4}^k \prod_{m=1}^p \frac{\sqrt{ R^m + 4R - m + 1.6 }}{\sin(m + R) - m};$	2	$k=5$ $p=8$	$a)9644244424.99799920$ $b)0.00000000$ $c)1293947622309.52320000$
17	$a) \prod_{R=1}^k \frac{R + 17}{2R^2 + 9};$	1	$k=2$ $p=1$	$a)1.82887701$ $b)-0.17647059$ $c)1.04368781$
	$b) \sum_{R=1}^p (-1) \cdot \frac{R \cdot \sqrt[R]{R + 1} + R^2}{2R^2 + 4R + 11};$ $c) \sum_{m=1}^k \prod_{n=1}^p \sqrt{\frac{m^3 - n^2 + 1.7}{m^n + m^m + 12}};$	2	$k=5$ $p=8$	$a)0.25873599$ $b)-2.83439919$ $c)0.51638448$
18	$a) \sum_{n=1}^k \frac{18}{5 - 17n + n^3};$	1	$k=2$ $p=1$	$a)-2.49350649$ $b)0.41833001$ $c)1.00407388$
	$b) \prod_{m=-12}^p \frac{m^2 \sqrt{ m } + 1.8}{m^2 + 4m + (-1)^m};$ $c) \sum_{i=1}^k \prod_{R=1}^p \frac{\sqrt{e^{i+R} (i + R)^R}}{ 4i^3 + R^4 };$	2	$k=5$ $p=8$	$a)14.95912509$ $b)2.78668250$ $c)24681022165.89471400$
19	$a) \sum_{n=1}^k \frac{19n}{3 + n + n^2};$	1	$k=2$ $p=1$	$a)8.02222222$ $b)0.00000000$ $c)1.00000000$
	$b) \sum_{t=3}^p \frac{\operatorname{tg}(t + 3)}{t^3 + 2t + e^{t-1}};$	2	$k=5$ $p=8$	$a)18.00535793$ $b)-0.17303251$ $c)9737209203758591500000$

	$c) \prod_{i=1}^k \prod_{m=2}^p (i \cdot m)^{\frac{i\sqrt{ i } + m}{(i+m)}};$			00000000000.00000000
20	$a) \prod_{n=1}^k (-1)^n \frac{1+n^2}{1+n^3};$ $b) \sum_{m=10}^p \frac{\sin(m)}{\sqrt[4]{m^2 + e^{m+13}}};$ $c) \prod_{n=1}^k \prod_{R=1}^p \frac{n^3 - R^2 + 20}{(n-R + n)^R};$	1	$k=2$ $p=1$	$a)-0.55555556$ $b)0.00000000$ $c)180.00000000$
		2	$k=5$ $p=18$	$a)-0.25360195$ $b)-0.00392117$ $c)0.00000000$

14-amaliy mashg'ulot. C++ algoritmik tilning bir o'lchamli va ko'p o'lchamli massivlari.

Ishdan maqsad: C++ algoritmik tilning bir o'lchamli va ko'p o'lchamli massivlari bilan ishlash.

Vazifa.

- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Mavzu bo'yicha sodda dasturlar tuzish.

1 . Berilgan nazariy materiallarni o'rganish.

Bir o'lchovli massivlarni e'lon qilish va ular ustida amallar. Xotirada ketma-ket (regulyar) joylashgan bir xil turdagi qiymatlarga massiv deyiladi. Odatda massivlarga zarurat, katta hajmdagi, lekin cheklangan miqdordagi va tartiblangan qiymatlarni qayta ishlash bilan bog'liq masalalarni yechishda yuzaga keladi. Faraz qilaylik, talabalar guruhining reyting ballari bilan ishlash masalasi qo'yilgan. Unda guruhning o'rtacha reytingini aniqlash, reytinglarni kamayishi bo'yicha tartiblash, konkret talabaning reytingi haqida ma'lumot berish va boshqa masala ostilarini yechish zarur bo'lsin. Qayd etilgan masalalarni yechish uchun berilganlarning (reytinglarning) tartiblangan ketma-ketligi zarur bo'ladi. Bu yerda tartiblanganlik ma'nosi shundaki, ketma-ketlikning har bir qiymati o'z o'rniga ega bo'ladi (birinchi talabaning reytingi massivda birinchi o'rinda, ikkinchi talabaniki - ikkinchi o'rinda va hakoza). Berilganlar ketmaketligini ikki xil usulda hosil qilish mumkin. Birinchi yo'l - har bir reyting uchun alohida o'zgaruvchi aniqlash: $Reyting_1, \dots, Reyting_N$.

Lekin, guruhdagi talabalar soni yetarlicha katta bo'lganda, bu o'zgaruvchilar qatnashgan programmani tuzish katta qiyinchiliklarni yuzaga keltiradi. Ikkinchi yo'l - berilganlar ketma-ketligini yagona nom bilan aniqlab, uning qiymatlariga murojaatni, shu qiymatlarning ketma-ketlikda joylashgan o'rnining nomeri (indeksi) orqali amalga oshirishdir. Reytinglar ketma-ketligini Reyting deb nomlab, undagi qiymatlariga $Reyting_1, \dots, Reyting_N$ ko'rinishida murojaat qilish mumkin. Odatda berilganlarning bunday ko'rinishiga massivlar deyiladi. Massivlarni matematikadagi sonlar vektoriga o'xshatish mumkin, chunki vektor ham o'zining individual nomiga ega va u

fiksirlangan miqdordagi bir turdagi qiymatlardan - sonlardan iboratdir.

Demak, massiv - bu fiksirlangan miqdordagi ayrim qiymatlarning (massiv elementlarining) tartiblangan majmuasidir. Barcha elementlar bir xil turda bo'lishi kerak va bu tur element turi yoki massiv uchun tayanch tur deb nomlanadi. Yuqoridagi keltirilgan misolda Reyting - haqiqiy turdagi vektor deb nomlanadi.

Programmada ishlatiladigan har bir konkret massiv o'zining individual nomiga ega bo'lishi kerak. Bu nomni to'liq o'zgaruvchi deyiladi, chunki uning qiymati massivning o'zi bo'ladi. Massivning har bir elementi massiv nomi, hamda kvadrat qavsga olingan va element selektori deb nomlanuvchi indeksni ko'rsatish orqali oshkor ravishda belgilanadi. Murojaat sintaksisi:

$\langle \text{massiv nomi} \rangle [\langle \text{indeks} \rangle]$

Bu ko'rinishga xususiy o'zgaruvchi deyiladi, chunki uning qiymati massivning alohida elementidir. Bizning misolda Reyting massivining alohida komponentalariga Reyting[1],...,Reyting[N] xususiy o'zgaruvchilar orqali murojaat qilish mumkin. Boshqacha bu o'zgaruvchilar indeksli o'zgaruvchilar deyiladi. Massiv indeksi sifatida butun son qo'llaniladi. Umuman olganda indeks sifatida butun son qiymatini qabul qiladigan ixtiyoriy ifoda ishlatilishi mumkin va uning qiymati massiv elementi nomerini aniqlaydi. Ifoda sifatida o'zgaruvchi ham olinishi mumkinki, o'zgaruvchining qiymati o'zgarishi bilan murojaat qilinayotgan massiv elementini aniqlovchi indeks ham o'zgaradi.

Shunday qilib, programmadagi bitta indeksli o'zgaruvchi orqali massivning barcha elementlarini belgilash (aniqlash) mumkin bo'ladi. Masalan, Reyting[I] o'zgaruvchisi orqali I o'zgaruvchining qiymatiga bog'liq ravishda Reyting massivining ixtiyoriy elementiga murojaat qilish mavjud.

Haqiqiy turdagi (float, double) qiymatlar to'plami cheksiz bo'lganligi sababli ular indeks sifatida ishlatilmaydi. C++ tilida indeks doimo 0 dan boshlanadi va uning eng katta qiymati massiv e'lonidagi uzunlikdan bittaga kam bo'ladi.

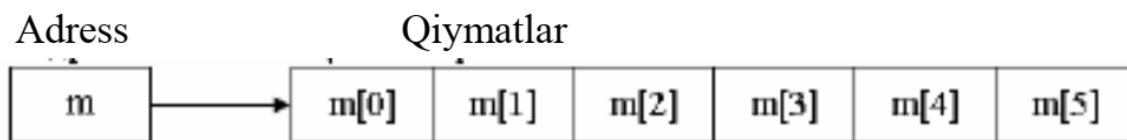
Massiv e'loni quyidagicha bo'ladi:

$\langle \text{tur} \rangle \langle \text{nom} \rangle [\langle \text{uzunlik} \rangle] = \{ \text{boshlang'ich qiymatlar} \}.$

Bu yerda $\langle \text{uzunlik} \rangle$ - o'zgarmas ifoda. Misollar:

int m[6]={1,4,-5,2,10,3}; float a[4];

Massiv statik va dinamik bo'lishi mumkin. Statik massivning uzunligi oldindan ma'lum bo'lib, u xotirada ma'lum adresdan boshlanib ketma-ket joylashadi. Dinamik massivni uzunligi programma bajarilish jarayonida aniqlanib, u dinamik xotiradagi ayni paytda bo'sh bo'lgan adreslarga joylashadi. Masalan, **int m[6];** ko'rinishida e'lon qilingan bir o'lchamli massiv elementlari xotirada quyidagicha joylashadi.



Bir o'lchamli massivning xotirada joylashuvi

Massivning i - elementiga $m[i]$ yoki $*(m+i)$ - vositali murojaat qilish mumkin. Massiv uzunligini $\text{sizeof}(m)$ amali orqali aniqladi. Massiv e'lonida uning elementlariga boshlang'ich qiymatlar berish mumkin va uning bir nechta variantlari mavjud.

1) o'lchami ko'rsatilgan massiv elementlarini to'liq inisializasiyalash:

int t[5]={-10,5,15,4,3};

Bunda 5 ta elementdan iborat bo'lgan t nomli butun turdagi bir o'lchamli massiv e'lon qilingan va uning barcha elementlariga boshlang'ich qiymatlar berilgan. Bu e'lon quyidagi e'lon bilan ekvivalent:

int t[5]; t[0]=-10; t[1]=5; t[2]=15; t[3]=4; t[4]=3;

2) o'lchami ko'rsatilgan massiv elementlarini to'liqmas inisializasiyalash:

int t[5]={-10,5,15};

Bu yerda faqat massiv boshidagi uchta elementga boshlang'ich qiymatlar berilgan. Shuni aytib o'tish kerakki, massivning boshidagi yoki o'rtasidagi elementlariga qiymatlar bermasdan, uning oxiridagi elementlarga boshlang'ich qiymat berish mumkin emas. Agarda massiv elementlariga boshlang'ich qiymat berilmasa, unda kelishuv bo'yicha static va extern modifikatori bilan e'lon qilingan massiv uchun elementlarning qiymati 0 soniga teng deb, automatic massivlar elementlarining boshlang'ich qiymatlari noma'lum hisoblanadi.

3) o'lchami ko'rsatilmagan massiv elementlarini to'liq inisializasiyalash:

int t[]={-10,5,15,4,3};

Bu misolda massivni barcha elementlariga qiymatlar berilgan hisoblanadi, massiv uzunligi kompilyator tomonidan boshlang'ich qiymatlar soniga qarab aniqlanadi. Agarda massiv uzunligi berilmasa, boshlang'ich qiymati berilishi shart.

Massivni e'lon qilishga misollar:

char ch[4]={'a','b','c','d'}; //belgilar massivi

int in[6]={10,20,30,40}; // butun sonlar massivi

char str[]="abcd"; //sitr uzunligi 5 ga teng, chunki uning oxiriga '\0' belgisi qo'shiladi

char str[]={ 'a','b','c','d'}; // yuqoridagi satrning boshqacha yozilishi

Masala. Berilgan massiv elementlarining har xil qiymatlilari sonini toping. N – massiv elementlari soni ($1 \leq N \leq 100000$). ($-1000 \leq a[i] \leq 1000$).

Dastur matni:

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main() {
    int k, s=0, a[2001]={0}, n, i;
    cin>>n;
    for (i=0; i<n; i++)
    {
        cin>>k;
        a[k+10]=1;
    }
    for (i=0; i<=2001; i++)
        s+=a[i];
    cout<<s;
    system("pause>>void");
    return 0;}

```

14.1-rasm. Massiv elementlari sonini topish.

Berilgan masala uchun algoritm. Berilgan masalada massiv elementlari qiymatlari ishoralarini almashtirish va almashtirishlar sonini aniqlash uchun bizga massiv elementlarining nechitasi 0 ga teng ekanligini aniqlash lozim bo'ladi. Chunki 0 qiymatli sonning ishorasi yo'q.

Asosiy dastur matni.

```

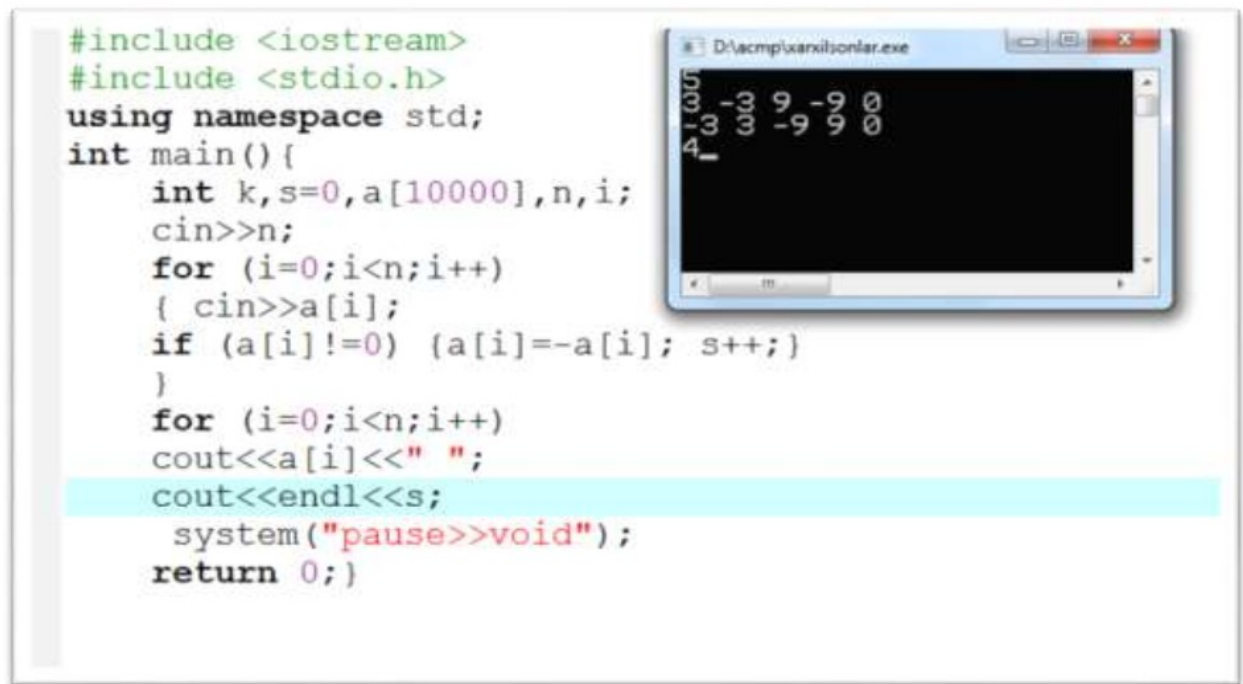
#include <iostream>
#include <stdio.h>
using namespace std;
int main() {
    int k, s=0, a[10000], n, i;
    cin>>n;
    for (i=0; i<n; i++)
    { cin>>a[i];
      if (a[i]!=0) {a[i]=-a[i]; s++;}
    }
    for (i=0; i<n; i++)
    cout<<a[i]<<" ";
    cout<<endl<<s;
    system("pause>>void");
    return 0;}

```

14.2-rasm. Berilgan masala uchun dastur

Berilgan misol ishlanish natijalari tahlili. Berilgan misolda massiv elementlarini kiritish jarayonida ushbu massiv elementlarining qiymati 0 dan farqli bo'lsa uning qiymatini qaramaqarshisiga almashtirilyapti va ushbu almashtirishlar shu joyda sanalayapti.

Natija



14.3-rasm. Berilgan dastur natijasi

Massiv chegarasidan tashqariga chiqish (ya'ni mavjud bo'lmagan elementni o'qish/yozishga urinish) dastur bajarilishida kutilmagan natijalarga olib kelishi mumkin. SHuni ta'kidlab o'tamizki, bu eng ko'p tarqalgan xatolardan biridir.

Agar massiv initsializatsiya qilinganda elementlar chegarasi ko'rsatilgan bo'lsa, ro'yxatdagi elementlar soni bu chegaradan kam bo'lishi mumkin, lekin ortiq bo'lishi mumkin emas.

Misol uchun `int a[5]={2,-2}`. Bu holda `a[0]` va `a[1]` qiymatlari aniqlangan bo'lib, mos holda 2 va -2 ga teng. Agar massiv uzunligiga qaraganda kamroq element berilgan bo'lsa, qolgan elementlar 0 hisoblanadi:

```
int a10[10]={1, 2, 3, 4}; //va 6 ta nol
```

Agar nomlangan massivning tavsifida uning o'lchamlari ko'rsatilmagan bo'lsa, kompilyator tomonidan massiv chegarasi avtomatik aniqlanadi:

```
int a3[]={1, 2, 3};
```

Massivda musbat elementlar soni va summasini hisoblash.

```

#include <iostream>
using namespace std;
int main()
int s=0,k=0;
int x[]={-1,2,5,-4,8,9};
for(int i=0; i<6; i++)
{
    if (x[i]<=0) continue;
    k++;
    s+=x[i];
}

```

```
};
cout<<k<<endl;
cout<<s;
return 0;
};
```

Massivning eng katta, eng kichik elementi va o'rta qiymatini aniqlash:

```
#include <iostream>
using namespace std;
int main()
{
    int i,j,n;
    int min,max,s;
    float a,b,d,x[100];
    while(1)
    {
        cout<<("\n n=");
        cin>>n;
        if ( n>0 && n<=100 ) break;
        cout<<("\n Hato 0<n<101 bo'lishi kerak");
    }
    cout<<("\n elementlar qiymatlarini kiriting:\n");
    for (i=0;i<n;i++)
    {
        cout<<"x["<<i<<"]="";
        cin>>x[i];
    }
    max=x[0];
    min=x[0];
    for (s=0,i=0;i<n;i++)
    {
        s+=x[i];
        if (max<x[i]) max=x[i];
        if (min>x[i]) min=x[i];
    };
    s/=n;
    cout<<("\n max="<<max;
    cout<<("\n min="<<min;
    cout<<("\n o'rta qiymat="<<s;
    return 0;
}
```

Massivlarni navlarga ajratish. Navlarga ajratish - bu berilgan ko'plab ob'ektlarni biron-bir belgilangan tartibda qaytadan guruhlash jarayoni.

Massivlarning navlarga ajratilishi tez bajarilishiga ko'ra farqlanadi. Navlarga ajratishning $n \cdot n$ ta qiyoslashni talab qilgan oddiy usuli va $n \cdot \log(n)$ ta qiyoslashni talab qilgan tez usuli mavjud. Oddiy usullar navlarga ajratish tamoyillarini

tushuntirishda qulay hisoblanadi, chunki sodda va kalta algoritmlarga ega. Murakkablashtirilgan usullar kamroq sonli operatsiyalarni talab qiladi, biroq operatsiyalarning o'zi murakkabroq, shuning uchun uncha katta bo'lmagan massivlar uchun oddiy usullar ko'proq samara beradi.

Oddiy usullar uchta asosiy kategoriyaga bo'linadi:

- oddiy kiritish usuli bilan navlarga ajratish;
- oddiy ajratish usuli bilan navlarga ajratish;
- oddiy almashtirish usuli bilan navlarga ajratish

Oddiy kiritish usuli bilan navlarga ajratish

Massiv elementlari avvaldan tayyor berilgan va dastlabki ketma-ketliklarga bo'linadi. $i=2$ dan boshlab, har bir qadamda dastlabki ketma-ketlikdan i -nchi element chiqarib olinadi hamda tayyor ketma-ketlikning kerakli o'rniga kiritib qo'yiladi. Keyin i bittaga ko'payadi va h.k.

Kerakli joyni izlash jarayonida, ko'proq o'ngdan bitta pozitsiyadan tanlab olingan elementni uzatish amalga oshiriladi, ya'ni tanlab olingan element, $j:=i-1$ dan boshlab, navlarga ajratib bo'lingan qismning navbatdagi elementi bilan qiyoslanadi. Agar tanlab olingan element $a[i]$ dan katta bo'lsa, uni navlarga ajratish qismiga qo'shadilar, aks holda $a[j]$ bitta pozitsiyaga suriladi, tanlab olingan elementni esa navlarga ajratilgan ketma-ketlikning navbatdagi elementi bilan qiyoslaydilar. To'g'ri keladigan joyni qidirish jarayoni ikkita turlicha shart bilan tugallanadi:

- agar $a[j]>a[i]$ elementi topilgan bo'lsa;
- agar tayyor ketma-ketlikning chap uchiga yetilgan bo'lsa.

```
int i, j, x;
```

```
for(i=1; i<n; i++)
```

```
{
```

```
x=[i]; // kiritib qo'ishimiz lozim bo'lgan elementni esda saqlab qolamiz
```

```
j=i-1;
```

```
while(x<a[j]&& j>=0) // to'g'ri keladigan joyni qidirish
```

```
}
```

```
a[j+1]=a[j] / o'nga surilish
```

```
j--;
```

```
}
```

```
a[j+1]=x; // elementni kiritish
```

```
}
```

Massivning minimal elementi tanlanadi hamda massivning birinchi elementi bilan joy almashtiriladi. Keyin jarayon qolgan elementlar bilan takrorlanadi va h.k.

```
int i, min, n_min, j;
```

```
for(i=0; i<n-1; i++)
```

```
{
```

```
min=a[i]; n_min=i; // minimal qiymatni qidirish
```

```
for(j=i+1; j<n; j++)
```

```
if(a[j]<min) {min=a[j]; n_min=j;}
```

```
a[n_min]=a[i]; // almashtirish
```

```
a[i]=min;
```

}

Elementlar juftlari oxirgisidan boshlab qiyoslanadi va o'rin almashinadi. Natijada massivning eng kichik elementi uning eng chapki elementiga aylanadi. Jarayon massivning qolgan elementlari bilan davom yettiriladi.

2 . Mavzu bo'yicha sodda dasturlar tuzish.

Topshiriqlar. Quyidagi variantlarda berilgan massivlarga oid misollar dasturini tuzing.

№	Variantlar	№	Kirish	Chiqish
1	Bir o'lchamli sonli massivning o'rtacha qiymatidan kichik elementlarining o'rtacha qiymati hisoblansin.	1	6 58 22 17 84 50 53	19.50
		2	13 8 37 42 64 16 7 40 12 13 21 24 11 8	12.00
2	Bir o'lchamli sonli massiv [a,b] qismidagi elementlari massivning eng kichik elementiga bo'lib chiqilsin qolganlari o'zgartirishsiz qoldirilsin.	1	4 44 99 55 12 1 3	3.7 8.3 4.6 12.0
		2	2 15 8 1 1	1.9 8.0
3	Bir o'lchamli sonli massiv k – elementidan l – elementigacha bo'lgan elementlarining o'rtacha qiymati hisoblansin.	1	3 38 39 41 1 2	38.5
		2	10 6 93 73 62 26 28 65 74 53 59 8 9	63.5
4	Bir o'lchamli sonli massivning minimum elementini massivning oxirgi elementi bilan o'rin almashtirilsin.	1	4 74 0 1 33	74 33 1 0
		2	7 8 37 42 64 16 7 40	8 37 42 64 16 40 7
5	Bir o'lchamli sonli massiv [a,b] qismda yotmaydigan elementlarining o'rtacha qiymati hisoblansin.	1	6 51 49 9 76 56 78 1 4	67.00
		2	17 60 66 34 23 40 68 4 31 36 86 61 59 84 10 28 1 30 11 13	36.93
6	Bir o'lchamli sonli massiv elementlari kvadratlarning yig'indisi hisoblansin.	1	5 24 50 72 96 95	26501
		2	1 43	1849
7	Bir o'lchamli sonli massivning barcha elementlari massiv eng katta elementiga bo'lib chiqilsin.	1	9 37 23 51 47 12 23 90 85 98	0.38 0.23 0.52 0.48 0.12 0.23 0.92 0.87 1.00
		2	3 60 19 27	1.00 0.32 0.45
8	Bir o'lchamli sonli massivni barcha elementlari massiv eng kichik elementiga bo'lib chiqilsin.	1	9 69 48 58 45 57 55 88 89 85	1.53 1.07 1.29 1.00 1.27 1.22 1.96 1.98 1.89
		2	3 4 84 79	1.00 21.00 19.75
9	Bir o'lchamli sonli massiv M dan kattta elementlarini ko'paytmalarini logarifmi hisoblansin.	1	5 26 18 32 68 81 6	18.23
		2	10 49 39 42 12 53 35 94 21 35 12 12	29.96
10	Bir o'lchamli sonli massiv K yoki M ga teng elementlari ko'paytmasi	1	5 7 11 83 18 31 31 3	31

15-amaliy mashg'ulot. C++ algoritmik tilning bir o'lchamli va ko'p o'lchamli massivlari.

Ishdan maqsad: C++ algoritmik tilning bir o'lchamli va ko'p o'lchamli massivlari bilan ishlash.

Vazifa.

- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Mavzu bo'yicha sodda dasturlar tuzish.

1 . Berilgan nazariy materiallarni o'rganish.

Bir o'lchovli massivlarni e'lon qilish va ular ustida amallar. Xotirada ketma-ket (regulyar) joylashgan bir xil turdagi qiymatlarga massiv deyiladi. Odatda massivlarga zarurat, katta hajmdagi, lekin cheklangan miqdordagi va tartiblangan qiymatlarni qayta ishlash bilan bog'liq masalalarni yechishda yuzaga keladi. Faraz qilaylik, talabalar guruhining reyting ballari bilan ishlash masalasi qo'yilgan. Unda guruhning o'rtacha reytingini aniqlash, reytinglarni kamayishi bo'yicha tartiblash, konkret talabaning reytingi haqida ma'lumot berish va boshqa masala ostilarini yechish zarur bo'lsin. Qayd etilgan masalalarni yechish uchun berilganlarning (reytinglarning) tartiblangan ketma-ketligi zarur bo'ladi. Bu yerda tartiblanganlik ma'nosi shundaki, ketma-ketlikning har bir qiymati o'z o'rniga ega bo'ladi (birinchi talabaning reytingi massivda birinchi o'rinda, ikkinchi talabaniki - ikkinchi o'rinda va hakoza). Berilganlar ketmaketligini ikki xil usulda hosil qilish mumkin. Birinchi yo'l - har bir reyting uchun alohida o'zgaruvchi aniqlash: $Reyting_1, \dots, Reyting_N$.

Lekin, guruhdagi talabalar soni yetarlicha katta bo'lganda, bu o'zgaruvchilar qatnashgan programmani tuzish katta qiyinchiliklarni yuzaga keltiradi. Ikkinchi yo'l - berilganlar ketma-ketligini yagona nom bilan aniqlab, uning qiymatlariga murojaatni, shu qiymatlarning ketma-ketlikda joylashgan o'rnining nomeri (indeksi) orqali amalga oshirishdir. Reytinglar ketma-ketligini $Reyting$ deb nomlab, undagi qiymatlariga $Reyting_1, \dots, Reyting_N$ ko'rinishida murojaat qilish mumkin. Odatda berilganlarning bunday ko'rinishiga massivlar deyiladi. Massivlarni matematikadagi sonlar vektoriga o'xshatish mumkin, chunki vektor ham o'zining individual nomiga ega va u fiksirlangan miqdordagi bir turdagi qiymatlardan - sonlardan iboratdir.

Demak, massiv - bu fiksirlangan miqdordagi ayrim qiymatlarning (massiv elementlarining) tartiblangan majmuasidir. Barcha elementlar bir xil turda bo'lishi kerak va bu tur element turi yoki massiv uchun tayanch tur deb nomlanadi. Yuqoridagi keltirilgan misolda $Reyting$ - haqiqiy turdagi vektor deb nomlanadi.

Programmada ishlatiladigan har bir konkret massiv o'zining individual nomiga ega bo'lishi kerak. Bu nomni to'liq o'zgaruvchi deyiladi, chunki uning qiymati massivning o'zi bo'ladi. Massivning har bir elementi massiv nomi, hamda kvadrat qavsga olingan va element selektori deb nomlanuvchi indeksni ko'rsatish orqali oshkor

ravishda belgilanadi. Murojaat sintaksisi:

$\langle \text{massiv nomi} \rangle [\langle \text{indeks} \rangle]$

Bu ko'rinishga xususiy o'zgaruvchi deyiladi, chunki uning qiymati massivning alohida elementidir. Bizning misolda Reyting massivining alohida komponentalariga Reyting[1],...,Reyting[N] xususiy o'zgaruvchilar orqali murojaat qilish mumkin. Boshqacha bu o'zgaruvchilar indeksli o'zgaruvchilar deyiladi. Massiv indeksi sifatida butun son qo'llaniladi. Umuman olganda indeks sifatida butun son qiymatini qabul qiladigan ixtiyoriy ifoda ishlatilishi mumkin va uning qiymati massiv elementi nomerini aniqlaydi. Ifoda sifatida o'zgaruvchi ham olinishi mumkinki, o'zgaruvchining qiymati o'zgarishi bilan murojaat qilinayotgan massiv elementini aniqlovchi indeks ham o'zgaradi.

Shunday qilib, programmadagi bitta indeksli o'zgaruvchi orqali massivning barcha elementlarini belgilash (aniqlash) mumkin bo'ladi. Masalan, Reyting[I] o'zgaruvchisi orqali I o'zgaruvchining qiymatiga bog'liq ravishda Reyting massivining ixtiyoriy elementiga murojaat qilish mavjud.

Haqiqiy turdagi (float, double) qiymatlar to'plami cheksiz bo'lganligi sababli ular indeks sifatida ishlatilmaydi. C++ tilida indeks doimo 0 dan boshlanadi va uning eng katta qiymati massiv e'lonidagi uzunlikdan bittaga kam bo'ladi.

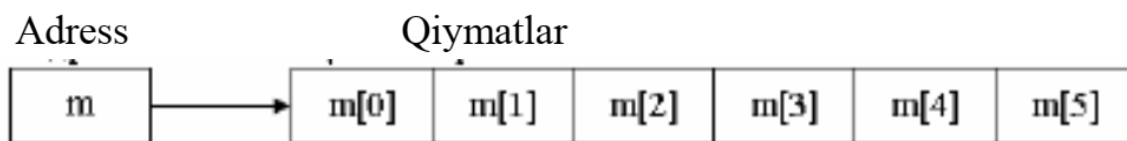
Massiv e'loni quyidagicha bo'ladi:

$\langle \text{tur} \rangle \langle \text{nom} \rangle [\langle \text{uzunlik} \rangle] = \{ \text{boshlang'ich qiymatlar} \}.$

Bu yerda $\langle \text{uzunlik} \rangle$ - o'zgarmas ifoda. Misollar:

int m[6]={1,4,-5,2,10,3}; float a[4];

Massiv statik va dinamik bo'lishi mumkin. Statik massivning uzunligi oldindan ma'lum bo'lib, u xotirada ma'lum adresdan boshlanib ketma-ket joylashadi. Dinamik massivni uzunligi programma bajarilish jarayonida aniqlanib, u dinamik xotiradagi ayni paytda bo'sh bo'lgan adreslarga joylashadi. Masalan, **int m[6];** ko'rinishida e'lon qilingan bir o'lchamli massiv elementlari xotirada quyidagicha joylashadi.



Bir o'lchamli massivning xotirada joylashuvi

Massivning i- elementiga m[i] yoki *(m+i) - vositali murojaat qilish mumkin. Massiv uzunligini sizeof(m) amali orqali aniqladi. Massiv e'lonida uning elementlariga boshlang'ich qiymatlar berish mumkin va uning bir nechta variantlari mavjud.

1) o'lchami ko'rsatilgan massiv elementlarini to'liq inisializasiyalash:

int t[5]={-10,5,15,4,3};

Bunda 5 ta elementdan iborat bo'lgan t nomli butun turdagi bir o'lchamli massiv e'lon qilingan va uning barcha elementlariga

boshlang'ich qiymatlar berilgan. Bu e'lon quyidagi e'lon bilan ekvivalent:

int t[5]; t[0]=-10; t[1]=5; t[2]=15; t[3]=4; t[4]=3;

2) o'lchami ko'rsatilgan massiv elementlarini to'liqmas inisializasiyalash:

int t[5]={-10,5,15};

Bu yerda faqat massiv boshidagi uchta elementga boshlang'ich qiymatlar berilgan. Shuni aytib o'tish kerakki, massivning boshidagi yoki o'rtasidagi elementlariga qiymatlar bermasdan, uning oxiridagi elementlarga boshlang'ich qiymat berish mumkin emas. Agarda massiv elementlariga boshlang'ich qiymat berilmasa, unda kelishuv bo'yicha static va extern modifikatori bilan e'lon qilingan massiv uchun elementlarning qiymati 0 soniga teng deb, automatic massivlar elementlarining boshlang'ich qiymatlari noma'lum hisoblanadi.

3) o'lchami ko'rsatilmagan massiv elementlarini to'liq inisializasiyalash:

int t[]={-10,5,15,4,3};

Bu misolda massivni barcha elementlariga qiymatlar berilgan hisoblanadi, massiv uzunligi kompilyator tomonidan boshlang'ich qiymatlar soniga qarab aniqlanadi. Agarda massiv uzunligi berilmasa, boshlang'ich qiymati berilishi shart.

Massivni e'lon qilishga misollar:

char ch[4]={'a','b','c','d'}; //belgilar massivi

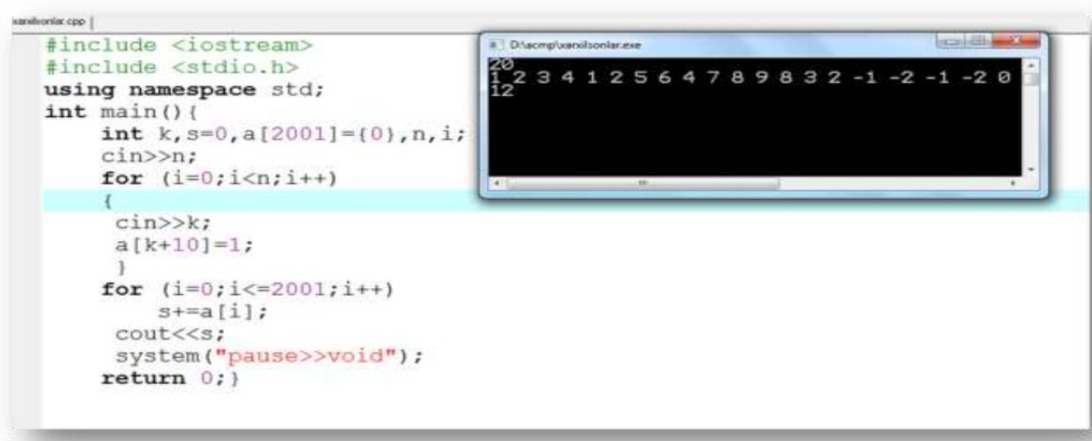
int in[6]={10,20,30,40}; // butun sonlar massivi

char str[]="abcd"; //satr uzunligi 5 ga teng, chunki uning oxiriga '\0' belgisi qo'shiladi

char str[]={ 'a','b','c','d'}; // yuqoridagi satrning boshqacha yozilishi

Masala. Berilgan massiv elementlarining har xil qiymatlilari sonini toping. N – massiv elementlari soni ($1 \leq N \leq 100000$). ($-1000 \leq a[i] \leq 1000$).

Dastur matni:



15.1-rasm. Massiv elementlari sonini topish.

Berilgan masala uchun algoritm. Berilgan masalada massiv elementlari qiymatlari ishoralarini almashtirish va almashtirishlar sonini aniqlash uchun bizga massiv elementlarining nechitasi 0 ga teng ekanligini aniqlash lozim bo'ladi. Chunki 0 qiymatli sonning ishorasi yo'q.

Asosiy dastur matni.

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main() {
    int k,s=0,a[10000],n,i;
    cin>>n;
    for (i=0;i<n;i++)
    { cin>>a[i];
    if (a[i]!=0) {a[i]=-a[i]; s++;}
    }
    for (i=0;i<n;i++)
    cout<<a[i]<<" ";
    cout<<endl<<s;
    system("pause>>void");
    return 0;}

```

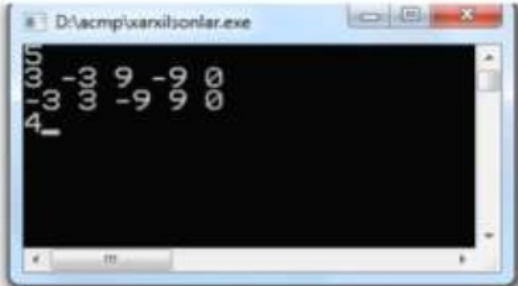
15.2-rasm. Berilgan masala uchun dastur

Berilgan misol ishlanish natijalari tahlili. Berilgan misolda massiv elementlarini kiritish jarayonida ushbu massiv elementlarining qiymati 0 dan farqli bo'lsa uning qiymatini qarama-qarshisiga almashtirilyapti va ushbu almashtirishlar shu joyda sanalayapti.

Natija

```
#include <iostream>
#include <stdio.h>
using namespace std;
int main() {
    int k,s=0,a[10000],n,i;
    cin>>n;
    for (i=0;i<n;i++)
    { cin>>a[i];
    if (a[i]!=0) {a[i]=-a[i]; s++;}
    }
    for (i=0;i<n;i++)
    cout<<a[i]<<" ";
    cout<<endl<<s;
    system("pause>>void");
    return 0;}

```



15.3-rasm. Berilgan dastur natijasi

Massiv chegarasidan tashqariga chiqish (ya'ni mavjud bo'lmagan elementni o'qish/yozishga urinish) dastur bajarilishida kutilmagan natijalarga olib kelishi mumkin. SHuni ta'kidlab o'tamizki, bu eng ko'p tarqalgan xatolardan biridir.

Agar massiv initsializatsiya qilinganda elementlar chegarasi ko'rsatilgan bo'lsa, ro'yxatdagi elementlar soni bu chegaradan kam bo'lishi mumkin, lekin ortiq bo'lishi mumkin emas.

Misol uchun `int a[5]={2,-2}`. Bu holda `a[0]` va `a[1]` qiymatlari aniqlangan bo'lib, mos holda 2 va -2 ga teng. Agar massiv uzunligiga qaraganda kamroq element berilgan bo'lsa, qolgan elementlar 0 hisoblanadi:

```
int a10[10]={1, 2, 3, 4}; //va 6 ta nol
```

Agar nomlangan massivning tavsifida uning o'lchamlari ko'rsatilmagan bo'lsa, kompilyator tomonidan massiv chegarasi avtomatik aniqlanadi:

```
int a3[]={1, 2, 3};
```

Massivda musbat elementlar soni va summasini hisoblash.

```
#include <iostream>
using namespace std;
int main()
{
    int s=0,k=0;
    int x[]={-1,2,5,-4,8,9};
    for(int i=0; i<6; i++)
    {
        if (x[i]<=0) continue;
        k++;
        s+=x[i];
    };
    cout<<k<<endl;
    cout<<s;
    return 0;
};
```

Massivning eng katta, eng kichik elementi va o'rta qiymatini aniqlash:

```
#include <iostream>
using namespace std;
int main()
{
    int i,j,n;
    int min,max,s;
    float a,b,d,x[100];
    while(1)
    {
        cout<<("\n n=");
        cin>>n;
        if ( n>0 && n<=100 ) break;
        cout<<"\n Hato 0<n<101 bo'lishi kerak";
    }
    cout<<"\n elementlar qiymatlarini kiriting:\n";
    for (i=0;i<n;i++)
    {
        cout<<"x["<<i<<"]="";
        cin>>x[i];
    }
}
```

```

}
max=x[0];
min=x[0];
for (s=0,i=0;i<n;i++)
{
s+=x[i];
if (max<x[i]) max=x[i];
if (min>x[i]) min=x[i];
};
s/=n;
cout<<"\n max="<<max;
cout<<"\n min="<<min;
cout<<"\n o'rta qiymat="<<s;
return 0;
}

```

Massivlarni navlarga ajratish. Navlarga ajratish - bu berilgan ko'plab ob'ektlarni biron-bir belgilangan tartibda qaytadan guruhlash jarayoni.

Massivlarning navlarga ajratilishi tez bajarilishiga ko'ra farqlanadi. Navlarga ajratishning $n \cdot n$ ta qiyoslashni talab qilgan oddiy usuli va $n \cdot \log(n)$ ta qiyoslashni talab qilgan tez usuli mavjud. Oddiy usullar navlarga ajratish tamoyillarini tushuntirishda qulay hisoblanadi, chunki sodda va kalta algoritmlarga ega. Murakkablashtirilgan usullar kamroq sonli operatsiyalarni talab qiladi, biroq operatsiyalarning o'zi murakkabroq, shuning uchun uncha katta bo'lmagan massivlar uchun oddiy usullar ko'proq samara beradi.

Oddiy usullar uchta asosiy kategoriyaga bo'linadi:

- oddiy kiritish usuli bilan navlarga ajratish;
- oddiy ajratish usuli bilan navlarga ajratish;
- oddiy almashtirish usuli bilan navlarga ajratish

Oddiy kiritish usuli bilan navlarga ajratish

Massiv elementlari avvaldan tayyor berilgan va dastlabki ketma-ketliklarga bo'linadi. $i=2$ dan boshlab, har bir qadamda dastlabki ketma-ketlikdan i -nchi element chiqarib olinadi hamda tayyor ketma-ketlikning kerakli o'rniga kiritib qo'yiladi. Keyin i bittaga ko'payadi va h.k.

Kerakli joyni izlash jarayonida, ko'proq o'ngdan bitta pozitsiyadan tanlab olingan elementni uzatish amalga oshiriladi, ya'ni tanlab olingan element, $j:=i-1$ dan boshlab, navlarga ajratib bo'lingan qismning navbatdagi elementi bilan qiyoslanadi. Agar tanlab olingan element $a[i]$ dan katta bo'lsa, uni navlarga ajratish qismiga qo'shadilar, aks holda $a[j]$ bitta pozitsiyaga suriladi, tanlab olingan elementni esa navlarga ajratilgan ketma-ketlikning navbatdagi elementi bilan qiyoslaydilar. To'g'ri keladigan joyni qidirish jarayoni ikkita turlicha shart bilan tugallanadi:

- agar $a[j]>a[i]$ elementi topilgan bo'lsa;
- agar tayyor ketma-ketlikning chap uchiga yetilgan bo'lsa.

```

int i, j, x;
for(i=1; i<n; i++)

```

```

{
x=[i];// kiritib qo‘ishimiz lozim bo‘lgan elementni esda saqlab qolamiz
j=i-1;
while(x<a[j]&& j>=0)//to‘g‘ri keladigan joyni qidirish
}
a[j+1]=a[j] /o‘nga surilish
j--;
}
a[j+1]=x;//elementni kiritish
}

```

Massivning minimal elementi tanlanadi hamda massivning birinchi elementi bilan joy almashtiriladi. Keyin jarayon qolgan elementlar bilan takrorlanadi va h.k.

```

int i,min,n_min,j;
for(i=0;i<n-1;i++)
{
min=a[i];n_min=i; //minimal qiymatni qidirish
for(j=i+1;j<n;j++)
if(a[j]<min){min=a[j];n_min=j;}
a[n_min]=a[i];//almashtirish
a[i]=min;
}

```

Elementlar juftlari oxirgisidan boshlab qiyoslanadi va o‘rin almashinadi. Natijada massivning eng kichik elementi uning eng chapki elementiga aylanadi. Jarayon massivning qolgan elementlari bilan davom yettiriladi.

2 . Mavzu bo'yicha sodda dasturlar tuzish.

Topshiriqlar. Quyidagi variantlarda berilgan massivlarga oid misollar dasturini tuzing.

№	Variantlar	№	Kirish	Chiqish
1	Bir o'lchamli sonli massivning o'rtacha qiymatidan kichik elementlarining o'rtacha qiymati hisoblansin.	1	6 58 22 17 84 50 53	19.50
		2	13 8 37 42 64 16 7 40 12 13 21 24 11 8	12.00
2	Bir o'lchamli sonli massiv [a,b] qismidagi elementlari massivning eng kichik elementiga bo'lib chiqilsin qolganlari o'zgartirishsiz qoldirilsin.	1	4 44 99 55 12 1 3	3.7 8.3 4.6 12.0
		2	2 15 8 1 1	1.9 8.0
3	Bir o'lchamli sonli massiv k – elementidan l – elementigacha bo'lgan elementlarining o'rtacha qiymati hisoblansin.	1	3 38 39 41 1 2	38.5
		2	10 6 93 73 62 26 28 65 74 53 59 8 9	63.5
4	Bir o'lchamli sonli massivning minimum elementini massivning oxirgi elementi bilan o'rin almashtirilsin.	1	4 74 0 1 33	74 33 1 0
		2	7 8 37 42 64 16 7 40	8 37 42 64 16 40 7
5	Bir o'lchamli sonli massiv [a,b] qismda yotmaydigan elementlarining o'rtacha qiymati hisoblansin.	1	6 51 49 9 76 56 78 1 4	67.00
		2	17 60 66 34 23 40 68 4 31 36 86 61 59 84 10 28 1 30 11 13	36.93
6	Bir o'lchamli sonli massiv elementlari kvadratlarining yig'indisi hisoblansin.	1	5 24 50 72 96 95	26501
		2	1 43	1849
7	Bir o'lchamli sonli massivning barcha elementlari massiv eng katta elementiga bo'lib chiqilsin.	1	9 37 23 51 47 12 23 90 85 98	0.38 0.23 0.52 0.48 0.12 0.23 0.92 0.87 1.00
		2	3 60 19 27	1.00 0.32 0.45
8	Bir o'lchamli sonli massivni barcha elementlari massiv eng kichik elementiga bo'lib chiqilsin.	1	9 69 48 58 45 57 55 88 89 85	1.53 1.07 1.29 1.00 1.27 1.22 1.96 1.98 1.89
		2	3 4 84 79	1.00 21.00 19.75
9	Bir o'lchamli sonli massiv M dan kattta elementlarini ko'paytmalarini logarifmi hisoblansin.	1	5 26 18 32 68 81 6	18.23
		2	10 49 39 42 12 53 35 94 21 35 12 12	29.96
10	Bir o'lchamli sonli massiv K yoki M ga teng elementlari ko'paytmasi	1	5 7 11 83 18 31 31 3	31

16-amaliy mashg'ulot. C++ algoritmik tilida funksiyalarning grafigini qurish.

Ishdan maqsad: C++ algoritmik tilning bir va ikki o'lchamli funksiyalarning grafigini qurishni o'rganish.

Vazifa.

- 1 . Berilgan nazariy materiallarni o'rganish.
- 2 . Mavzu bo'yicha testlarni bajarish.

1 . Berilgan nazariy materiallarni o'rganish.

Lokal o'zgaruvchilar. Funksiyaga qiymatlar uzatish bilan birga uning tanasida o'zgaruvchilarni e'lon qilish ham mumkin. Bu lokal o'zgaruvchilar orqali amalga oshiriladi. Qachonki dasturni bajarilishi funktsiyadan asosiy qismga qaytsa, bu funktsiyadagi lokal o'zgaruvchilar xotiradan o'chiriladi.

Lokal o'zgaruvchilar xuddi boshqa o'zgaruvchilar kabi aniqlanadi. Funktsiyaga beriladigan parametrlarni ham lokal o'zgaruvchilar deb atash mumkin va ularni funktsiya tanasida aniqlangan o'zgaruvchilar kabi ishlatish mumkin.

Funktsiya lokal o'zgaruvchilari va parametrlarining qo'llanilishi:

```
#include <iostream>
using namespace std;
float Almashtirish(float);
int main()
{
    float TempFer;
    float TempCel;
    cout << "Feringait bo'yicha temperaturani";
    cout << "kiriting:";
    cin >> TempFer;
    TempCel = Almashtirish(TempFer);
    cout << "\n Bu temperatura selziy shkalasi";
    cout << "bo'yicha: ";
    cout << TempCel << endl;
    return 0 ;
}
float Almashtirish(float TempFer)
{
    float TempCel;
    TempCel=((TempFer-32)*5)/9;
    return TempCel;
};
```

Natija:

Feringait bo'yicha temperaturani kiriting: 50

Bu temperatura selziy shkalasi bo'yicha: 10

Global o'zgaruvchilar. main() funktsiyasida aniqlangan o'zgaruvchilar dasturdagi barcha funktsiyalar uchun murojaat qilishga imkonli va ko'rinish

sohasiga ega hisoblanadi. Bunday o'zgaruvchilar dasturdagi funktsiyalar uchun global o'zgaruvchilar deyiladi.

Global o'zgaruvchi nomi bilan funktsiya ichida nomlari ustma-ust tushadigan lokal o'zgaruvchilar faqatgina joriy funktsiyaning ichidagina global o'zgaruvchining qiymatini o'zgartiradi. Lekin global o'zgaruvchi funktsiya o'z ishini tugatgach u chaqirilishidan oldingi qiymatini saqlab qoladi, ya'ni funktsiya tanasida e'lon qilingan lokal o'zgaruvchi funktsiyaning ichida global o'zgaruvchini yashiradi xolos. Bunda lokal o'zgaruvchi alohida hosil qilinadi va funktsiya ishlash vaqtida global va lokal o'zgaruvchilarning nomlari bir xil bo'lsa faqatgina lokal o'zgaruvchi ustida amallar bajariladi. Global o'zgaruvchi esa funktsiyaning bajarilishi davomida oldingi qiymatini saqlab turadi.

Global va lokal o'zgaruvchilarning qo'llanishi

```
#include <iostream>
```

```
using namespace std;
```

```
void MeningFunktsiyam() ; // prototip
```

```
int x = 5, y = 7; // global o'zgaruvchilar
```

```
int main()
```

```
{
```

```
cout << "main()dagi x ning qiymati:" << x << "\n";
```

```
cout << "main()dagi y ning qiymati y:" << y << "\n";
```

```
MeningFunktsiyam();
```

```
cout << "MeningFunktsiyam()funktsiyasi" << "ishini tugatdi! \n";
```

```
cout << "main()dagi x ning qiymati:" << x << "\n";
```

```
cout << "main()dagi y ning qiymati:" << y << "\n";
```

```
return 0;
```

```
}
```

```
void MeningFunktsiyam()
```

```
{
```

```
int y = 10;
```

```
cout << "MeningFunktsiyam()dagi" << "x:" << x << "\n";
```

```
cout << "MeningFunktsiyam()dagi" << "y:" << y << "\n";
```

```
}
```

Hatija:

```
main()dagi x ning qiymati: 5
```

```
main()dagi y ning qiymati: 7
```

```
MeningFunktsiyam()dagi x: 5
```

```
MeningFunktsiyam()dagi y: 10
```

```
MeningFunktsiyam()funktsiyasiishini tugatdi!
```

```
main()dagi x ning qiymati: 5
```

```
main()dagi y ning qiymati: 7
```

Lokal o'zgaruvchilar haqida batafsilroq ma'lumot. Funktsiyani ichida aniqlangan o'zgaruvchilar lokal ko'rinish sohasiga ega deyiladi. Yuqorida aytib o'tilganidek, bu o'zgaruvchilarni faqatgina funktsiyaning ichidagina qo'llash

mumkinligini anglatadi. C++da o'zgaruvchilarni nafaqat dasturning boshida balki ixtiyoriy joyda aniqlash mumkin. Agarda o'zgaruvchi funktsiya tanasidagi biror bir blok ichida aniqlangan bo'lsa, bu o'zgaruvchi faqatgina shu blok ichidagina ta'sirga ega bo'lib butun funktsiyaning ichida ko'rinish sohasiga ega bo'lmaydi.

Funksiyada ishlatiladigan operatorlar. Funksiyada ishlatiladigan operatorlar soni va turiga hech qanday chegara yo'q. Funksiya ichida istalgancha boshqa funktsiyalarni chaqirish mumkin bo'lsada, lekin funktsiyalarni aniqlash mumkin emas.

C++ tilida funktsiyaning hajmiga hech qanday talab qo'yilmasa ham uni kichikroq tarzda tuzgan ma'quldir. Har bir funktsiya tushunish oson bo'ladigan bitta masalani bajarishi lozim. Agarda funktsiya hajmi kattalashayotganligini sezsangiz, bu funktsiyani bir nechta kichikroq funktsiyalarga ajratish haqida o'ylashingiz kerak.

Funktsiya argumentlari. Funktsiyaning argumentlari turli tipda bo'lishi mumkin. SHuningdek, argument sifatida C++ tilidagi biror bir qiymat qaytaradigan o'zgarmaslar, matematik va mantiqiy ifodalar va boshqa ixtiyoriy funktsiyalarni berish mumkin.

Misol sifatida bizda biror bir qiymat qaytaruvchi `double()`, `triple()`, `square()` va `cube()` funktsiyalari berilgan bo'lsin. Biz quyidagi ifodani yozishimiz mumkin:

```
Answer=(double(triple(square(my Value))))
```

Bu ifodada `myValue` o'zgaruvchisini qabul qilinadi va u `cube()` funktsiyasiga argument sifatida uzatiladi.

`cube()` funktsiyasi qaytargan qiymat esa `square()` funktsiyasiga argument sifatida uzatiladi. `square()` funktsiyasi qiymat qaytargandan keyin, buning qiymati o'z navbatida `triple()` funktsiyasiga argument sifatida beriladi. `triple()` funktsiyasining qiymati esa `double()` funktsiyasiga argument qilib beriladi va u `Answer` o'zgaruvchisiga o'zlashtiriladi.

Parametrlar bu lokal o'zgaruvchilardir. Funktsiyaga uzatilgan har bir argument, bu funktsiyaga nisbatan lokal munosabatda bo'ladi. Funktsiyani bajarilishi jarayonida argumentlar ustida bajarilgan o'zgartirishlar funktsiyaga qiymat sifatida berilgan o'zgaruvchilarga ta'sir qilmaydi.

Funktsiyaning qaytaradigan qiymatlari. Funktsiya yo biror bir real qiymatni, yo kompilyatorga hech qanday qiymat qaytarilmasligi haqida xabar beruvchi `void` tipidagi qiymatni qaytaradi.

Funktsiyani qiymat qaytarishi uchun `return` kalitli so'zidan foydalaniladi. Bunda oldin `return` kalitli so'zi, keyin esa qaytariladigan qiymat yoziladi. Qiymat sifatida esa o'zgarmaslar kabi butun bir ifodalarni ham berish mumkin. Masalan:

```
return 5 ;  
return (x > 5) ;  
return (MyFunction()) ;
```

`MyFunction()` funktsiyasi biror bir qiymat qaytarishidan kelib chiqsak, yuqoridagi barcha ifodalar to'g'ri keltirilgan. `return(x>5)` ifodasi esa `x > 5` dan katta bo'lsa `true`, kichik yoki teng bo'lsa `false` mantiqiy qiymatini qaytaradi.

Agarda funktsiyada `return` kalit so'zi uchrasa undan keyingi ifoda bajariladi va uning natijasi funktsiya chaqirilgan joyga uzatiladi. `return` operatori

bajarilgandan keyin dastur funktsiya chaqirilgan satrdan keyingi ifodaga o'tadi. return kalitli so'zidan keyingi funktsiya tanasidagi operatorlar bajarilmaydi.

Grafik biblioteka. Turbo C va Borland C++ kompilyatorlarida grafik biblioteka bilan bog'lanish uchun graphic.h – sarlavxali fayl qo'llaniladi.

Bu bibliotekaga kiruvchi ba'zi grafik funktsiyalar:

void initgraph(int* graphdriver, int* graphmode, char* pathtodriver)- grafik rejimga o'tkazish

void closegraph(void)-grafik rejimdan matnli rejimga o'tkazish.

void putpixel(int x, int y, int color) - Ekranda color rangli(x,y) kordinatali nuqtani tasvirlaydi.

void line (int x1, int y1, int x2, int y2) - Ekranda chiziq chizadi chizadi.

void rectangle (int left, int top, int right, int bottom) - Ekranda to'rtburchak chizadi.

void circle (int x, int y; int radius) - Ekranda aylana chizadi.

void ellipse (int x, int y, int stangle, int endangle, int xradius, int yradius) - Ekranda ellips chizadi.

void outtextxy (int x, int y, char* textstring) – Textni berilgan pozitsiyada chiqaradi.

void outtext (char* textstring) – Textni joriy pozitsiyada chiqaradi.

int getbcolor(void) - Fon rangini qaytaradi

int getcolor(void) - Tasvir rangini qaytaradi.

void getimage (int left, int top, int right, int bottom, void* bitmap) - ekran oynasini xotirada saqlash;

putimage (int left, int top, void* bitmap, int op)- xotirada saqlangan tasvirni ekranga joylash;

Grafik sinfga misol. Misol tariqasida nuqta tushunchasini aniqlovchi point sinfini yaratib point.cpp fayliga yozib qo'yamiz:

```
#ifndef POINTH
#define POINTH 1
class point
{
protected:
int x,y;
public:
point ( int xi=0, int yi=0);
int& givex(void);
int& givey (void);
void show(void);
void move(int xn=0, int yn=0);
private:
void hide();
}
#endif
```

Kelgusida point sinfini boshqa sinflarga qo'shish mumkin bo'lgani uchun shartli protsessor direktivasi `#ifndef POINTH` ishlatilgan. Protsessorli identifikator `POINTH` `#define POINT 1` direktivasi orqali kiritilgan.

SHuning uchun `#include "point.h"` direktivasi bir necha marta ishlatilganda ham `POINT` sinfi ta'riflari faqat bir marta kompilyatsiya qilinaetgan faylda paydo bo'ladi.

`POINT` sinfi komponenta funktsiyalarini quyidagicha ta'riflaymiz:

```

#ifdef POINTCPP
#define POINTCPP 1
#include <graphics.h>
#include "point.h"
point::point(int xi=0, int yi=0)
{
    x=xi; y=yi;
}
int &point::givex(void)
{
    return x;
}
int &point::givey(void)
{
    return y;
}
void point::show(void)
{
    putpixel(x,y, getcolor());
}
void point::hide (void)
{
    putpixel(x,y,getbcolor());
}
void point::move(int xn=0, int yn=0)
{
    hide( );
    x=xn; y=yn;
    show( );
}
#endif
```

Kiritilgan point sinfi va komponenta funktsiyalari bilan ishlovchi dasturni keltiramiz:

```

#include <graphics.h>
#include <conio.h>
#include "point.cpp"
int main()
{
```

```

point A(200,50);
point B;
point D(500,200);
int dr=DETECT, mod;
initgraph(&dr, &mod, "c:\\borlandc\\bgi");
A.show();getch();
B.show();getch();
D.show();getch();
A.move();getch();
B.move(50,60);getch();
closegraph();
return 0;
}

```

Qo'shimcha yuklash. Misol tarikasida '+' amalini point sinfi bilan ta'riflanuvchi displaydagi nuqtalarga qo'llaymiz. Soddashtirish uchun point sinfida eng kerakli komponentalarni qoldiramiz.

```

#include <graphics.h>
#include <conio.h>
class point
{
protected:
int x,y;
public:
point (int xi=0, int yi=0)
{
x=xi; y=yi;
}
void show (void)
{
putpixel (x,y,get color ( ) );
};
point operator+ (point p);
};
point point : : operator + (point &p)
{
point d;
d.x=this.x+p.x;
d.y=this.y+p.y;
return d;
}

int main()
{
int dr=DETECT, mod;
point A(200,50);

```

```

point B;
point D(50,120);
INITGRAPH (&dr, &mod, "C\\borlandc \\ bgi");
A.show ( );
getch ( );
B.show ( ); getch ( );
D.show ( ); getch ( );
B=A+D
;
B.show ( ); getch ( );
B=A.operator + (B);
B.show ( ); getch ( );
closegraph ( );
return 0;
}

```

Dastur bajarilishi natijasida displey ekraniga ketma-ket quyidagi nuqtalar qo'yiladi: A(200,50); B(0,0); D(50,120); B(250,70), B(450,220)

2 . Mavzu bo'yicha testlarni bajarish.

1.C++ tilida necha turdagi leksemalar ishlatiladi?

- A) 5
- B)4
- C)3
- D)6

2.C++ tilida identifikator nima?

- A) arifmetik amallar;
- B) funksiyalar;
- C) lotin harflari,ostki chiziq belgisi va sonlar ketma ketligidan iborat;
- D) o'zgarmas sonlardan iborat;

3. C++ tilida xizmatchi so'zlar deb nimaga aytiladi?

- A) Lotin harflari,ostki chiziq belgisi va sonlar ketma ketligidan iborat ifoda;
- B) Tilda ishlatiluvchi ya'ni dasturchi tomonidan o'zgaruvchilar nomlari sifatida ishlatish mumkin bo'lmagan identifikatorlar ;
- C) C++ kompilyatorida matnlarni formatlovchi maxsus belgilar;
- D)O'zgarmas e'lon qilinishi bilan unga qiymat beruvchi belgilar;

4. C++ tilida maxsus belgilar deb nimaga aytiladi?

- A) Lotin harflari,ostki chiziq belgisi va sonlar ketma ketligidan iborat ifoda;
- B) Tilda ishlatiluvchi ya'ni dasturchi tomonidan o'zgaruvchilar nomlari sifatida ishlatish mumkin bo'lmagan identifikatorlar ;
- C)O'zgarmas e'lon qilinishi bilan unga qiymat beruvchi belgilar;
- D) C++ kompilyatorida matnlarni formatlovchi belgilar;

5.Eskeyp simvollar deb nimaga aytiladi?

- A) '\' simvolidan boshlangan simvollar;
- B) '*' simvolidan boshlangan simvollar;
- C) '=' simvolidan boshlangan simvollar;

D) ';' simvolidan boshlangan simvollar;

6. Ma'lumotlarning haqiqiy son turi necha qismdan iborat bo'lishi mumkin?

A) 3

B) 4

C) 5

D) 6

7. C++ tilida funksiyalarni yozish uchun qanday belgidan foydalaniladi?

A) figurali qavs – "{ "

B) burchak qavs – "<<"

C) to'g'ri burchakli qavs – "["

D) kichik qavs – "("

8. C++ tilida funktsiya tanasi deb nimaga aytiladi?

A) Barcha o'zgarmaslar;

B) Barcha leksemalar;

C) Figurali qavslarni ichida joylashgan barcha satrlar;

D) Barcha operatorlar;

9. Chiqarish operatori ko'rsatilgan qatorni tanlang

A) (<<)

B) (=)

C) (+)

D) (;)

10. C++ tilida qanday o'zgaruvchilar ishlatiladi?

A) cout, cin;

B) tab, bell

C) main

D) int, float, char

**O‘ZBEKISTON RESPUBLIKASI
OLIY VA O‘RTA MAXSUS TA‘LIM VAZIRLIGI**

**ISLOM KARIMOV NOMIDAGI TOSHKENT
DAVLAT
TEXNIKA UNIVERSITETI TERMIZ FILIALI**

**KIMYO INJINERINGIDA
AXBOROT TEXNOLOGIYALARI
o‘quv fanidan**

LABORATORIYA MASHG‘ULOTLARI

1-laboratoriya ishi. Algoritmlar. Asosiy ma'lumot turlari.

Ishdan maqsad: Algoritmlar va uning turlari bilan tanishish, algoritmlar tuzishni o'rganish.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. Masalalarni kompyuterda yechish bosqichlarini o'rganish.
3. Masalalarni kompyuterda yechishning algoritmlash bosqichi.
4. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Maqsad - bosqichlarning qaysi birlarini mutaxassis kompyuterdan foydalanmasdan va qaysi bosqichlarini kompyuterdan foydalanib bajarishini aniqlash hamda bosqichlarni to'la o'rganib chiqishdan iborat.

Kompyuterdan foydalanib "ilmiy – texnik, kimyo sohalaridagi masalalarni yechish tushunchasi keng ma'nodagi so'z bo'lib, u quyidagi bosqichlarga bo'linadi.

Ilmiy – texnik, kimyo sohalaridagi masalalarni yechishda kompyuterdan foydalanish bosqichlari:

1. Masalaning qo'yilishi va maqsadning aniqlanishi;
2. Masalani matematik ifodalash;
3. Masalani yechish uslubini ishlab chiqish, sonli usullarni tanlash;
4. Masalani yechish algoritmini ishlab chiqish;
5. Ma'lumotlarni tayyorlash va tarkibini aniqlash (tanlash);
6. Dasturlash;
7. Dastur matnini va ma'lumotlarni axborot tashuvchiga o'tkazish;
8. Dastur xatolarini tuzatish;
9. Dasturning avtomatik tarzda kompyuterda bajarilishi;
10. Olingan natijalarni izohlash, tahlil qilish va dasturdan foydalanish uchun ko'rsatma yozish.

1- 4 bosqichlar qisqa ma'noda, xususiy holatlar, ko'p uchraydigan murakkab bo'lmagan muammolar uchun tushuntiriladi.

Bu bosqichlar tom ma'noda to'laligicha mutaxassislikni egallash davomida maxsus kurslar vositasida o'rgatiladi.

8- va 9-bosqichlarni bajarishda mutaxassis kompyuterdan foydalanadi.

7-bosqichda kompyuterdan foydalanish ham, foydalanmaslik ham mumkin.

ilmiy – texnik, kimyo va oziq-ovqat sohalaridagi masalalarni kompyuterda yechish bosqichlarini alohida ko'rib chiqamiz.

1-bosqich. Masalaning qo'yilishi va maqsadni aniqlash.

Xalq xo'jaligining muayyan sohasi (texnika, iqtisod, lingvistika, ta'lim va x.k.) bo'yicha ishlayotgan (ishlagan) malakali va etakchi mutaxassis tomonidan bajariladigan ish, masalani qo'yish va maqsadni aniqlash uchun malakali mutaxassis bir necha kun, oy, xattoki yillab izlanishi mumkin.

Qo'yilgan maqsadni amalga oshirish uchun kerakli ma'lumotlar tarkibi

(strukturasi), tuzilishi, ifodalanishi aniqlangan bo'lib, ular orasidagi bog'lanishlar aniq ifodalanagan bo'lsa, "masala qo'yilgan" deb aytiladi.

2-bosqich. Masalani matematik ifodalash.

Bu bosqichda masalani yechish uchun kerakli va etarli bo'lgan dastlabki ma'lumotlarning tarkibi, tavsifi, turi, tuzilishi xosiga olingan xolda matematik terminlarda ifodalanadi xamda masalani yechishning matematik modeli yaratiladi.

Buning uchun xar xil (sohasiga qarab) matematik usul ishlatilishi mumkin.

Masalan iqtisod sohasidagi mutaxassislar - chiziqli dasturlash, dinamik dasturlash, stoxastik dasturlash, bashorat (prognoz) qilish bilan bog'liq masalalarni yechish matematik apparatini bilishlari kerak; texnika sohasidagi mutaxassislar oddiy differentsial tenglamalar va ularning tizimlari, mexanikaning chetki (kraevie) masalalarini, gaz dinamikasiga oid masalalarni, integral ko'rinishdagi masalalarni ifodalash va yechish uchun ishlatiladigan matematik jaryonni to'liq tushunib etgan bo'lishi kerak.

Mutaxassis o'z sohasini har tomonlama yaxshi o'rgangan va amaliy jixatdan puxta o'zlashtirgan va qo'llaniladigan har xil matematik jaryonning barcha imkoniyatlarini to'liq tushunib yetgan va amaliyotga qo'llay oladigan bo'lishi kerak.

Bu bosqichda 2 ta asosiy savolga javob topish kerak:

1. Masalani ifodalash uchun qanday matematik strukturalar maqsadga muvofiq keladi?

2. Yechilgan o'xshash masalalar bormi?

Tanlangan matematik struktura (apparat)da masalaning elementida ob'ektlari to'la ifodalanishi zarur.

3-bosqich. Masalani yechish usulini ishlab chiqish, sonli usulni tanlash.

Agar dastlabki ma'lumotlar bilan izlanayotgan natijalar (miqdorlar, ma'lumotlar) o'rtasida aniq bog'liqlik (qonuniyat) o'rnatilgan bo'lib va masalani yechish uslubi ishlab chiqilgan bo'lsa yoki o'sha bog'lanishni amalga oshirish uchun tayyor sonli usul (lar) tanlab olinib (masala uchun, masalaning bir qismi uchun) masalaning yechish uslubi yaratilgan bo'lsa, "masalani yechish uslubi ishlab chiqilgan" deyiladi.

Bunda: X - dastlabki ma'lumotlar; Y - natija, maqsad funktsiyasi, izlanayotgan miqdor (lar) bo'lsa, ular orasidagi bog'lanish $Y = f(X)$ kabi olinishi mumkin.

f -dastlabki ma'lumotlar bilan natijani bog'lovchi qonuniyat, qoidalar majmuasi, ya'ni X ma'lumotlar ustida bajariladigan amallar ketma-ketligi yoki tanlab olingan usul.

Masalani yechishning ishlab chiqilgan uslubi yoki tanlab olingan usulning to'g'riligi, samaradorligi keyingi bosqichlarda tekshirib aniqlanadi.

4-bosqich. Masalani yechish algoritmini yaratish.

4-bosqichda asosan masalani yechish algoritmi yaratiladi. Masalani yechish algoritmi kompyuterning imkoniyatlarini, yechish aniqligini xamda masalani kompyuterda yechish vaqtini va qiymatini xisobga olgan xolda yaratilsa maqsadga muvofiq kelgan bo'lar edi.

Masalaning algoritmini yaratishda oraliq ma'lumotlarni iloji boricha kamaytirish, tashqi qurilmalar bilan bo'ladigan aloqalarni minimumga keltirish

kerak.

Dasturning samarador va unumdorligi, masalani yechish algoritmining qanchalik puxta tashkil qilinganligiga bog'liq.

3-4 bosqichlar bir-biri bilan jips, mustahkam bog'langan. Ya'ni yaratilgan uslubni har xil usullar bilan amalga oshirish mumkin, shu sababdan masalani yechish uslubi va algoritmining bir nechta variantlari bo'lishi mumkin va keraklisi tanlab olinadi.

Murakkab masalaning algoritmini yaratishda qadamma-qadam oydinlashtirish uslubidan foydalangan ma'qul, har bir qadamda algoritmning tarkibi sodda va tushunarli bo'lib qolishiga erishmoq kerak.

Masalani algoritmlash jarayonida, algoritmning ba'zi bo'laklarini, lavhalarini, mantiqan alohida qismlarini ifodalashda tipik algoritmlar va amaliyotda tekshirilgan algoritmlardan iloji boricha ko'p foydalangan ma'qul.

Algoritmlashda modullik tamoyilidan foydalanish algoritmni o'qishda va dasturlashda qulayliklar yaratadi. Oxir oqibatda masalani yechish algoritmi ishchi holatga keltiriladi, ya'ni algoritm grafik ko'rinishda biror algoritmik til vositasida ifodalash darajasiga keltiriladi.

Masalani algoritmlash - masalani kompyuterdan foydalanib yechish algoritmini yaratish jarayonidir.

Algoritmlash - masalani yechish bosqichi bo'lib, masalaga qo'yilgan shart va talablar asosida oxirgi natijani, masalaning echimini olish uchun ishlab chiqilgan algoritmlarni yaratish bilan shug'ullanadigan informatikaning bo'limidir.

5-bosqich. Ma'lumotlarni tayyorlash va tarkibini aniqlash.

Ma'lumotlarni tasvirlash usulini tanlash algoritmning bajarilishi bilan chambarchas bog'langan. Shu sababdan ma'lumotni tasvirlashning shunday turini, usulini tanlash kerakki, masalani yechish jarayoni sodda va tushunarli bo'lsin.

Ma'lumotlar oddiy o'zgaruvchilar ko'rinishida (bu xol juda kam uchraydi), massiv ko'rinishida, alohida ma'lumot fayllari (ketma-ket o'qiladigan yoki bevosita o'qiladigan) ko'rinishida axborot tashuvchida joylashgan bo'lishi mumkin.

6-bosqich. Dasturlash.

Masalani ishchi xolatga keltirilgan yechish algoritmini tanlangan algoritmik til vositasida ifodalash (tavsiflash, tasvirlash) "dasturlash" deyiladi.

Algoritmning xar bir mayda bo'lagi algoritmik tilning operatorlari yordamida, tilning sintaksis va semantika qoidalari asosida yozib chiqiladi. Algoritm mukammal tuzilgan bo'lsa dasturlashda qiyinchilik tug'ilmaydi. Dasturlash jarayonida quyidagi takliflar inobatga olinsa xatolarni tuzatish jarayoni engillashadi.

1. Dastur umumiy bo'lishi kerak, ya'ni ma'lumotlarni aniq biror turiga bog'liq bo'lmasligi kerak, massivning chegara parametrlarini tekshirmoq lozim. Massiv elementlarining soni 0 yoki 1 bo'lib qolish, yoki yuqori chegarasidan oshib ketish xolati.

2. O'zgarmas kattalik xamda o'zgaruvchi kattalik ko'rinishida ishlatish. (Biror o'zgarmas kattalikni boshqasi bilan almashtirish zarurati bo'lib qolsa, dastur matnini chaqirib o'zgartirish kerak - bu noqulay xolat EXE, COM fayllarida aslo

mumkin emas).

Dasturda kiritiladigan ma'lumotlarni nazorat qilish qismi bo'lishi kerak.

3. Dasturdagi arifmetik amallarni kamaytirish va dasturning ishlashini tezlatish uchun:

- darajaga oshirish amallari ko'paytirish amali bilan almashtirilgani ma'qul;
- bir xil ma'lumot bilan xisoblanayotgan arifmetik (algebraik) ifodalarni bir marta xisoblab qiymatini biror o'zgaruvchida saqlab ishlatish.
- takrorlashlarni tashkil qilishda takrorlanishning chegarasini tekshirish uchun ifodalardan emas balki oddiy o'zgaruvchilardan foydalanish.
- takroriy xisoblashlar tarkibida uchraydigan va takrorlanish davomida qiymatini o'zgartirmaydigan ifodalarni takrorlanishdan tashqarida xisoblash.

4. Dasturning xar bir bo'lagi, moduli qismiga tushuntirishlar yozilgan bo'lishi kerak. Dasturdagi tushuntirishlar, masalani yechish ketma-ketligini ifodalovchi mantiqiy ketma-ketlikdan iborat bo'lmog'i kerak.

Dasturdagi modullar, qismlar aniq ko'rsatilgan bo'lishi kerak. Takrorlanish boshi va takrorlanish oxiri alohida qatorda turgani ma'qul.

7-bosqich. Dastur matnini va ma'lumotlarni axborot tashuvchiga o'tkazish.

Kompyuter uchun axborot tashuvchi vositalar bo'lib: magnitli disk (egiluvchi magnitli disk, magnitli karta), fleshkalar xizmat qilishi mumkin.

Dastur matni alohida maxsus qurilmalar yordamida yoki kompyuterdan foydalanib axborot tashuvchiga o'tkaziladi.

8-bosqich. Dasturning xatosini tuzatish.

Bu bosqich masalani kompyuterda yechish bosqichlari ichidagi ko'p vaqt talab qiladigan, mutaxassisdan sabr, qanoat, chidam, aql, zakovat, mantiqiy tez fikrlash, algoritmik tilning barcha imkoniyatlarini, tuzatish (otladka) qilish uslubini, yo'llarini, masalaning mag'zini iker-chikirlarigacha mukammal bilishni talab qiladigan murakkab izlanuvchan jarayondir.

Bu bosqich "dasturni test bo'yicha tekshirish" deb xam yuritiladi. Dasturning to'g'ri ishlashi va yo'l qo'yilgan xatoliklarni aniqlab tuzatish algoritmini yaratishda yo'l qo'yilgan kamchiliklarni bartaraf qilish xamda tanlangan usulning yaroqli yoki yaroqsiz ekanligini aniqlab beruvchi jarayondir.

Test - maxsus tayyorlangan dastlabki malumotlar bo'lib, ular ustida amallar bajarish bilan masalaning echimi-natija olinadi. Test tayyorlash juda murakkab ish bo'lib, qo'lda hisob-kitob ishlarini bajarishni talab qiladi xamda dasturning xamma qismlarini, bo'laklarini, modullarini tekshirish shart.

Dasturning xatosini tuzatish bo'yicha yo'l - yo'riqlar:

1. Maxsus tayyorlangan ma'lumotlar asosida dasturni qo'lda echib chiqish (imkoni bo'lsa) yoki mantiqan alohida bo'lgan bo'laklarini, modullarini qo'lda xisoblash.

2. Dasturni va uning bo'laklarini, modullarini test yordamida tekshirish.

3. Dasturning kerakli joylariga bosib chiqarish buyrug'ini qo'yish (tuzatishlardan keyin olib tashlanadi).

4. Dasturning xatolarini tuzatishda, muloqot rejimida bajarilganda (STOP) to'xtash buyrug'idan foydalanish.

5. Dasturlash tilini va amal bajaruvchi tizimi (AT)ning maxsus xatolarni

tuzatish imkoniyatlaridan foydalanish.

6. Xatolarni tuzatish jarayonida kam xajmdagi ma'lumotlar bilan ishlashni tashkil qilish.

9-bosqich. Dasturning avtomatik tarzda kompyuterda bajarilishi.

Kompyuter xatolari tuzatilib tayyorlangan dastlabki ma'lumotlardan foydalangan xolda masalaning echimini (echimlarini) avtomatik tarzda xisoblaydi.

Agar natijalar masalaning echimi uchun yaroqli deb topilsa masalani yechish tugallangan xisoblanadi, aks xolda yuqoridagi bosqichlar qaytadan ko'rib chiqiladi.

10-bosqich. Olingan ma'lumotlarni izohlash, tahlil qilish va dasturdan foydalanish uchun yo'riqnoma yozish.

Masalani yechish natijasida olingan sonlar yoki sonlar massivi, matnlar yoki matn ko'rinishidagi massivlar xar taraflama izoxlanib, tushuntiriladi. Dasturdan foydalanish uchun ko'rgazma yozish quyidagilarni o'z ichiga oladi:

- Dastur ishlashi uchun ma'lumotlarni tayyorlash usuli, tuzilishi aniq belgilangan;

- Dasturning ishlashi uchun kompyuterni sozlash yo'llari;

- Dasturni ishga tushirish va ishlash paytida bo'ladigan savol-javoblar;

- Dasturni ishlash jarayonida kelib chiqadigan xar xil xolatlarni bartaraf qilish yo'llari aniq va puxta tushunarli qilib yozilgan bo'lishi kerak.

Masalani yechishning uchta bosqichini quyidagi misollarda ko'rib chiqamiz.

1-MISOL.

1. Masalaning qo'yilishi va maqsadning aniqlanilishi. Koptok 29, 5 m / sek tezlik bilan tepaga tik ravishda tepilgan. U qancha balandlikka ko'tariladi? (Havoning qarshiligi xisobga olinmasin).

2. Masalani matematik ifodalash.

Berilgan: $V_0 = 29, 5 \text{ m / sek.}$; $V = V_0$.

Koptokni balandlikka ko'tarilish xarakatini ifodalovchi qonuniyat:

$h = V_0 \cdot t - g \cdot t^2 / 2$ (1) bu erda: t - koptokning ko'tarilish vaqti, sek. ; g - erkin tushish tezlanishi ($9, 8 \text{ m / sek}$);

3. Masalani yechish usulini ishlab chiqish.

Koptokning tezligi eng yuqori balandlikka yetganda nolga teng bo'ladi:

$V = 0$. Fizika kursidan ma'lumki, tezlik yo'ldan vaqt bo'yicha olingan hosila.

$V = dh / dt$. (2) (1) dan hosila olsak

$V = V_0 - g \cdot t$ (3) (3) -ni nolga tenglab t ning qiymatini topamiz:

$t = V_0 / g$ (4) (4)-dan t ni topib (1) ga qo'yamiz.

2-MISOL.

1. Masalaning qo'yilishi va maqsadni aniqlash.

XOY koordinata tekisligida $Y=0$, $X=a$, $X=b$ to'g'ri chiziqlar va egri chiziq bilan chegaralangan shaklning yuzasi aniqlansin.

2. Masalani matematik ifodalash.

Masalaning qo'yilishidan ma'lumki bu shakl egri chiziqli trapetsiyadir. Uning yuzasini topish aniq integral yordamida quyidagicha hisoblanadi.

3. Masalani yechish usulini ishlab chiqish (tanlash).

Bu turdagi masalalarni yechishda to'rtburchaklar, trapetsiya yoki Simpson taqribiy usullaridan biri tanlab olinadi va yuza hisoblanadi.

2-laboratoriya ishi. Algoritmash asoslari.

Ishdan maqsad: Algoritmalar va uning turlari bilan tanishish, algoritmalar tuzishni o'rganish.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. Masalalarni kompyuterda yechishning algoritmash bosqichi.
3. Algoritm turlari bilan tanishish.
4. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

“Algoritmash” deganda masalani biri ketidan boshqasini bajariladigan xamda oldingisining natijalari keyingilarining bajarilishida ishlatiladigan bosqichlar ketma-ketligiga keltirish tushuniladi. Ayni paytda bu bosqichlardagi amallarni kompyuter bajara olishi ko'zda tutilishi kerak.

Kengroq ma'noda qaraydigan bo'lsak algoritmash, o'zidan oldingi bosqich - masalani yechish usulini tanlash bosqichi ham, o'zidan keyingi bosqich - kompyuterning xususiyatlarini hisobga olgan holda boshlang'ich, oraliq va natijaviy axborotlarni tuzilishining ifoda shakllarini tanlashni ham o'z ichiga oladi.

Algoritmash bosqichining natijasi masalani yechish algoritmi bo'ladi, yani bu bosqichda masalani yechish algoritmi ishlab chiqiladi. Bunda masalani matematik qo'yilishi va tanlangan usul qidirilayotgan natijani olishga olib keladigan xarakterlar ketma-ketligini aniqlash uchun asos bo'lib xizmat qiladi.

“Algoritm” tushunchasi va unga misollar.

Algoritm deb, masalani yechish uchun bajarilishi lozim bo'lgan amallar ketma-ketligini aniq tavsiflaydigan qoidalar tizimiga aytiladi.

Boshqacha aytganda, algoritm – boshlang'ich va oraliq ma'lumotlarni masalani yechish natijasiga aylantiradigan jarayonni bir qiymatli qilib, aniqlab beradigan qoidalarining biror bir chekli ketma-ketligidir.

Buning mohiyati shundan iboratki, agar algoritm ishlab chiqilgan bo'lsa, uni echilayotgan masala bilan tanish bo'lmagan biron bir ijrochiga, shu jumladan kompyuterga xam bajarish uchun topshirsa bo'ladi va u algoritmning qoidalariga aniq rioya qilib masalani echadi.

Masalan, ko'rib o'tilgan birinchi misolni yechish algoritmini quyidagicha bayon qilsa bo'ladi:

- 1) kompyuter xotirasiga V_0 va g o'zgaruvchilarning sonli qiymatlari kiritilsin;
- 2) t ning qiymati $t = V_0 / g$ formula bilan xisoblansin;
- 3) h ning qiymati $h = V_0 t - g t^2 / 2$ (1) formula bilan xisoblansin;
- 4) t va h o'zgaruvchilarning sonli qiymatlari ekranga yoki qog'ozga chiqarilsin;
- 5) hisoblash to'xtatilsin.

Masalaning qo'yilishida koptok 29, 5 m /sek bilan tepilsa, degan shart bor edi. ya'ni, $V_0 = 29, 5$ va $g = 9, 81$ bo'lsa, t va h qancha bo'ladi? (Talabalarning

o'ziga yechish taklif etiladi: $t=3$ sek, $h=43, 35$ m.) Natija xammada bir xil chiqadi.

Ikkinchi misolning yechish algoritmi quyidagicha bo'ladi:

- 1) kompyuter xotirasiga a va b ning qiymati kiritilsin;
- 2) to'g'ri to'rtburchaklar soni n kiritilsin;
- 3) to'rtburchaklar asosi (eni) xisoblansin: $h = (b-a)/n$
- 4) 1-to'rtburchak balandligi (bo'yi) aniqlansin: $x_1 = a$
- 5) 1-to'rtburchak yuzi xisoblansin: $S_1 = \text{sqr}(x_1) * h$
- 6) S_1 ning qiymati eslab qolinsin;
- 7) 2-to'rtburchakka o'tilsin; $x_2 = x_1 + h$ (balandligi shunga bog'liq)
- 8) 2-to'rtburchak yuzi xisoblansin: $S_2 = \text{sqr}(x_2) * h$
- 9) S_2 ning qiymati S_1 ning qiymatiga qo'shib qo'yilsin va yig'indi eslab qolinsin;
- 10) n -to'rtburchakka o'tilsin: $x_n = x_{n-1} + h = b$
- 11) n -to'rtburchak yuzi xisoblansin: $S_n = \text{sqr}(b) * h$
- 12) S_n ning qiymati $S_1, S_2, \dots, S_{n-1}$ lar qiymatiga qo'shilsin;

Algoritmni ishlab chiqish uchun avvalo masalaning yechish yo'lini yaxshi tasavvur qilib olish, keyin esa uni formallashtirish, yani aniq qoidalar ketma-ketligi ko'rinishida yozish kerak.

Algoritmni ishlab chiqishda masalani yechish jarayonini shunday formallashtirish kerakki, bu jarayon etarli darajadagi oddiy qoidalarining chekli ketma-ketligi ko'rinishiga keltirilsin.

Masalan, biz ko'pincha ko'p xonali sonlar ustida asosiy arifmetik amallarni bajarishda vatandoshimiz Al-Xorazmiyning IX asrda yaratgan qoidalarini ishlatamiz. "Algoritm" atamasi xam ana shu buyuk matematik nomidan kelib chiqqan.

Masala yechimining algoritmi ishlab chiqilayotgan davrda asosan uch xil turdagi algoritmlardan foydalanib, murakkab ko'rinishdagi algoritmlar yaratiladi.

Algoritmning asosiy turlariga chizig'li (a), tarmoqlanadigan (b) va takrorlanadigan (c) ko'rinishlari kiradi.

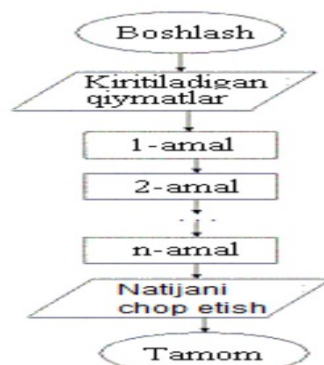
Murakkab masalalarning yechimini olish algoritmlari yuqoridagi turlarining barchasini o'z ichiga olishi mumkin.

Chiziqli turdagi algoritmlarda bloklar biri ketidan boshqasi joylashgan bo'lib, berilgan tartibda bajariladi. Bunday bajarilish tartibi "tabiiy tartib" deb ham yuritiladi.

Har qanday murakkab algoritmni ham uch asosiy struktura yordamida tasvirlash mumkin. Bular ketma-ketlik, ayri va takrorlash strukturalaridir. Ushbu strukturalar asosida chiziqli, tarmoqlanuvchi va takrorlanuvchi hisoblash jarayonlarining algoritmlarini tuzish mumkin. Umuman olganda, algoritmlarni shartli ravishda quyidagi turlarga ajratish mumkin:

- chiziqli algoritmlar;
- tarmoqlanuvchi algoritmlar;
- takrorlanuvchi algoritmlar;
- ichma-ich joylashgan takrorlanuvchi algoritmlar;
- rekurrent algoritmlar;

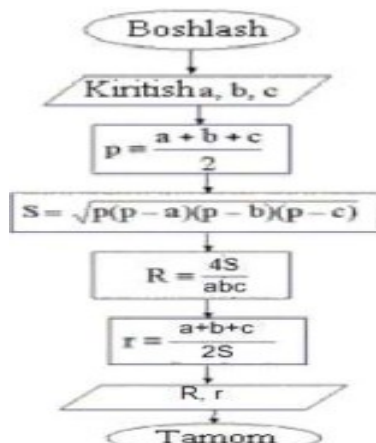
- takrorlanishlar soni oldindan no'malum algoritmlar;
- ketma-ket yaqinlashuvchi algoritmlar.



2.1.-rasm. Chiziqli algoritmlar blok–sxemasining umumiy tuzilishi.

Faqat ketma-ket bajariladigan amallardan tashkil topgan algoritmlarga - *chiziqli* algoritmlar deyiladi. Bunday algoritmnı ifodalash uchun ketma-ketlik strukturasi ishlatiladi. Strukturada bajariladigan amal mos keluvchi shakl bilan ko'rsatiladi. Chiziqli algoritmlar blok-sxemasining umumiy tuzilishi 2.1.-rasmda keltirilgan.

1-misol. Uchburchak tomonlarining uzunligi bilan berilgan. Uchburchakka ichki r va tashqi R chizilgan aylanalar radiuslarini hisoblang. Ichki chizilgan aylana radiusi $r = (a+b+c)/2S$, tashqi chizilgan aylana radiusi $R = \frac{4S}{abc}$ formulalar orqali hisoblanadi. Bu yerda S - uchburchakning yuzi, a, b, c – uchburchak tomonlarining uzunliklari. Masala yechimining blok-sxemasi 2.2.-rasmda keltirilgan.



2.2.-rasm. Uchburchakka ichki va tashqi chizilgan aylanalar radiuslarini hisoblash blok sxemasi

2-misol. Quyida keltirilgan munosabatni hisoblash algoritmini ko'rib chiqaylik. Jarayon amallarni ketma-ket bajarilishidan iborat.

$$z = x^2 + \sqrt{|\sin(x+y)|},$$

bu yerda

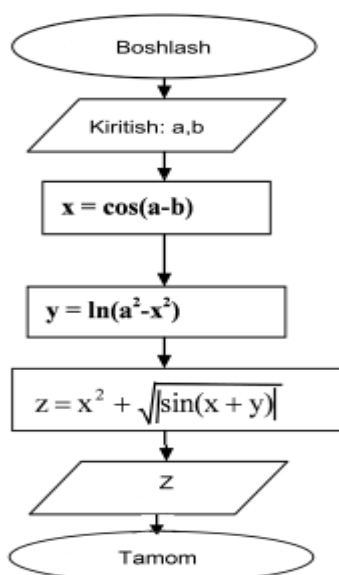
$$x = \cos(a-b), y = \ln(a^2 - x^2), a = 0.7, b = 2.1.$$

Bunda:

a, b - aniq qiymatga ega bo'lgan boshlang'ich ma'lumotlar;

x, y - oraliq ma'lumotlar;
z - natija.

Masalani yechish jarayoni chiziqli hisoblanadi, chunki boshlang'ich ma'lumotlar kiritilgach, munosabatlarning qiymati dasturda joylashgan tartibda hisoblanadi, ya'ni dastlab x, so'ng - y qiymati va nihoyat z natija hisoblanadi. Mazkur jarayonning blok-sxemasini 2.3.-rasmida keltirilgan.



2.3.-rasm. Hisoblash blok-sxemasini

Topshiriq.Quyidagi variantlarda berilgan masalalarning algoritmlarini tuzing.

№	Variantlar
1	Ikkita butun A va B sonlari berilgan. Jumlaning rostlikka tekshirish algoritmini tuzing: A va B sonlarning faqat bittasi toq son".
2	Ikkita butun A va B sonlari berilgan. Jumlaning rostlikka tekshirish algoritmini tuzing "A va B sonlarining har ikkalasi ham yoki toq son yoki juft son".
3	Uchta A, B, C butun sonlar berilgan. Jumlaning rostlikka tekshirish algoritmini tuzing : A, B, C sonlarning har biri musbat".
4	Uchta A, B, C butun sonlar berilgan. Jumlaning rostlikka tekshirish algoritmini tuzing:A, B, C sonlarning hech bo'lmaganda bittasi musbat".
5	Uchta A, B, C butun sonlar berilgan. Jumlaning rostlikka tekshirish algoritmini tuzing:A, B, C sonlaridan faqat bittasi musbat son".
6	Musbat butun son berilgan. Jumlaning rostlikka tekshirish algoritmini tuzing: "Berilgan son ikki xonali juft son".
7	Musbat butun son berilgan. Jumlaning rostlikka tekshirish algoritmini tuzing: "Berilgan son uch xonali toq".
8	Jumlaning rostlikka tekshirish algoritmini tuzing "Berilgan uchta butun sonlarning hech bo'lmaganda 2 tasi bir biriga teng".

9	Jumlani rostlikka tekshirish algoritmini tuzing: "Berilgan uchta butun sonlarning hech bo'lmaganda bir jufti o'zaro qarama qarshi".
10	Uch xonali son berilgan. Jumlani rostlikka tekshirish algoritmini tuzing: "Ushbu sonning barcha raqamlari xar xil".
11	n butun soni berilgan ($n > 0$). Quyidagi yig'indini hisoblovchi algoritm tuzilsin. $S = 1 + 1/2 + 1/3 + \dots + 1/n$
12	n butun soni berilgan ($n > 0$). Quyidagi yig'indini hisoblovchi algoritm tuzilsin. $s = n^2 + (n+1)^2 + (n+2)^2 + \dots + (2n)^2$
13	n butun soni berilgan ($n > 0$). Quyidagi yig'indini hisoblovchi algoritm tuzilsin. $S = 1 - 2 + 3 - \dots$ (n ta qo'shiluvchi, ishoralar almashib keladi.)
14	n butun soni berilgan ($n > 0$). Shu sonning kvadratini quyidagi formula asosida hisoblovchi algoritm tuzilsin. $n^2 = 1 + 3 + 5 + \dots + (2n - 1)$ har bir qo'shiluvchidan keyin natijani ekranga chiqarib boring. Natijda ekranda 1 dan n gacha bo'lgan sonlar kvadrati chiqariladi.
15	n butun soni va a haqiqiy soni berilgan ($n > 0$). a ning n - darajasini aniqlovchi algoritm tuzilsin. $a^n = a * a * \dots * a$;
16	n butun soni va a haqiqiy soni berilgan ($n > 0$). Bir sikldan foydalanib a ning 1 dan n gacha bo'lgan barcha darajalarini chiqaruvchi algoritm tuzilsin.
17	n butun soni va a haqiqiy soni berilgan ($n > 0$). Bir sikldan foydalanib quyidagi a ning 1 dan n gacha bo'lgan barcha darajalarini chiqaruvchi va yig'indini hisoblovchi algoritm tuzilsin. $1 + a + a^2 + a^3 + \dots + a^n$
18	n butun soni va a haqiqiy soni berilgan ($n > 0$). Bir sikldan foydalanib quyidagi a ning 1 dan n gacha bo'lgan barcha darajalarini chiqaruvchi va yig'indini hisoblovchi algoritm tuzilsin. $1 - a + a^2 - a^3 + \dots + (-1)^n a^n$
19	n butun soni berilgan ($n > 0$). Quyidagi ko'paytmani hisoblovchi algoritm tuzilsin. $S = 1.1 * 1.2 * 1.3 * \dots$ (n ta ko'payuvchi)
20	n va k butun son berilgan. k sonini n marta chiqarish algoritmini tuzing.

3-laboratoriya ishi. Tarmoqlanuvchi va takrorlanuvchi hisoblash jarayonini tashkil qilish.

Ishdan maqsad: Tarmoqlanuvchi va takrorlanuvchi algoritmlarni tuzish malakasini shakllantirish.

Berilgan masala uchun uni ishlanish algoritmini va blok-sxemasini tuzish usullarini o'rganish va berilgan misolni ishlash.

Topshiriq:

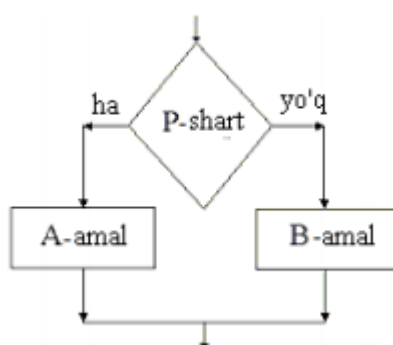
1. Nazariy qism bilan tanishish.
2. Tarmoqlanuvchi hisoblash jarayonini tashkil qilish.

3. Takrorlanuvchi hisoblash jarayonini tashkil qilish
4. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Tarmoqlanuvchi algoritmlar.

Agar hisoblash jarayoni biror-bir berilgan shartning bajarilishiga qarab turli tarmoqlar bo'yicha davom ettirilsa va hisoblash jarayonida har bir tarmoq faqat bir marta bajarilsa, bunday hisoblash jarayonlari tarmoqlanuvchi algoritmlar deyiladi. Tarmoqlanuvchi algoritmlarni tasvirlash uchun "ayri" tuzilmasi ishlatiladi. Tarmoqlanuvchi tuzilmasi berilgan shartning bajarilishiga qarab ko'rsatilgan tarmoqdan faqat bittasining bajarilishi ta'minlanadi (3.1.-rasm).



3.1.-rasm. Tarmoqlanishning umumiy ko'rinishi

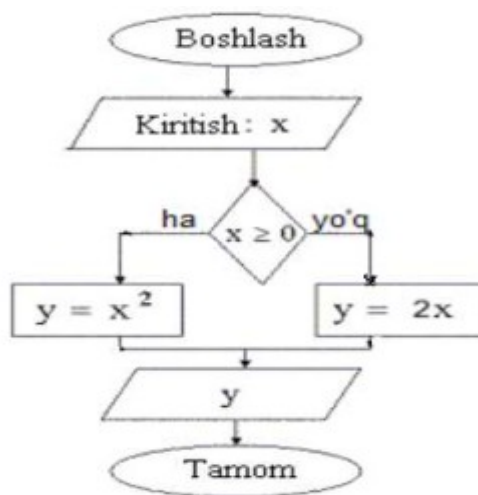
Berilgan R-shart romb figurasi ichida tasvirlanadi. Agar *shart* bajarilsa, "ha" tarmoq bo'yicha A-amal, aks holda (*shart* bajarilmasa) "yo'q" tarmoq bo'yicha V-amal bajariladi.

1-misol. Tarmoqlanuvchi algoritmgaga *misol* sifatida quyidagi sodda *masala* keltiriladi:

$$y = \begin{cases} x^2, & \text{agar } x \geq 0 \\ 2x, & \text{aks holda.} \end{cases}$$

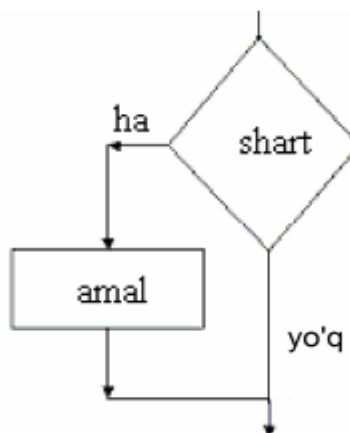
Natijaviy qiymat y berilgan x ning qiymatiga bog'liq holda bo'ladi: agar $x \geq 0$ shart *rost* bo'lsa, tarmoq bo'yicha $y = x^2$ munosabatning qiymati, aks holda, $y = 2 \cdot x$ munosabatning qiymati hisoblanadi. Bu masala bajarilishining so'z bilan ifodalangan algoritmi quyidagicha:

agar ($x \geq 0$) shart bajarilsa, u holda $u = x^2$, aks holda $u = 2 \cdot x$. Masala yechimining blok-sxemasi 3.2.-rasmida keltirilgan.



3.2-rasm. Interval ko‘rinishidagi funktsiya qiymatini hisoblash blok-sxemasi

Ko‘pgina masalalarni yechishda, *shart* asosida tarmoqlanuvchi algoritmnining ikki tarmog‘idan biri, ya’ni «rost» yoki «yolg‘on»ning bajarilishi etarli bo‘ladi. Bu holat tarmoqlanuvchi algoritmnining xususiy holi sifatida qisqartirilgan strukturasi deb atash mumkin. Qisqartirilgan struktura blok-sxemasi quyidagi ko‘rinishga ega (3.3-rasm).



3.3-rasm. Qisqartirilgan strukturaning umumiy ko‘rinishi

2-misol. Berilgan x , u , z sonlari ichidan eng kattasini aniqlang. Ushbu masalaga mos matematik modelni quyidagicha tasvirlash mumkin:

$$p = \max \{x, y, z\}.$$

Berilgan x , y , z sonlardan eng kattasi p deb belgilangan. So‘zlar orqali ifodalangan algoritm asosida masala echimini quyidagicha tasvirlash mumkin:

kiritish (x, y, z) ;

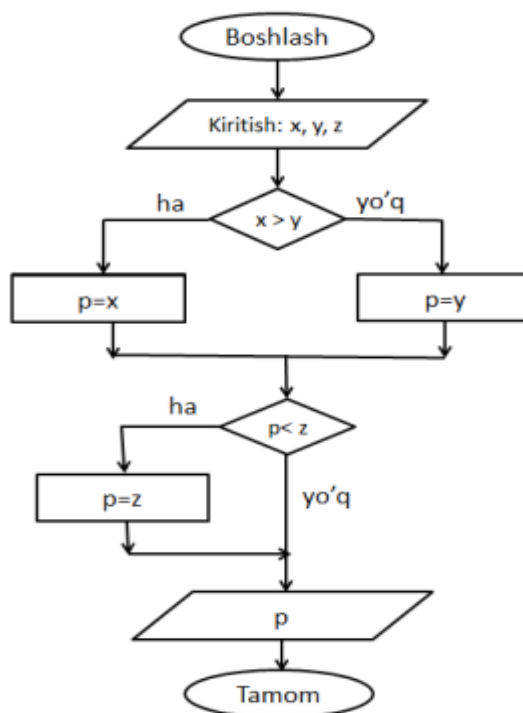
agar $(x > u)$ bo‘lsa, u holda $p = x$, aks holda $p = u$;

agar $(r < z)$ bo‘lsa, u holda $p = z$;

muhrlash (r) .

Keltirilgan algoritmgaga mos blok-sxema 3.4-rasmida tasvirlangan. Bu algoritmda, avval x va y o‘zaro solishtiriladi, katta qiymatligi ega r ga

yuklanadi. Soʻngra x va y larning kattasi deb aniqlangan r va z oʻzaro solishtiriladi. Agar $r < z$ sharti bajarilsa, u holda eng katta qiymat $p = z$ deb olinadi, aks holda boshqarish navbatdagi amalga uzatiladi. Natijada p da uchta qiymatdan eng kattasi aniqlanadi.



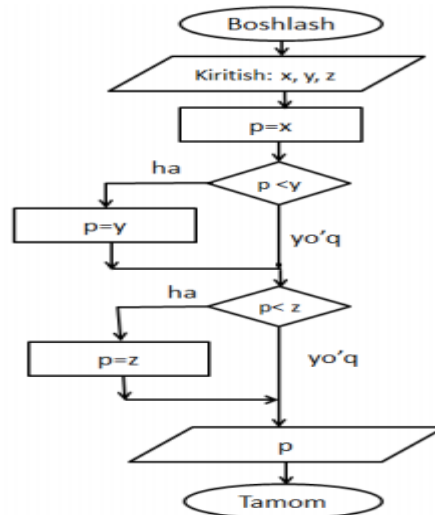
3.4-rasm. Berilgan x , y , z sonlar ichidan eng kattasini topish blok-sxemasi

Ushbu masalani yechish algoritmining yana bir usulini koʻrib chiqamiz.

- 1) kiritish (x , y , z);
- 2) $p = x$;
- 3) agar $(p < y)$ boʻlsa, u holda $p = y$;
- 4) agar $(p < z)$ boʻlsa, u holda $p = z$;
- 5) muhrlash (r).

Bu algoritmgaga mos blok-sxema 3.5-rasmda tasvirlangan.

Bu usulga asosan, avvalo sonlar ichida birinchisi eng kattasi deb faraz qilinadi, yaʼni $p = x$. Soʻngra har bir qadamda navbatdagi son – r ning qiymati bilansolishtiriladi va shart bajarilsa, u eng kattasi deb qabul qilinadi. Bu algoritmning afzalligi shundaki, uning asosida uchta va undan koʻp sonlar ichidan eng kattasini (kichigini) topishning qulay imkoniyati mavjud.



3.5-rasm. Hisoblash blok-sxemasi

3-misol. Quyidagi ifoda bilan berilgan munosabatni hisoblang.

$$Y = \begin{cases} b - x, & \text{agar } x > 0, \\ x + a, & \text{agar } x < 0, \\ a + b, & \text{aks holda.} \end{cases}$$

Bu misol natija x ning qiymatiga bog'liq shart bilan berilgan va masala quyidagicha so'zlar orqali ifodalangan algoritm asosida aniqlanadi:

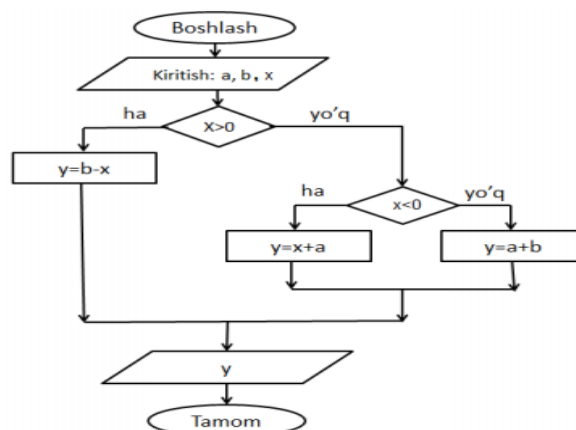
agar $x > 0$ bo'lsa, u holda $u = b - x$ bo'ladi, aks holda;

agar $x < 0$ bo'lsa, u holda $u = x + a$, aks holda $u = a + b$.

Avvalo, birinchi shart tekshiriladi va agar u bajarilsa, $y = b - x$ amal bajariladi, aks holda

$$Y = \begin{cases} x + a, & \text{agar } x < 0, \\ a + b, & \text{aks holda.} \end{cases}$$

Bu fikrlar quyidagi blok-sxemada o'z aksini topgan (3.6-rasm).



3.6-rasm. Hisoblash blok-sxemasi

Takrorlanuvchi algoritmlar.

Agar biror masalani yechish uchun zarur bo'lgan amallar ketma-ketligining ma'lum bir qismi biror parametrga bog'liq holda ko'p marta qayta bajarilsa, bunday jarayon takrorlanuvchi algoritm deyiladi. Takrorlanuvchi algoritmlarga misol sifatida odatda qatorlarning yig'indisi yoki ko'paytmasini hisoblash jarayonlarini qarash mumkin.

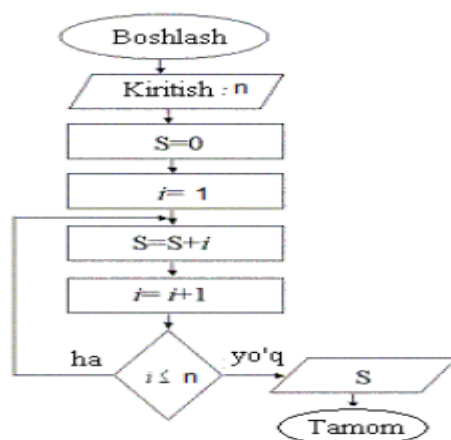
1-misol. Birdan n gacha bo'lgan natural sonlarning yig'indisini hisoblash algoritmini tuzaylik. Masalaning matematik modeli quyidagicha:

$$S = 1 + 2 + 3 + \dots + n = \sum_{i=1}^n i$$

Bu yig'indini hisoblash uchun, avvalo, natija boshlangich qiymatini $S=0$ va indeksning boshlangich qiymatini $i = 1$ deb olamiz va joriy amallar $S = S + i$ va $i = i + 1$ hisoblanadi. Bu erda birinchi va ikkinchi qadamlar uchun yig'indi hisoblandi va keyingi qadamda i parametr yana bittaga orttiriladi va navbatdagi qiymat avvalgi yig'indi S ga qo'shiladi. Mazkur jarayon shu tartibda indeksning joriy qiymati $i \leq n$ sharti bajarilmaguncha davom ettiriladi va natijada, izlangan yig'indiga ega bo'lamiz. Ushbu fikrlarni quyidagi so'zlar orqali ifodalangan algoritm bilan ifodalash mumkin:

- 1) kiritish (n);
- 2) $S = 0$ - natijaning boshlang'ich qiymati;
- 3) $i = 1$ - indeksning boshlang'ich qiymati;
- 4) $S = S + i$ - natijaning joriy qiymatini hisoblang;
- 5) $i = i + 1$ - indeksning joriy qiymatini hisoblang;
- 6) agar $(i \leq n)$ sharti tekshirilsin va u bajarilsa $\Rightarrow$ (4) ;
- 7) muhrlash (S).

Bu jarayonga mos keladigan blok-sxemaning ko'rinishi tasvirlangan



3.7-rasm. 1 dan n -gacha bo'lgan sonlar yig'indisini hisoblash blok-sxemasi

Yuqorida keltirilgan so'zlar asosida ifodalangan algoritm va blok-sxemadan

ko‘rinib turibdiki, amallar ketma-ketligining ma’lum qismi parametr i ga nisbatan n marta takrorlanadi.

2-misol. Quyidagi ko‘paytmani hisoblash algoritmi va blok-sxemasini tuzaylik:

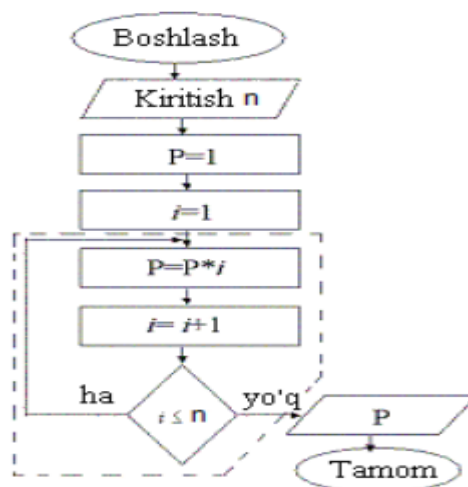
$P = 1 \cdot 2 \cdot 3 \cdots n = n!$ (odatda, 1 dan n gacha bo‘lgan natural sonlarning ko‘paytmasi $n!$ ko‘rinishda belgilanadi va “en” faktorial deb ataladi: $P=n!$ (jarayonning matematik modeli:

$$P = \prod_{i=1}^n i \quad [2, 57-58 \text{ b.}].$$

Ko‘paytmani hosil qilish algoritmi ham yig‘indini hosil qilish algoritmiga o‘xshash, faqat ko‘paytmani hosil qilish uchun, avvalo, $i=1$ da $P=1$ deb olinadi, so‘ngra $i=i+1$ da $P=P \cdot i$ munosabatlar hisoblanadi. Keyingi qadamda i parametrning qiymati yana bittaga orttiriladi va navbatdagi qiymat avvalgi hosil bo‘lgan ko‘paytma - P ga ko‘paytiriladi. Bu jarayon shu tartibda to $i \leq n$ sharti bajarilmaguncha davom ettiriladi va natijaviy ko‘paytmaning qiymatiga ega bo‘lamiz. Quyidagi so‘zlar orqali ifodalangan algoritmda bu fikrlar o‘z aksini topgan:

- 1) kiritish (n);
- 2) $P=1$ - natijaning boshlang‘ich qiymati;
- 3) $i=1$ - indeksning boshlang‘ich qiymati;
- 4) $P=P \cdot i$ - natijaning joriy qiymatini hisoblash;
- 5) $i=i+1$ - indeksning joriy qiymatini hisoblash;
- 6) *agar* ($i \leq n$) shart bajarilsa, u holda $\Rightarrow$ (4);
- 7) muhrlash (P).

Bu algoritmga mos blok-sxema 3.8-rasmda keltirilgan.



3.8-rasm. 1 dan n gacha bo‘lgan sonlar ko‘paytmasini hisoblash blok-sxemasi

Yuqorida ko‘rilgan yig‘indi va ko‘paytmalarning blok-sxemalaridagi takrorlanuvchi qismlariga (punktir chiziqlar ichiga olingan) 3.8-rasmdagi sharti

keyin berilgan takrorlanuvchi struktura mos kelishini ko‘rish mumkin.

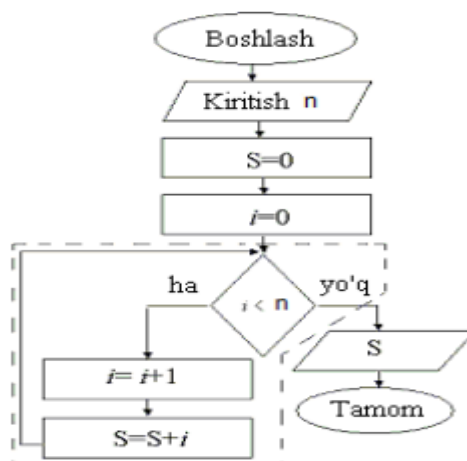
3-misol. Yuqoridagi blok-sxemalarda shartni oldin tekshiriladigan holatda chizish mumkin edi. Masalan,

$$S = \sum_{i=1}^n i$$

yig‘indini xisoblash algoritmi tadqiqi keltiriladi. Bu masalani yechishda algoritmnining takrorlanuvchi qismiga quyidagi sharti oldin berilgan takrorlanuvchi strukturaning mos kelishini ko‘rish mumkin.

- 1) kiritish (n);
- 2) $S=0$;
- 3) $i = 0$;
- 4) agar ($i > n$) $\Rightarrow$ (8);
- 5) $i = i + 1$;
- 6) $S = S + i$;
- 7) shartsiz o‘tish $\Rightarrow$ (4);
- 8) muhrlash (S).

Bu algoritmgaga mos blok-sxema 3.9- rasmda keltirilgan.



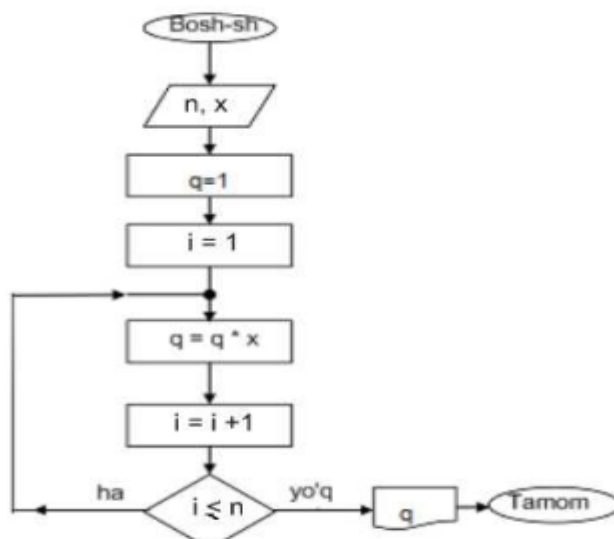
3.9-rasm. 1 dan n gacha bo‘lgan sonlar yig‘indisini hisoblash blok-sxemasi

4-misol. Haqiqiy x sonining n chi darajasini hisoblash masalasi ko‘riladi. Uning matematik modeli: $q = x^n$ ko‘rinishga ega.

Takrorlanuvchi jarayonni tashkil etish quyidagidan farqli, yuqoridagilar bilan bir xil:

- ko‘paytirish jarayoni uchun boshlang‘ich qiymat berilishi: $q = 1$;
- joriy natijani hisoblash: $q = q * x$ ifoda bo‘yicha amalga oshiriladi.

Shunday qilib, x ning n chi darajasini hisoblash uchun takrorlanuvchi jarayonni tashkil etish blok-sxemasi 3.10-rasmda keltirilgan.



3.10-rasm.Hisoblash blok-sxemasi

1. Bir to'g'ri chiziqda yotmaydigan uchta nuqta koordinatlari orqali o'tuvchi aylanani yuzasini hisoblash algoritmini tuzing.
2. "Svetofordan (uchchiroqli) foydalanish" algoritmini tuzing.
3. Geron formulasi bo'yicha uchburchak yuzasini hisoblash algoritmining 1.2 da keltirilgan masalaga oid blok-sxemasini tuzing.
- 4.2. $y=x^2+2x^4+b$ funksiyani $x=3$ qiymatida hisoblash algoritmining blok-sxemasini tuzing.
- 5.Quyidagi ifodani hisoblash algoritmini tuzing:
 $y = (\sin(a^2) + \cos^2(a - b \ln(/)) a + b) + c^2$ bu yerda
 $a = (p + q *) b - c$, $b = \sin(e^2 - (p + q^3))$, $c = \text{tg}(3p + q) - b$.
- 6.Quyidagi ifodani hisoblash algoritmini tuzing:
 $s = (\text{tg}(y^2) + (x - y^3 \ln(/)) x + y^4)$, bu yerda
 $x = (a + b)^2 - (y - c^3)$, $y = e^{(a+b)} - (c^2 - (a - b^3))$.
- 7.Quyidagi ifodalarni hisoblash algoritmini tuzing:
 $z = (a - b)^3 + \lg^2(2a + 3c) / (\text{tg}^2(c - \sin a))$ bu yerda
 $a = \sin((b - c^2))^3$, $b = \text{tg}(x + y^2)$, $c = \cos(e^{(x-y)} + 5\sin(b + 2))$.
- 8.Quyidagi ifodalarni hisoblash algoritmini tuzing:
 $f = \lg(x^2 - y + z^3) + \cos^4(x + y)$ bu yerda
 $x = (\cos(a^3 - b^2) + \sin(e^{a+b})) / (a + \cos(y + 2))$, $y = |a^2 - b|$, $z = (a + |y - x|)^2$
- 9.Haqiqiy a qiymat berilgan. Uchta ko'paytirish amali yordamida a^8 qiymatini hisoblash algoritmini tuzing.
- 10.Berilgan x uchun quyidagi ifodaning qiymatini hisoblash algoritmini tuzing:
 $A = |x - 2| / (2 \cdot x + \sqrt{x} - 2)$.
- 11.Berilgan ikki $(x1, y1)$ va $(x2, y2)$ nuqtalar orasidagi masofani hisoblash algoritmini tuzing.
- 12.Geron formulasi asosida uchlari koordinatalari bilan berilgan uchburchak yuzasini hisoblash algoritmini tuzing.
- 13.Diskreminant hisoblash asosida uchlari koordinatalari bilan berilgan uchburchak yuzasini hisoblash algoritmini tuzing.

14. Teng bo'laklarga bo'lish usuli yordamida $[a,b]$ oralig'ida $f(x) = 0$ tenglamaning yagona ildizini berilgan aniqlik bilan hisoblash algoritmini tuzing.

15. Berilgan 4 ta a, b, c, d haqiqiy sonlar orasidan eng katta va eng kichik qiymatlar o'rtasidagi ayirmani aniqlang.

16. 100 ballik shkala bo'yicha talabalarning to'plagan ballari klaviatura orqali kiritiladi. Ballar taqsimotining qabul qilingan usuliga ko'ra talabalarning ballar bo'yicha baholarini aniqlash dasturini tuzing.

17. Xizmatchining oylik maoshidan o'sib boruvchi shkala bo'yicha ushlab qolinadigan daromad solig'ini hisoblang.

18. Fibonachchi sonlar qatori: $1, 1, 2, 3, 5, \dots$ dan dastlabki n – ta sonlarning yig'indisini aniqlang.

19. Aylananing yuzasi S va kvadratning yuzasi R berilgan. Kvadratning aylanaga sig'ishni yoki sig'masligi aniqlash algoritmini tuzing.

20. Berilgan uchta a, b, s sonlardan foydalanib tomonlarining uzunliklari shu sonlarga teng bo'lgan uchburchakning mavjudligini aniqlang va shunday uchburchakni yasash mumkin bo'lsa, uning yuzasini hisoblash algoritmini tuzing.

4-laboratoriya ishi. MS Excel elektron jadval haqida umumiy tushuncha.

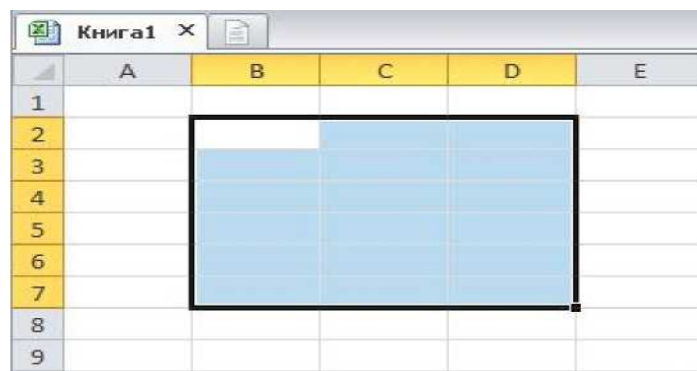
Ishdan maqsad: WINDOWS uchun mo'ljallangan elektron jadvallar bilan ishlovchi EXCEL dasturi bilan ishlashni o'rganish.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. MS Excel elektron jadvalini yuklash.
3. MS Excel elektron jadvali ishchi oynasi va menyulari bilan tanishish.
4. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

MS Excel ishchi varag'ida biror soha sichqoncha chap tugmasi yoki klaviaturadagi maxsus tugmalar yordamida markerlanadi (ajratiladi). Ko'rsatkichni berilgan katakchaga keltirish uchun, sichqonchaning chap tugmasini shu katakcha ustida bosish yoki klaviaturadagi yo'naltiruvchi tugmalar dan foydalanish kerak. Bir nechta katakchalarni markerlash uchun ajratilayotgan soha boshida sichqoncha chap tugmasini bosgan holda soha oxirigacha surish kerak. Natijada to'gri to'rtburchak shaklida soha ajratiladi, ya'ni markerlanadi (uni diapazon yoki blok ham deb yuritiladi).



4.1-rasm

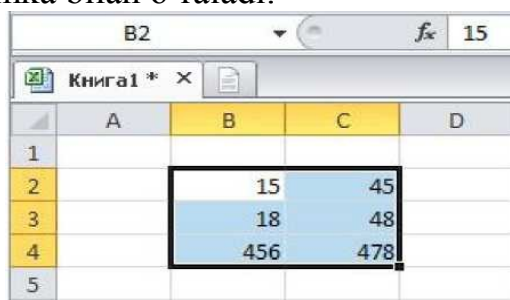
Ajratilgan soha to'g'ri to'rtburchak shaklida bo'lib, uni blok yoki diapazon deb yuritiladi va [A1:B3] yoki (D4:H6) kabi belgilanadi. Bu rasmda [B2:D7] diapazon ajratilgan.

Sohaning markerlanganligini bekor qilish ishchi varaqning ixtiyoriy joyida sichqoncha chap tugmasini bosish kerak. Klaviatura yordamida bir nechta katakchalarni markerlash uchun ko'rsatkichni soha boshiga keltirib **Shift** tugmasini bosgan holda yo'naltirish tugmalaridan foydalanamiz.

To'liq ustunni markerlash uchun ustun nomi ustida, satrni markerlash uchun esa satr nomeri ustida sichqoncha chap tugmasini bosish kerak. Butun ishchi varaqni belgilash uchun esa ustunlar nomlari va satrlar nomarlari joylashgan qatorlar kesishgan joydagi tugmaga bosish kerak.

Ma'lumotlarni nusxalash.

Nusxalash kerak bo'lgan ma'lumotlar joylashgan sohani ajratib Правка tavsiyanomasidan Копировать buyrug'ini (yoki Vosita panelidagi Копировать tugmasi yoki Kontekst menyudagi Копировать bo'limini) tanlash kerak, natijada soha xarakatlanuvchi ramka bilan o'raladi:



4.2- rasm

Endi nusxani joylashtirish lozim bo'lgan joyni sichqoncha yordamida belgilab Правка tavsiyanomasidan Вставить buyrug'ini tanlaymiz. 3.3- rasm da blokka olingan ma'lumot nusxalanib o'rnatilgan holati ko'rsatilgan.

Nusxalangan ma'lumotni o'rnatish uchun yoki teng soha tanlanadi yoki bitta katakcha belgilanadi, bunda katakcha soha boshi deb qaraladi.

	A	B	C	D	E	F
1						
2		15	45		15	45
3		18	48		18	48
4		456	478		456	478
5						

4.3- rasm

Satrlar va ustunlar ustida ish olib borish.

Satrlar va ustunlarni formatlash.

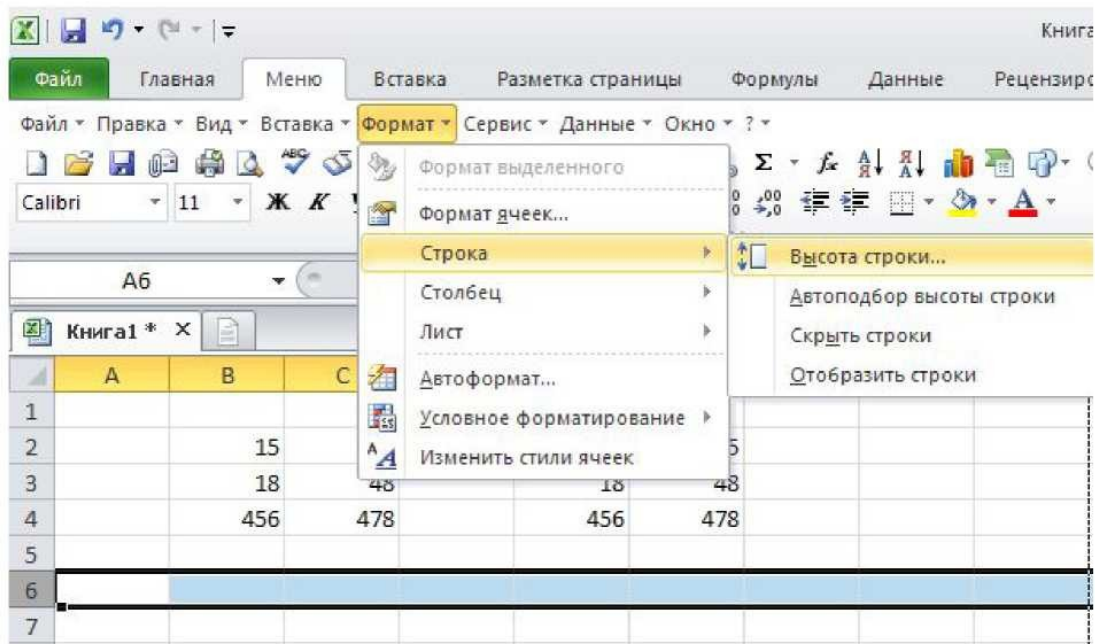
Excel elektron jadvali 1048576 (Excel – 2007da) qator (row) va 16384 ta ustun (column) dan iborat. Qatorlar 1dan 1048576 gacha bo'lgan butun sonlar bilan tartiblangan, ustunlar esa lotin alifbosining bosh harflari (A, B , ... , Z, AA, AB, ..., IV,....XFD) bilan belgilangan.

Satrlarni formatlash uchun kiritilgan ma'lumotlarga asoslangan holda satrlar balandligini o'zgartirish, vaqtinchalik yashirib qo'yish, kerak bo'lganda yashirilgan satrlarni ekranga chiqarish, standart balandlik o'rnatish kabi ishlarni bajarish mumkin.

Undan tashqari belgilangan satrlar ustida ***konteks menyu*** chaqirib satrlar va undagi ma'lumotlarga o'zgarish kiritishga taalluqli bo'lgan buyruqlardan foydalanish mumkin.

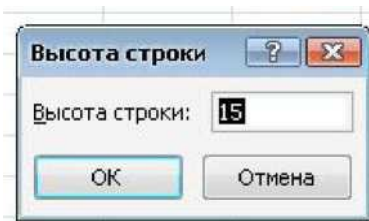
Ustunlar ustida formatlash ishlarini olib borish uchun xam avval ular belgilanib, satrlarni formatlashdagidek ishlarni bajaramiz.

Qatorlarning balandligini va ustunlarning enini o'zgartirish uchun ular belgilangan bo'lishi lozim. Buning uchun kerakli satrlarni belgilab bosh tavsiyanomaning Format bo'limidagi Ctpoka bo'limini tanlaymiz.



6- rasm

Ochilgan bo'lim osti tavsiyanomasidan “Высота”ni tanlaymiz va ochilgan muloqot darchasida kerakli parametrlarni kiritamiz.



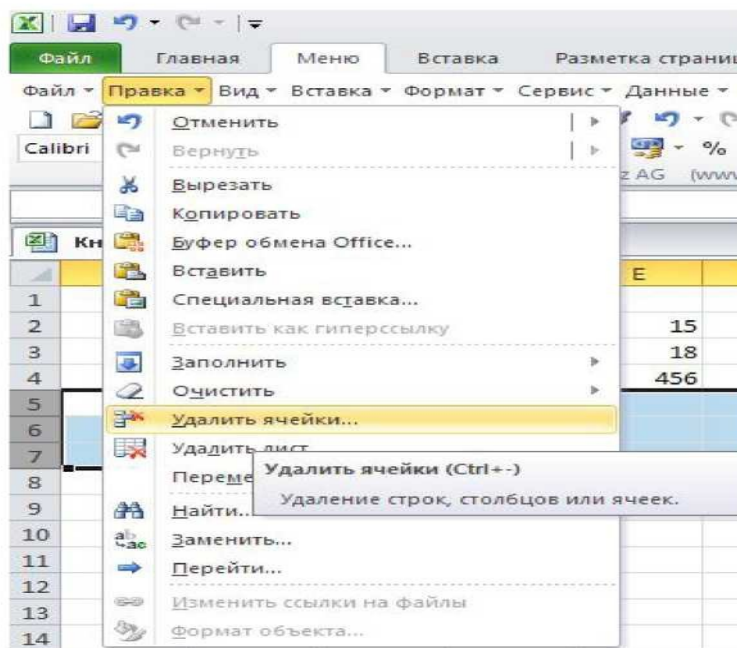
4.5- rasm

Katakchalar, qatorlar va ustunlar qo'yish.

Taxrirlash jarayonida jadvalga katakchalar, qatorlar va ustunlar qo'shish kerak bo'ladi. Satr qo'shish kerak bo'lsa keyingi satr markerlanadi va kontekst menyudan “Добавить ячейки” yoki “Вставка” tavsiyanomasining “Строка” bo'limi tanlanadi. Yangi qator belgilangan qatordan oldinga qo'yiladi. Ustun qo'shish ham xuddi shu tarzda amalga oshiriladi.

Jadvalga katakchalarni qo'shish uchun “Вставка” tavsiyanomasining “Ячейки” bo'limini tanlaymiz. Ochilgan muloqot darchasida katakchalarni qayerga siljitishni ko'rsatamiz.

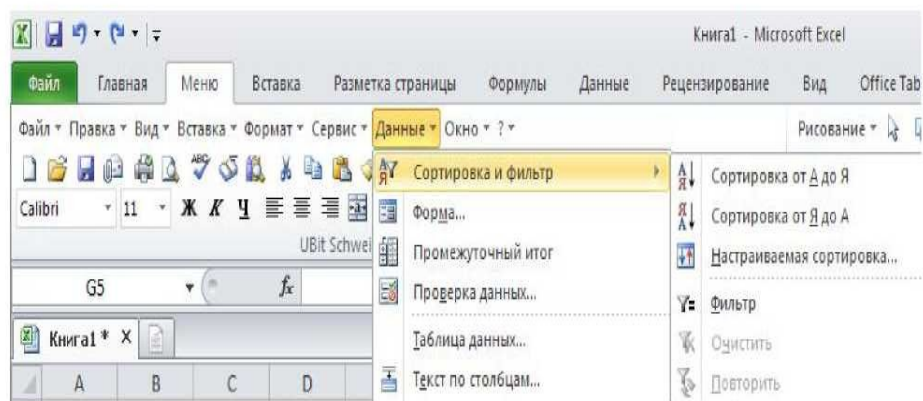
Ortiqcha katakchalarni, qatorlarni va ustunlarni yoqotish (o'chirish) uchun “Правка” menyusidan “Удалить” bo'limini tanlaymiz va ochilgan muloqot darchasida katakchalarni qayerga surishni ko'rsatamiz.



4.6- rasm

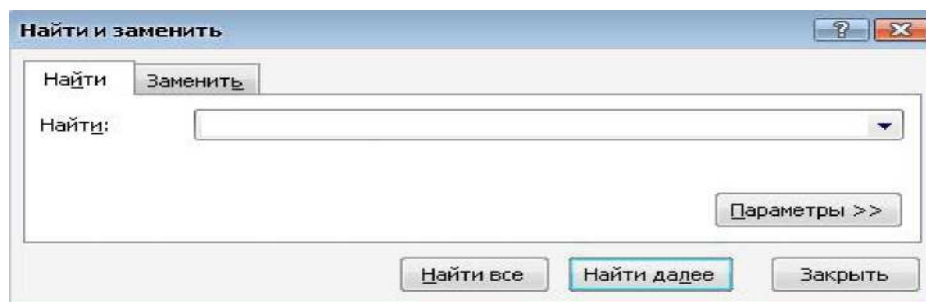
Kiritilgan ma'lumotlar bilan ishlash.

Ma'lumotlarni saralash. Excel da saralash funksiyasi mavjud, unda berilganlarni va matnni alfavit bo'yicha o'sish yoki kamayish tartibida saralash mumkin. Uning uchun saralash kerak bo'lgan sohani belgilaymiz. “Данные” tavsiyanomasining “Сортировка” bo'limini tanlaymiz. Ochilgan muloqot darchasida “По возрастанию”, “По убыванию” hamda boshqa kerakli parametrlarni o'rnatib **ОК**ni bosamiz.



9- rasm

Qidiruv. Katta jadvallar bilan ishlashda adashib berilganlarni yo'qotish mumkin. Ularni qidirish uchun "Правка" tavsiyanomasidagi "Найти" bo'limidan foydalanamiz. "Найти" muloqot darchasida kerakli ifodalar kiritiladi:

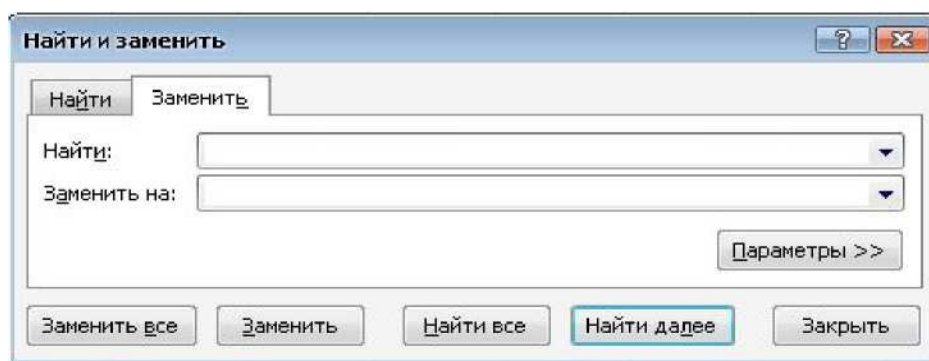


4.8- rasm

- nima qidirmoqchi bo'layapsiz;
- qidiruvning tartibi (oldin qator yoki ustun bo'yicha);
- ishchi varoqning qaysi elementlarini ko'rib chiqish (formulani yoki qiymatini). - qidiruvni boshlash yoki keyingi qiymatni topish.
- qidiruvni to'xtatish.

Qidiruv va o'zgartirish.

Matnda biron bir bo'lakni boshqa matnga almashtirishda заменить buyrug'idan foydalanamiz. Buning uchun "Правка" tavsiyanomaning "Заменить" (4.9- rasm) bo'limini tanlaymiz va bu darchada:



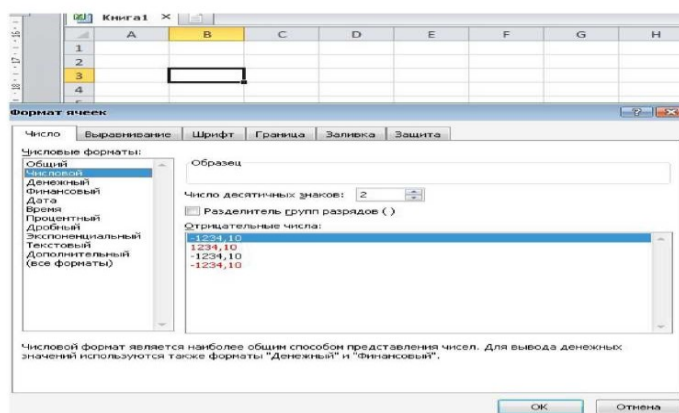
4.9- rasm

Nimani qidiryapsiz, topgan fragmentlarni nimaga o'zgartirmoqchisiz,

qidiruvning tartibi (oldin qatorlar yoki ustunlar bo'yicha), qidiruvni boshlash yoki keyingi qiymatni topish, qidiruvni to'xtatish, topilgan fragmentni o'zgartirish va topilgan va keyingi fragmentlarni o'zgartirish kabi parametrlarni kiritamiz.

Sonlarni taxrirlash.

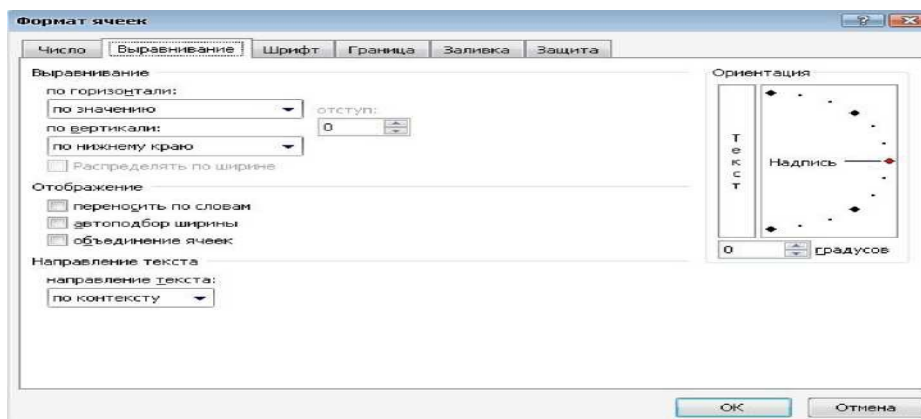
MS Excel ishchi varag'iga kiritilgan ma'lumotlarga moslab katakchalar formatini o'zgartirish uchun avval taxrirlanadigan katakchani belgilab olish kerak. Undan keyin sichqonchanning o'ng tugmasi yordamida kontekst menyuni chaqirib, undan “Формат ячейки” bo'limini tanlaymiz. Natijada ekranda “Формат ячейки” muloqot darchasi ochiladi Bu darchaning chap tomonidan kerakli bo'limni tanlasak darchaning o'ng tomonida tanlangan bo'limga tegishli parametrlar xosil bo'ladi. Masalan, “Числовой” bo'limini tanlasak darchaning o'ng tomonida o'zgarish bo'ladi.



4.10- rasm

Ma'lumotlarni tekislash.

Agar ma'lumotlarni biror ko'rinishda tekislash kerak bo'lsa “Формат ячейки” muloqot darchasining yuqori qismida keltirilgan bo'limlardan “Выравниванием” tanlaymiz va unda kerakli parametrlarni o'rnatamiz.

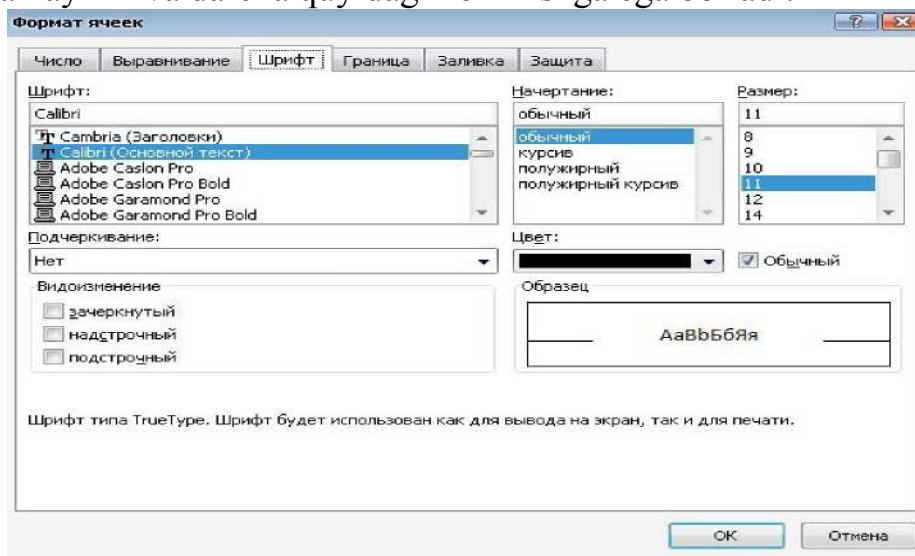


4.11-rasm

Shriftlarni o'rnatish.

Shriftlarni o'rnatish uchun “Формат ячейки” muloqot darchasida “Шрифт”

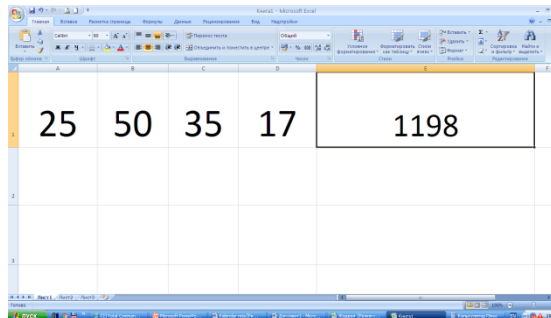
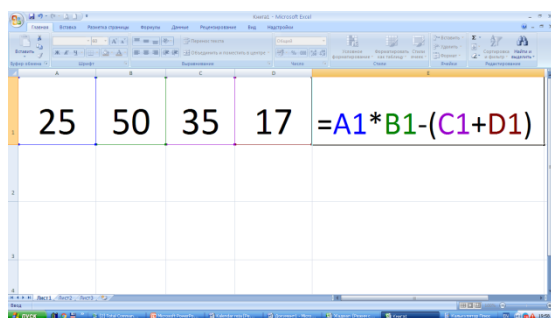
bo'limini tanlaymiz va darcha quyidagi ko'rinishga ega bo'ladi.



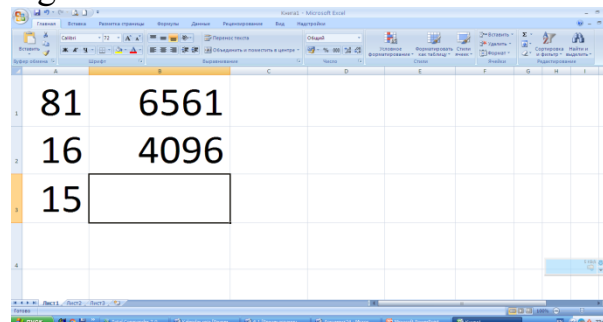
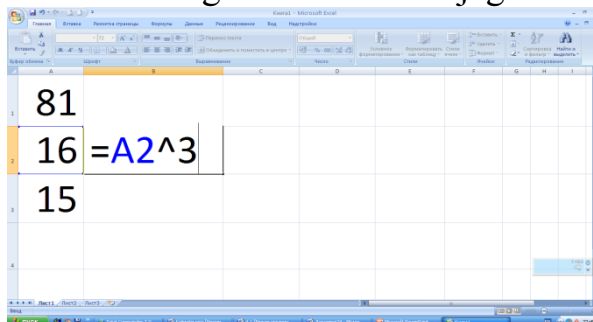
4.12- rasm

Bu muloqot darchasining turli xil ko'rinishlarida kerakli parametrlar o'rnatiladi.

1-misol: bir nechta sonlar ustida amallar bajarishni ko'rib o'tishimiz mumkin:



2-misol. Berilgan sonlarni darajaga ko'taring. $81^2 =$ $16^3 =$ $15^4 =$



Topshiriq. MS Excel elektron jadvaliga ma'lumotlar kiriting.

1. Biror gurux talabalarini familiyasi va ismi, ularning biror oyning har haftasidagi davomati aks ettirilgan jadval hosil qiling. Jadvalda darsga qatnashmagan o'quvchilar "Y", kechikkan o'quvchilar "K" kabi belgilansin hamda katakchalarga darsga qatnashmaganlik va kechikishga mos sabab ko'rsatilgan izoh kiritilsin.
2. gurux jurnaliga mos elektron jadval tuzing. Jadvalda Talabalarining 1-semestr va 2-semestr olgan ballari, o'nlik kasr qismidan faqat 3 xonasi aks etgan o'rtacha ballar va yakuniy ballari aks etsin.
3. A5 katakchada qiymati 50 dan oshmaydigan M musbat butun songa mos

$P=1 \cdot 2 \cdot 3 \cdot \dots \cdot M$ ko'paytma qiymatini barcha raqamlari bilan B1 katakchada aniqlang.

4. A1 katakchada qiymati 100 dan oshmaydigan K musbat butun songa mos $P=1 \cdot 2 + 2 \cdot 3 + 3 \cdot 4 + \dots + (K-1) \cdot K$ yig'indini qiymatini B1 katakchada aniqlang.

5. Burgutning tezligi A m/s, gepardning tezligi V km/soat. Ularning tezliklarini taqqoslab A7 katakchada «Burgutning tezligi katta» yoki «Gepardning tezligi katta» yozuvini aks ettiring.

6. C1:D6 katakchalarga sonlar kiritib ulardan eng kattasini 1- va 2-raqamlari ko'paytmasini A2 katakchada aniqlang.

7. B1:B5 katakchalar blokida sonlar berilgan. Shu sonlardan nechitasi 0 dan farqliligini C3 katakchada aniqlang.

8. x ning -5 qiymatida $y = \left(5 - \sqrt{|x| + 0} \right) \cdot \left(- \frac{1}{2} \right)$ funksiyani hisoblang.

9. И(3>5, 15/2-4>3)= 'yolg'on', ИЛИ(99-27*3=5; 78/2-39>=-1)= 'rost' funksiyalar natijasini aniqlang.

10. ЗАМЕНИТЬ("Yasha"; 5; 1; "na") va ЛЕВСИМБ ("Vatanparvar"; 5) funksiyalari natijasini bitta matnga birlashtib uzunligini aniqlang.

ДЛСТР(ЦЕПИТЬ(ЗАМЕНИТЬ("Yasha"; 5; 1; "na"); ЛЕВСИМБ ("Vatanparvar"; 5)))=11

11. ЗАМЕНИТЬ("Maqsud "; 5; 1; "a"), ПРАВСИМБ("Barkamol"; 5) va ЛЕВСИМБ("topishmoq"; 7) funksiyalari natijasini bitta matnga birlashtiring.

5-laboratoriya ishi. MS Excel dasturi yordamida kimyo injiniringida muammoli vaziyatni yechish.

Ishdan maqsad. MS Excel dasturi yordamida kimyo injiniringida muammoli vaziyatni yechish.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. MS Excel elektron jadvalini yuklash.
3. MS Excel dasturi yordamida kimyo injiniringi masalalarini yechish.
4. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

1 – masala.

Kichik korxonada ishlab chiqarilayotgan non va non mahsulotlari va ularga sarf bo'ladigan mahsulotlarni hisoblash.

1. Masalaning qo'yilishi

Quyidagi jadvalda kichik korxonada ishlab chiqarilayotgan non va non mahsulotlari va ularga sarflangan mahsulotlarni hisoblash uchun ma'lumotlar keltirilgan.

100 kg unga solinadigan mahsulotlar, kg

	Qolipli non oliy nav	Qolipli non 1 nav	Yogʻli patir	Baton oliy nav	Roga-lik	Mayizli bulochka	Buloch-ka	Kunsovo buloch- kasi
	0,6 kg	0,6 kg	0,35 kg	0,4 kg	0,15 kg	0,2 kg	0,1 kg	0,06 kg
Oliy navli un	100,0		100,0	100,0	100,0	100,0	100,0	100,0
1 navli un		100,0						
Tuz	1,5	1,5	2,5	2,5	2,5	2,5	2,5	2,5
Achitqi (drojja)	1,5	1,5	2,0	2,0	2,0	2,0	2,0	2,0
Shakar				4,5	3,5	5,0	10,0	15,0
Yogʻ	0,45	0,45	1,5	1,5	1,5	1,5	1,5	1,5
Mayiz						3,5		
Margarin			15,0	5,0	2,0			

Berilgan ma'lumotlar asosida quyidagilar aniqlansin:

1. Sexda 1 sutkada har 2 soatda ishlab chiqarilgan non va non mahsulotlari hisobga olinsin.
2. Tayyor non va non mahsulotlari sotib oluvchi firma va magazinlarga taqsimlansin.
3. Ishlab chiqarilgan har bir non va non mahsulotlari uchun sarflangan mahsulotlar hisoblansin.
4. Sarflangan mahsulotlar aniqlangandan so'ng kun oxiriga mahsulotlarning qolgan qoldiqlari aniqlansin.

2. Masalani yechishda foydalaniladigan formulalar

Har bir non va non mahsulotlariga ma'lum bir miqdorda turli mahsulotlar solinadi. Masalan: 0,6 kg li oliy navli qolipli non uchun har 100 kg oliy navli unga 1,5 kg tuz, 1,5 kg achitqi (drojja) va 0,45 kg yogʻ ishlatiladi. Uning namligiga qarab, ya'ni unning namligi 11,5 % dan 14,5 % bo'lib, ko'effitsiyent spravochnikdan olinadi. U quyidagi formula asosida topiladi:

$$a = b * s * 100 / d$$

bu yerda: **b** – tayyor non va non mahsulotlari, dona;

c – bir dona non va non mahsulotlarini og'irligi, kg;

d – spravochnikdan olingan ko'effitsiyent.

3. Masalani MS EXCEL dasturida yechish texnologiyasi

Berilgan masalani MS EXCEL dasturida yechish texnologiyasini ko'rib chiqamiz.

1. Birinchi varaqqa «Normativ» deb nom beramiz. A1 yacheykaga «100 kg

unga solinadigan mahsulotlar, kg» matnini kiritamiz, A1:I1 yacheykalarini belgilab, tugmasini bosamiz, natijada A1:I1 yacheykalarining o'rtasida matn joylashadi.

2. Keyingi matnlarni ham mos ravishda quyidagi yacheykalarga kiritib chiqamiz:

V3 yacheykaga – Qolipli non oliy nav

S3 yacheykaga – Qolipli non 1 nav

D3 yacheykaga – Yog‘li patir

E3 yacheykaga – Baton oliy nav

F3 yacheykaga – Rogalik

G3 yacheykaga – Mayizli bulochka

H3 yacheykaga – Bulochka

I3 yacheykaga – Kunsovo bulochkasi

V4 yacheykaga – 0,6 kg

S4 yacheykaga – 0,6 kg

D4 yacheykaga – 0,35 kg

E4 yacheykaga – 0,4 kg

F4 yacheykaga – 0,15 kg

G4 yacheykaga – 0,2 kg

H4 yacheykaga – 0,1 kg

I4 yacheykaga – 0,06 kg

A5 yacheykaga – Oliy navli un

A6 yacheykaga – 1 navli un

A7 yacheykaga – Tuz

A8 yacheykaga – Achitqi (drojja)

A9 yacheykaga – Shakar

A10 yacheykaga – Yog‘

A11 yacheykaga – Mayiz

A12 yacheykaga – Margarin

3. **B5:I12** diapazonidagi yacheykalarga mos ravishda qiymatlarni qo‘yib chiqamiz. Hamma matn va qiymatlarni qo‘yib chiqqandan so‘ng quyidagi ko‘rinishdagi jadval hosil bo‘ladi.

Microsoft Excel - докa-хлеб.xls

	A	B	C	D	E	F	G	H	I
1	100 кг унга ишлатиладиган маҳсулотлар, кг								
2									
3		Қолипли нон олий нав	Қолипли нон 1 нав	Ёғли патир	Батон олий нав	Рогалик	Майизли булочка	Булочка	Кунцово булочкаси
4		0,6 кг	0,6 кг	0,35 кг	0,4 кг	0,15 кг	0,2 кг	0,1 кг	0,06 кг
5	Олий навли ун	100,0		100,0	100,0	100,0	100,0	100,0	100,0
6	1 навли ун		100,0						
7	Туз	1,5	1,5	2,5	2,5	2,5	2,5	2,5	2,5
8	Ачитки (дрожжа)	1,5	1,5	2,0	2,0	2,0	2,0	2,0	2,0
9	Шакар				4,5	3,5	5,0	10,0	15,0
10	Ёғ	0,45	0,45	1,5	1,5	1,5	1,5	1,5	1,5
11	Майиз						3,5		
12	Маргарин			15,0	5,0	2,0			

5.1– rasm. «Normativ» varag‘ining ko‘rinishi

2-varaqqa «Ishlab chiqarish» deb nom berib, yuqoridagi kabi qiymatlarni har bir yacheykaga yozib chiqib, 5.2 – rasm ko‘rinishga keltiramiz.

Microsoft Excel - докa-хлеб.xls

	A	B	C	D	E	F	G	H	I
1	1 кунда цехда ишлаб чиқарилган маҳсулот, донa								
2									
3	1 март 2007 й								
4		Қолипли нон олий нав	Қолипли нон 1 нав	Ёғли патир	Батон олий нав	Рогалик	Майизли булочка	Булочка	Кунцово булочкаси
5		0,6 кг	0,6 кг	0,35 кг	0,4 кг	0,15 кг	0,2 кг	0,1 кг	0,06 кг
6	1-смена								
7	19.00-21.00	50	250	25	50	80	120	250	300
8	21.00-23.00	45	240	60	50	80	120	250	300
9	23.00-01.00	60	230	100	50	80	130	250	300
10	01.00-03.00	30	260	80	50	90	130	250	300
11	03.00-05.00	25	270	75	50	90	140	250	300
12	05.00-07.00	45	300	80	50	100	140	250	300
13	1-смена жами:	255	1550	420	300	520	780	1500	1800
14	2-смена								
15	07.00-09.00	100	400	75	60	100	200	250	300
16	09.00-11.00	80	450	80	60	100	220	250	300
17	11.00-13.00	75	470	90	60	100	220	250	300
18	13.00-15.00	75	500	100	60	140	250	250	300
19	15.00-17.00	80	600	80	60	140	250	250	300
20	17.00-19.00	60	650	75	60	150	240	250	300
21	2-смена жами:	470	3070	500	360	730	1380	1500	1800
22	Жами:	725	4620	920	660	1250	2160	3000	3600
23									

5.2 – rasm. «Ishlab chiqarish» varag‘ining ko‘rinishi

Bu rasmda ko‘rib turganingizdek, 1-varaqdagi «Qolipli non oliy nav» matni 2-varaqda takrorlanmoqda. Yana qaytadan yozishning o‘rniga quyidagi vazifani ham bajarish mumkin. Buning uchun 2-varaqdagi V4 yacheykaga teng (=) belgisini qo‘yib, 1-varaq «Normativ»ga o‘tib, V3 yacheykasi tanlanadi va «Enter» tugmasi bosiladi. Natijada V4 yacheykada «Qolipli non oliy nav» matni hosil bo‘ladi (5.3 – rasm). Qolgan matnlarni ko‘chirish uchun esa V4 yacheykaga kursorni qo‘yamiz va pastki o‘ng burchagiga kursorni olib boramiz, natijada pastki o‘ng burchakda (+) belgisi paydo bo‘ladi. So‘ngra sichqonchaning o‘ng tugmasini bosgan holda I4 yacheykaga siljitamiz. Xuddi shunday qilib, B4:I4 yacheykalarni belgilab, pastga sijitamiz natijada 5.4 – rasmdagi ko‘rinish hosil bo‘ladi.

	A	B	C
1		1 кунда цехда ш	
2			
3			
4		Қолипчи нон олий нав	Қолипчи нон 1 нав
5		0,6 кг	0,6 кг
6			

5.3 – rasm.

	A	B	C	D	E	F	G	H	I
1		1 кунда цехда шиллаб чиқарилган маҳсулот, дон							
2									
3				1 март 2007 й					
4		Қолипчи нон олий нав	Қолипчи нон 1 нав	Ёғли патир	Батон олий нав	Рогалик	Майизли булочка	Булочка	Кунцово булочкаси
5		0,6 кг	0,6 кг	0,35 кг	0,4 кг	0,15 кг	0,2 кг	0,1 кг	0,06 кг
6									

5.4 – rasm.

3-varaqni «**Magazinlarga tarqatish**» deb nomlab, ma'lumotlarni 2-varaqdan yuqorida bajarilgan ketma-ketliklarni bajaramiz va quyidagiga ega bo'lamiz (5.5 – rasm).

	A	B	C	D	E	F	G	H	I
1		Магазин ва фирма магазинларига жўнатилган маҳсулотлар, дон							
2									
3				1 март 2007 й					
4		Қолипчи нон олий нав	Қолипчи нон 1 нав	Ёғли патир	Батон олий нав	Рогалик	Майизли булочка	Булочка	Кунцово булочкаси
5		0,6 кг	0,6 кг	0,35 кг	0,4 кг	0,15 кг	0,2 кг	0,1 кг	0,06 кг
6	Дийр магазини	50	150	25	25	50	125	250	300
7	Роҳат магазини		400	35	25	50	125	250	400
8	Океан магазини	50	350	40	25	50	125	250	200
9	Эшмат ота магазини	30	250	40	25	50	125	250	250
10	Содиқ ота магазини	25	150	40	25	50	125	250	250
11	1 фир. магазин	120	1000	150	150	250	300	500	500
12	2 фир. магазин	150	700	250	150	250	300	500	500
13	3 фир. магазин	300	1620	340	235	500	935	750	1200
14									

5.5 – rasm.

6. 4-varaqni «**Hisobot**» deb nomlab, bu yerda chiqarilgan non va non mahsulotlariga sarflangan mahsulotlarni hisoblaymiz.

Buning uchun «**Normativ**» varag'idagi ma'lumotlarni, ya'ni non va non mahsulotlarini va sarflanayotgan mahsulotlarni ko'chirish yo'li bilan «**Hisobot**» varag'ida hosil qilamiz. So'ngra V5 yacheykada (=) belgisini qo'yib «**Ishlab chiqarish**» varag'idagi V22 yacheykasidagi qiymatni «**Hisobot**» varag'idagi V4 yacheykaga ko'paytirib, natijani 100 ga ko'paytiramiz va ko'paytmani yuqorida aytib ketilgan namlikka asosan spravochnikdan tanlanadigan koeffitsiyentga bo'lamiz.

V5 yacheykada = 'Ishlab chiqarish'!B22 * Hisobot!B4 * 100/114

V7 yacheykada = V5 * Normativ!V7/100

V8 yacheykada = V5 * Normativ!V8/100

V10 yacheykada = V5 * Normativ!V10/100

Xuddi shunday

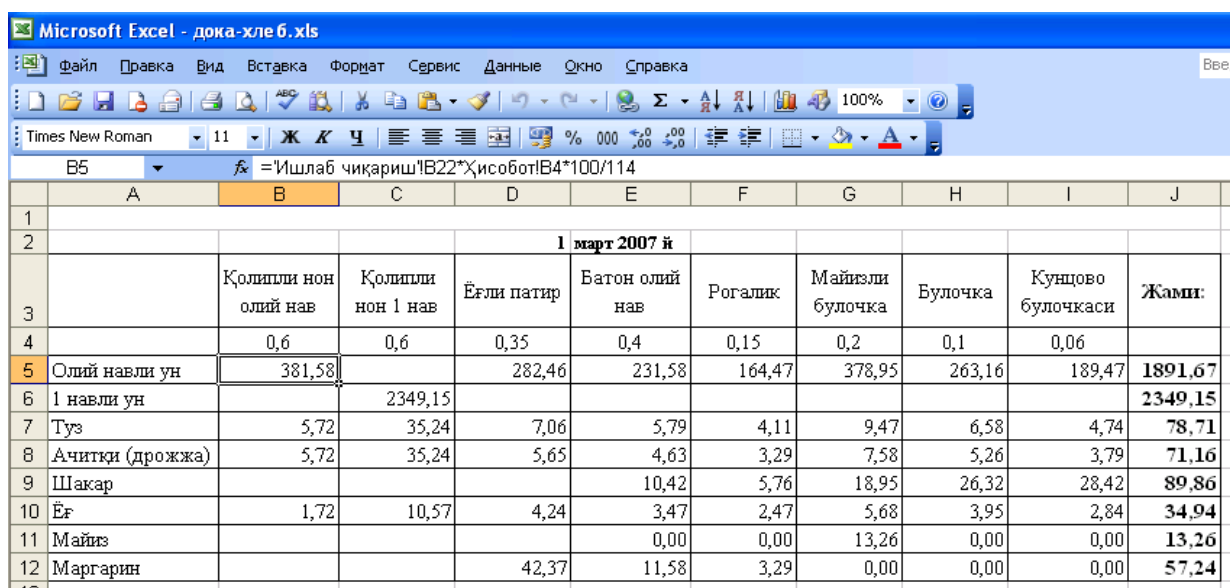
S6 yacheykada = 'Ishlab chiqarish'!S22 * Hisobot!S4 * 100/114

S7 yacheykada = S6 * Normativ!V7/100

$S8 \text{ yacheykada} = S6 * \text{Normativ!V8}/100$

$S10 \text{ yacheykada} = S6 * \text{Normativ!V10}/100$

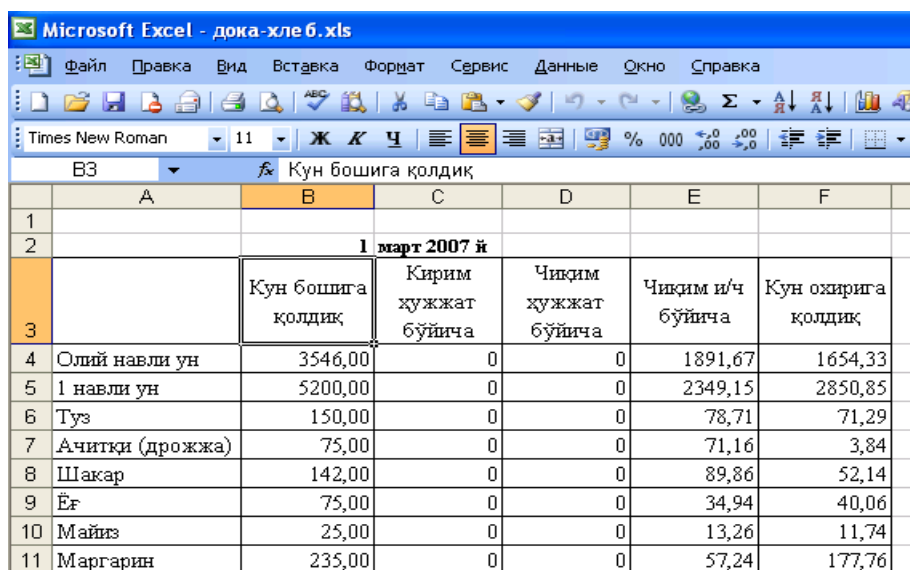
Qolgan non va non mahsulotlari ishlab chiqarish uchun sarflangan mahsulotlarni xuddi shunday qilib aniqlanadi. **J5** yacheykasida esa bir turdagi sarflangan mahsulot-larning yig'indisi topiladi. Buning uchun **J5** yacheykasida (=) belgisini qo'yib uskunalar panelidan (Σ) belgisi bosiladi, natijada =SUMM(V5:I5) paydo bo'ladi va «Enter» tugmasi bosiladi. Qolgan yacheykalarga ham shu formulani tadbiiq qilish uchun **J5** yacheykasiga kursorni olib kelib, pastki o'ng burchakdan sichqonchanning chap tugmasini bosgan holda, kerakli yacheykagacha tortib boriladi va quyidagi rasm ko'rinishda bo'ladi (5.6 – rasm).



	A	B	C	D	E	F	G	H	I	J
1										
2					1 март 2007 й					
3		Қолиқли нон олий нав	Қолиқли нон 1 нав	Ёғли патир	Багон олий нав	Рогалик	Майизли булочка	Булочка	Қунцово булочкаси	Жамп:
4		0,6	0,6	0,35	0,4	0,15	0,2	0,1	0,06	
5	Олий навли ун	381,58		282,46	231,58	164,47	378,95	263,16	189,47	1891,67
6	1 навли ун		2349,15							2349,15
7	Туз	5,72	35,24	7,06	5,79	4,11	9,47	6,58	4,74	78,71
8	Ачитқи (дрожжа)	5,72	35,24	5,65	4,63	3,29	7,58	5,26	3,79	71,16
9	Шакар				10,42	5,76	18,95	26,32	28,42	89,86
10	Ёғ	1,72	10,57	4,24	3,47	2,47	5,68	3,95	2,84	34,94
11	Майиз				0,00	0,00	13,26	0,00	0,00	13,26
12	Маргарин			42,37	11,58	3,29	0,00	0,00	0,00	57,24

5.6 – rasm.

7. 5-varaqni «**Kirim-chiqim**» deb nomlab, quyidagi ko'rinishda jadval tuzamiz . (5.7 – rasm).



	A	B	C	D	E	F
1						
2						
3		Кун бошига қолдиқ	Кирим хужжат бўйича	Чиким хужжат бўйича	Чиким и/ч бўйича	Кун охирига қолдиқ
4	Олий навли ун	3546,00	0	0	1891,67	1654,33
5	1 навли ун	5200,00	0	0	2349,15	2850,85
6	Туз	150,00	0	0	78,71	71,29
7	Ачитқи (дрожжа)	75,00	0	0	71,16	3,84
8	Шакар	142,00	0	0	89,86	52,14
9	Ёғ	75,00	0	0	34,94	40,06
10	Майиз	25,00	0	0	13,26	11,74
11	Маргарин	235,00	0	0	57,24	177,76

5.7 – rasm

Sarflangan mahsulotlarni omborxonadagi hisob-kitoblarni, ya'ni qoldiqlarini hisoblash uchun kun boshiga qoldiqni, ish kuni davomida mahsulotlarni kirimi va chiqimini, so'ngra kun oxiriga qoldiq natijasini ko'rish uchun quyidagi amallar bajariladi.

V4:V11 yacheykalarga kun boshiga qoldiqni yozib chiqamiz.

S4:S11 yacheykalarga kun davomida kirim bo'lganini yozamiz.

D4:D11 yacheykalarga kun davomida chiqim bo'lganini yozamiz.

YE4:YE11 yacheykalarga «**Hisobot**» varag'idagi chiqim qo'yiladi, buning uchun **YE4** yacheykasiga (=) belgisi qo'yiladi, «**Hisobot**» varag'idagi **J5** yacheykasi belgilanib «**Enter**» tugmasi bosiladi. **YE4** yacheykasini belgilab, pastki o'ng burchagiga kursorni olib kelib **YE11** yacheykasigacha tortiladi.

F4 yacheykasiga = **B4+C4-D4-E4** formulasi kiritiladi.

F4 yacheykasini belgilab, pastki o'ng burchagiga kursorni olib kelib **F11** yacheykasigacha tortiladi. Natijada yuqoridagi rasm ko'rinishdagi holatga keladi.

6-laboratoriya ishi. MS Excel elektron jadvali imkoniyatlaridan foydalanib masalalar yechish.

Ishdan maqsad. Berilgan masalalarni MS Excel elektron jadvalida hisoblashni o'rganish va berilgan masalalarni ishlash.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. MS Excel elektron jadvalini yuklash.
3. MS Excel dasturi yordamida diagrammalar hosil qilish.
4. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

1-vazifa. Berilgan arifmetik ifodalarni hisoblang.

A	B
$A_1 = 4 \cdot 10^{-12} \sqrt{ 6-2t } + \log_5 \left[\left(t + \frac{x}{S_{\text{int}}} \right) \right]$	$A_2 = \sin \frac{\alpha}{\beta} + \frac{\beta \cdot (2 - \cdot))}{\sqrt{t + \cos t}} - 4 \operatorname{rctg} \frac{w - \cdot}{w + \cdot}$

A) Excel elektron jadval protsessorida hisoblash ketma-ketligi quyidagicha:

1. Berilgan t va x larga qiymat bering.
2. A₁ qiymatini hisoblash uchun B9 yacheykaga ko'rsatilgan formulani kiritib, Enter tugmasini bosing. B) vazifani ham xuddi shunday bajaring.

	A	B	C	D	E	F	G	H	I
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									
14									
15									
16									
17									
18									
19									
20									
21									
22									
23									
24									
25									

6.1-rasm.

2-vazifa.

$$z = \frac{10ab}{\sqrt{a}(e^2 + 3x^2)} - \log_4 y, \quad y = \frac{\sqrt{x^2 + 3 \cdot a^2}}{b^2 + x^2 + 1}, \quad a = 3,23, \quad b = 33,2, \quad x \in \mathbb{R}$$

Bu vazifani bajarishda arifmetik ifodalarning bog'liqligiga e'tibor qiling. Bog'liq ifodalarni oldin hisoblanadi.

	A	B	C	D	E	F	G	H	I
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									
13									

6.2-rasm

3-vazifa. Arifmetik ifodani Excel mantiqiy funktsiyalaridan foydalanib hisoblash.

$$F = \begin{cases} f_1; a > x \\ f_2; a \leq x < b; f_1 = \frac{x - \frac{1}{2}}{3 + \frac{1}{2}}; \\ f_3; x \geq b \end{cases}$$

$$f_2 = x^3 - 1 + \frac{1}{2}x^2 + 1bx; \quad f_3 = \frac{b - \frac{1}{2}}{5 + 3x^2}; \quad a = 1; \quad b = 2,6; \quad x \in \mathbb{R}$$

Bu vazifani tartibi ham yuqoridagilar kabi bajariladi (arifmetik ifodalarning bog'liqligiga e'tibor qiling) va ESLI funktsiyasining ishlatishini oldin funktsiyalar bo'limidan yaxshilab o'rganib chiqing. Bu funksiya mantiqiy (logicheskie) funktsiyalar kategoriyasiga mansub bo'lib, u umumiy holda quyidagicha yoziladi:

$$= \text{ЕСЛИ}(\text{Mantiqiy ifoda, ya'ni shart; ifoda1; ifoda2})$$

Ifoda1 mantiqiy ifoda rost bo'lganda hisoblanadi, ifoda2 esa yolg'on bo'lganda hisoblanadi.

	A	B	C	D	E	F	G	H	I
1									
2									
3									
4									
5									
6									
7									
8									
9									
10	a=	1							
11	b=	2,6							
12	x=	1,5							
13	f1=	0,095							
14	f2=	-0,82							
15	f3=	0,094							
16	F=	-0,82							
17									

6.3-rasm

4-vazifa. Berilgan yig'indi va ko'paytmalarni hisoblang.

$\sum_{m=1}^{10} \frac{5}{m^2 + m + 4};$	$\prod_{n=1}^6 \frac{10n - 33}{10n^2 - 3n + 8};$	$\sum_{R=1}^7 \prod_{m=1}^7 \frac{\sqrt{tg(R+m)^2 + R}}{R + m^3 + m^{-R} + 0};$
--	--	---

4.1. Vazifa quyidagi tartibda bajariladi:

1. Birinchi misolni yechish uchun ikkita ustunni tanlang va ularga yig'indining o'zgarish parametri nomini, ikkinchisiga esa o'zgaruvchining har bir qiymatidagi hisoblanishni belgilang.

2. Birinchi ustunning keyingi qatoriga 1 qiymat kiriting.

3. Shu ustunning keyingi qatoriga =A9+1 formula kiriting va ENTER tugmasini bosing.

4. Bu yacheykani ajratib "bufer" xotirasiga oling va pastki qatorlarni ajratib ularga nusxa ko'chiring (bu amalni formula kiritilgan yacheykaga kursorni qo'yib, uning pastki burchagini ko'rsatib plyus belgisi chiqqach, sichqonchani bosgan holda pastka surish bilan juda oson bajarsangiz bo'ladi). Natijada siz ketma-ket sonlarga ega bo'lasiz.

5. Ikkinchi ustunning 9 qatoriga, ya'ni B9 yacheykaga =5/(A9^2+A9+4) formulani kiritib ENTER tugmasini bosing.

6. Kursorni shu yacheykaga qo'yib undan pastki yacheykalarga nusxa ko'chiring (buni xuddi birinchi ustundagi kabi bajaring, 4 punktdagidek).

7. Oxirgi 19 qator birinchi ustuniga SUMMA= so'zini yozing.

8. Ikkinchi ustuniga =СУММ (B9:B18) formulasini kiritib, ENTER tugmasini bosing (buni «Автосуммирование» piktogrammasidan foydalanib bajarsa ham bo'ladi, buning uchun ikkinchi ustundagi hisoblangan qiymatlarni ajrating va «Автосуммирование» piktogrammasini bosing).

4.2. Ikkinchi vazifani ham xuddi shunday bajaring. Bu vazifani bajarishda ko'paytmani hisoblashda ПРОИЗВЕД () funksiyasidan foydalaning, ya'ni A15 yacheykasiga quyidagi =ПРОИЗВЕД (F9:F14) formulasini kiriting.

	A	B	C	D	E	F	G	H	I
1									
2									
3	A)	$\sum_{m=1}^{10} \frac{5}{m^2 + m + 4}$		B)	$\prod_{n=1}^6 \frac{10}{n^2} \frac{n-33}{-3n+8}$				
4									
5									
6	Йиғинди ва кўпайтмани ҳисобланг								
7									
8	m	Sm			n	Pn			
9	1	0,833			1	-3,33			
10	2	0,5			2	-1,67			
11	3	0,313			3	-0,38			
12	4	0,208			4	0,833			
13	5	0,147			5	1,111			
14	6	0,109			6	1,154			
15	7	0,083			Ko'paytma=	-2,23	=ПРОИЗВЕД(F9:F14)		
16	8	0,066							
17	9	0,053							
18	10	0,044	=СУММ(B9:B18)						
19	SUMMA=	2,356							
20									

6.4-rasm

4.3.Uchinchi vazifani bajarish tartibi quyidagicha:

1. Ustun bo'ylab yig'indi o'zgaruvchisi qiymatlarini yozib chiqing (A11 dan A16 gacha).
2. Satr bo'ylab ko'paytma o'zgaruvchisi qiymatlarini yozib chiqing (B10 dan H10 gacha).
3. Hosil qilingan massivning birinchi yacheykasiga (B11) ko'rsatilgan formulani kiriting va Enter tugmasini bosong (formuladagi A\$10 yacheyka adresi pastga qarab yacheyka qiymati o'zgarmasligini, ya'ni siljimaslikni bildiradi) .
4. Shu yacheykadan boshqalariga nusxa ko'chiring.
5. S18 yacheykasiga ko'rsatilgan formulani kiriting va Enter tugmasini bosong.
6. Bajargan ishingizni faylga saqlab qo'ying.

	A	B	C	D	E	F	G	H	I	J
1										
2										
3										
4										
5										
6										
7	Йиғинди ва кўпайтмани СуммПроизв() функциясидан фойдаланиб ҳисобланг									
8										
9										
10	R \ m	1	2	3	4	5	6	7		
11	1	0,095	0,045	0,007	0,011	0,017	0,001	0,00175		
12	2	0,094	0,023	0,015	0,006	0,008	0,01	0,0026		
13	3	0,094	0,022	0,015	0,006	0,008	0,01	0,0026		
14	4	0,094	0,022	0,015	0,006	0,008	0,01	0,0026		
15	5	0,094	0,022	0,015	0,006	0,008	0,01	0,0026		
16	6	0,094	0,022	0,015	0,006	0,008	0,01	0,0026		
17										
18	SUMKOP=	0,966	=СУММПРОИЗВ(B11:H16)							

6.5-rasm

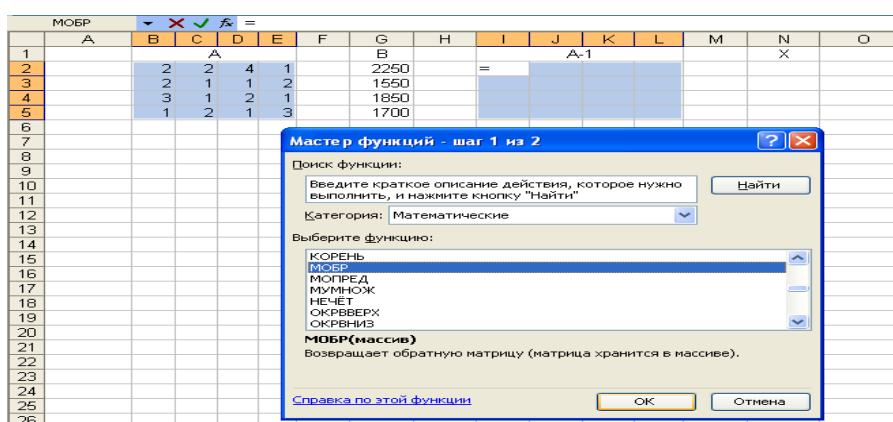
5-vazifa. Berilgan tenglamalar tizimini yeching.

$$\begin{cases} 2x_1 + 2x_2 + 4x_3 + 1x_4 = 2250 \\ 2x_1 + 1x_2 + 1x_3 + 2x_4 = 1550 \\ 3x_1 + 1x_2 + 2x_3 + 1x_4 = 1850 \\ 1x_1 + 2x_2 + 1x_3 + 3x_4 = 1700 \end{cases}$$

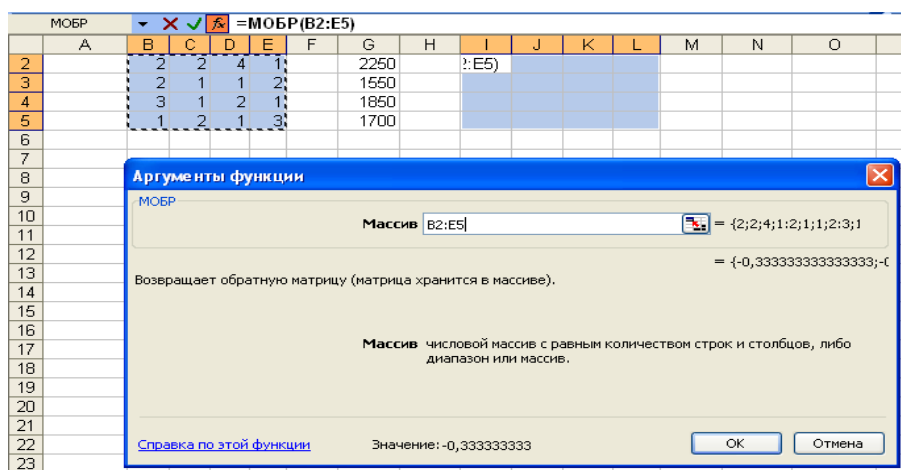
Bu masalani Excel elektron jadval protsessorida yechish ketma-ketligi

quyidagicha:

1. A matritsa elementlarini elektron jadvalga kiriting;
2. B vektor elementlarini elektron jadvalga kiriting;
3. A^{-1} teskari matritsa topish:
 - 3.1. A matritsa elementlarini ajrating;
 - 3.2. Ctrl tugmasini bosgan holda elektron jadvaldan A^{-1} teskari matritsa elementlari chiqishi kerak bo'lgan joylarni ham ajrating;
 - 3.3. "Vstavka funktsii (fx)" piktogrammasini bosning;
 - 3.4. Muloqot oynasining kategoriyalar bo'limidan matematika, keyingi darchadan MOBR funktsiyasini tanlab OK tugmasini bosning;
 - 3.5. Yana A matritsa elementlarini ajratib massiv adresini aniqlab va Ctrl+Shift+Enter tugmalarini baravar bosning. Natijada A^{-1} teskari matritsa elementlari hosil bo'ladi. Rasmlarda bu jarayon ko'rsatilgan.



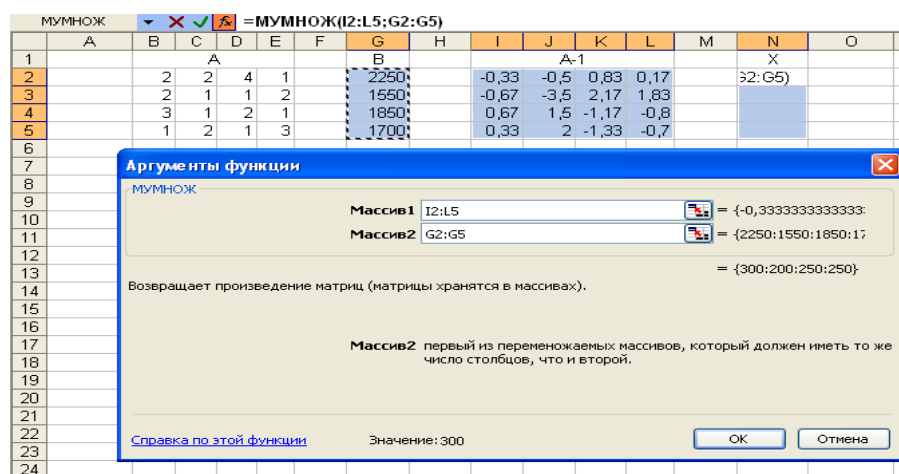
6.6-rasm.



6.7-rasm

4. Topilgan A^{-1} matritsani B vektorga ko'paytirish.
 - 4.1. A^{-1} teskari matritsa elementlarini ajrating;
 - 4.2. Ctrl tugmasini bosgan holda B vektor elementlari keyin hisoblanishi kerak bo'lgan X vektor elementlarini ajrating;
 - 4.3. "Vstavka funktsii (fx)" piktogrammasini bosning;
 - 4.4. Muloqot oynasining kategoriyalar bo'limidan matematika, keyingi darchadan MUMNOJ funktsiyasini tanlab OK tugmasini bosning;

4.5. Oldin A^{-1} matritsa elementlarini ajratib massiv adresini aniqlang, keyin B vektori elementlarini ajratib massiv adresini aniqlang va Ctrl+Shift+Enter tugmalarini baravar bosib. Natija X vektorida hosil bo'ladi. Rasmlarda bu jarayon ko'rsatilgan.



6.8-rasm

N5		{=МУМНОЖ(I2:L5;G2:G5)}													
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	
1															
2			2	2	4	1	2250		-0,33	-0,5	0,83	0,17		300	
3			2	1	1	2	1550		-0,67	-3,5	2,17	1,83		200	
4			3	1	2	1	1850		0,67	1,5	-1,17	-0,8		250	
5			1	2	1	3	1700		0,33	2	-1,33	-0,7		250	
6															
7															

6.9-rasm

4.6. Bajargan ishingizni faylga saqlab qo'ying.
6-vazifa: quyidagi hisoblash jadvalini quring.

	A	B	C	D	E	F
1		Иш хакидан солиқ олиш ҳисоби (сум ҳисобида)				
2	№	Фамилия, исми ва шарифи	Маоши	15% солиқ	1% сугурта	Қулга олиш
3	1	Хасанов Ф.Б.	17800			
4	2	Илҳамов Ч.Л.	14600			
5	3	Хужаев Т.Р.	15600			
6	4	Рахимов Р.С.	15200			
7	5	Рустамов Р.Л.	14600			
8	6	Исмаилов Р.Т.	12500			
9						

6.10-rasm

Bajarish tartibi:

1. Excel oynasida ustunlarni berilgan jadvalga moslab chiqing. Buning uchun sichqoncha kursorni ustunlarni bo'lib turuvchi chiziq ustiga olib kelib, uning tugmachasini bosgan holda, o'nga yoki chapga keraklicha surib, keyin qo'yib yuboring.

2. Ma'lumotlarni kiriting.

3. D3 yacheykaga $=C3*15/100$ formulasini kiriting va "Enter" tugmasini bosing.
4. E3 yacheykaga $=C3*1/100$ formulasini kiriting va "Enter" tugmasini bosing.
5. F3 yacheykaga $=C3-D3-E3$ formulasini kiriting va "Enter" tugmasini bosing.
6. D3, E3 va F3 yacheykalarini ajrating va vaqtinchalik "bufer" xotirasiga oling.
7. Qolgan D4:F8 gacha bo'lgan yacheykalarni ajrating va "Enter" tugmasini bosing.
8. "C" ustundagi ma'lumotlarni o'zgartirib ko'ring. Bunda siz D, E va F ustundagi yangi hisoblashlarni ko'rasiz. Demak siz avtomatik hisoblash jadvaliga ega bo'ldingiz.
9. Bajargan ishingizni faylga saqlab qo'ying.

7-vazifa: Quyidagi jadvalni quring va undagi ma'lumotlar asosida $y=x^2$ funktsiyasini $[-1,1]$ oraliqda, 0,2 qadamda uning qiymatlarini hisoblab grafigini yasang.



6.11-rasm

Bajarish tartibi:

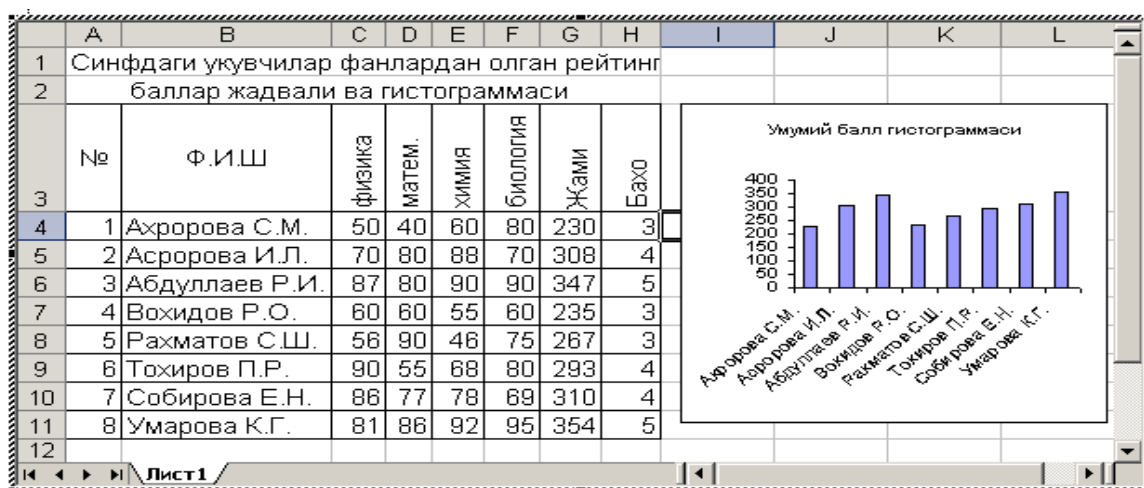
1. Excel oynasida ustunlarni berilgan jadvalga moslab chiqing va 1,2 qatorlarni to'ldiring.
2. A3 yacheykaga 1 va B3 yacheykaga -1 kiriting.
3. A4 yacheykaga $=A3+1$ va B4 yacheykaga $=B3+0,2$ formulalarini kiriting. Bu yacheykalarni ajratib "bufer" xotirasiga oling.
4. A5:B13 yacheykalarini ajratib, "Enter" tugmasini bosing.
5. C3 yacheykasiga $=B3^2$ formulani kiriting va "Enter" tugmasini bosing.
6. C3 yacheykasini ajratib, "bufer" xotirasiga oling.
7. C3:C13 yacheykalarini ajratib, "Enter" tugmasini bosing. Shu bilan siz hisob jadvalini yaratdingiz. Endi grafik yasashga o'ting.
8. B3:C13 yacheykalarini ajrating va standart instrumentlar panelidan "Master diagramm" tugmasini bosing.
9. Endi istalgan joyga kelib sichqoncha tugmasini bosgan holda darcha oching, ketma-ket o'zingizga kerakli grafik ko'rinishini tanlang va "Далее" tugmasini bosing. Keyin esa "Готово" tugmasini bosing va siz darchada funksiya grafigini ko'rasiz.
10. Hosil qilingan grafikni kattalashtirish yoki kichiklashtirish uchun, shu grafik ustiga kelib sichqonchani bir marta bosing. Uning ramkasidagi belgilarga kelib

sichqoncha tugmasini bosib, kerakli joyga kelib qo'yib yuboring.

11.Grafikni tahrirlash uchun grafik ustiga kelib, oldin bir marta va keyin ikki marta tez-tez bosing. Kerakli o'zgartirishlar qiling.

12.Bajargan ishingizni faylga saqlab qo'ying.

8-vazifa: Quyidagi jadvalni quring va gistogramma qurishga urinib ko'ring.



6.12-rasm

Bajarish tartibi:

1.Excel oynasida sarlavhani kiritib, ustunlarni berilgan jadvalga moslab chiqing va A, B, C, D, E, F ustunining 1,2,3 qatorlariga ma'lumotlarni kiriting.

2.G4 yacheykasiga = summ(C4:F4) yoki =C4+D4+E4+F4 formulasini kiriting va "Enter" tugmasini bosing.

3.H4 yacheykasiga =ЕСЛИ(G2/4<55;2; ЕСЛИ(G4/4<71; 3; ЕСЛИ(G4/4<85;4;5))) formulasini kiriting va "Enter" tugmasini bosing.

4.G4 va H4 yacheykalarini ajrating va "bufer" xotirasiga oling.

5.Qolgan G6:H11 gacha bo'lgan yacheykalarni ajrating va "Enter" tugmasini bosing. Siz yacheykalarda hisoblangan sonlarni ko'rasiz.

6.Endi gistogrammani qurish uchun oldin B4:B11 yacheykalarni keyin Ctrl tugmasini bosib, G4:G11 gacha bo'lgan yacheykalarni ajratib, standart instrumentlar panelidan "Мастер диаграмм" tugmasini bosing.

7.Endi istalgan joyga kelib sichqoncha tugmasini bosgan holda darcha oching, ketma-ket o'zingizga kerakli gistogramma ko'rinishini tanlang va "Далее" tugmasini bosing. Keyin esa "Готово" tugmasini bosing va siz darchada gistogrammani ko'rasiz.

8.Hosil qilingan gistogrammani kattalashtirish uchun, shu gistogramma ustiga kelib sichqonchani bir marta bosing. Uning ramkasidagi belgilarga kelib sichqoncha tugmasini bosib kerakli joyga kelib qo'yib yuboring.

9.Gistogrammani tahrirlash uchun gistogramma ustiga kelib, oldin bir marta va keyin ikki marta bosing.

10.Bajargan ishingizni faylga saqlab qo'ying.

Topshiriqlar. Quyida berilgan masalalarni hisoblang.

1.Tikuv fabrikasi bo'limida birinchi tikuvchi bir soatda a=200m., ikkinchisi b=215m., uchinchi c=225m., to'rtinchisi d=250m. ip ishlatadi. Agar ular bir

- kunda $t=7$ soat ishlasa hammasi bo'lib qancha ip ishlatadi.
2. Paxta punktida 12 ta paxta uyumi mavjud bo'lib, har bir uyum eni $a=10\text{m.}$, bo'yi $b=40\text{ m.}$, va balandligi $h=6\text{m.}$ 1m. kub paxta 13 kilogrammni tashkil etsa, punktdagi hamma paxta qanchaligini toping.
3. Bir juft 41-razmerli tufli tikish uchun $a=0.21\text{m.kvadrat}$ charm, 40-razmer uchun $b=0.19\text{m.kv.}$, 39 razmer uchun $c=0.17\text{m.kv.}$ charm ketadi. Agar ishchi bir oyda 41-razmerdan 4 ta, 40 razmerdan 6 ta, 39 razmerdan 15 ta tufli tiksa, u uch oyda qancha charm ishlatgan bo'ladi.
4. Qirqilgan gazlama tomonlarining uzunliklari $a=3\text{ sm.}$, $b=4.8\text{ sm.}$, $c=3.4\text{ sm.}$, $d=5.2\text{sm}$ bo'lsa, uning yuzasini toping.
5. Ishchi $s=36540$ so'm oylik maosh oladi. Agar 1so'mdan 4 ming so'mgacha 15%, 4 ming so'mdan 8 ming so'mgacha 25%, 8 ming so'mdan 12 ming so'mgacha 30% va undan yuqorisidan 36% soliq ushlansa, ishchi qo'lga qancha oylik oladi.
6. Uchburchak shaklda kesilgan matoning tamonlari $a=2.6\text{sm.}$, $b=3.4\text{sm.}$, $c=2.4\text{sm.}$ bo'lsa ular burchaklarini graduslarda aniqlang.
7. Tekislikda koordinatalari $x(i)$ va $y(i)$ ($i=1,2,...,n$) bo'lgan n ta nuqta berilgan. Ular ketma-ket biri ikkinchisi, ikkinchisi uchinchisi bilan va hakazo birlashtirilgan. Eng katta uzun kesma va ularni birlashtiruvchi nuqtalar koordinatalari topilsin. Kiritiladigan va chiqariladigan ma'lumotlar faylda saqlansin. $N=5$ deb olinsin.
8. $M_1, M_2, ..., M_n$ nuqtalar o'z koordinatalari bilan berilgan. Shu nuqtalardan markazi (x_0, y_0) radiusi R ga teng bo'lgan aylana ichida yotadiganlarini aniqlab, ular faylga yozilsin.
9. Sinfda n ta o'quvchi bor, ular har biri matematikadan a_i ($i=1,2,...,n$) reyting balli olgan. Agar umumiy ball 72 bo'lib, reyting mezoni $0 < a_i < 55\%$ da 2 baho, $55\% < a_i < 71\%$ da 3 baho, $71\% < a_i < 85\%$ da 4 baho va $85\% < a_i < 100\%$ da 5 baho deb olinsa, har bir o'quvchi bahosi aniqlansin. Kiritiladigan va chiqariladigan ma'lumotlar faylda jadval ko'rinishda saqlansin.
10. Fabrikada n ta ishchi ishlaydi va m xil mahsulot ishlab chiqariladi. i -chi ishchi j -chi mahsulotdan ishlab chiqargan hajmi a_{ij} bo'lsa, har bir ishchining qaysi mahsulotdan eng ko'p ishlab chiqarishi aniqlansin.
11. Korxonada n ta ishchi ishlaydi. Har bir i -chi ishchi oylik maoshi a_i . Agar har bir ishchining ish haqidan 1 so'mdan 8 ming so'mgacha 12%, undan yuqorisidan 25% soliq ushlansa, har bir ishchi qo'lga oladigan maoshi b_i va olingan soliq miqdori c_i topilsin va jadval ko'rinishda yozilsin.
12. Zavod n xil tovar ishlab chiqaradi. Har bir tovar tannarxi a_i va miqdori c_i bo'lsa, uchta eng ko'p foyda olib kelishi mumkin bo'lgan tovarlar aniqlansin.
13. Birdan boshlab 20 ta 7 ga karrali sonlar aniqlanib ularni $A(20)$ massiviga joylashtirilsin va ular ichida 9 ga karralilari topilsin.
14. Birdan o'n minggacha sonlar ichida nechta 7 hamda 9 ga karrali son borligi aniqlansin va ular faylga yozilsin.
15. Viloyatda n ta tuman bo'lib, har bir tumanning shu yil etishtirgan paxta miqdorlari mavjud. Viloyat bo'yicha o'rtacha qanchadan paxta yetishtirilgan? Eng ko'p paxtani qaysi tuman yetishtirgan.
16. Natural n soni berilgan, shu sonning hamma bo'linuvchilari topilsin va ular

faylga yozilsin.

17. Tekislikda n ta nuqta berilgan $M_1(x_1, y_1), M_2(x_2, y_2), \dots, M_n(x_n, y_n)$. Koordinata boshi $M(x_0, y_0)$ nuqta bo'lsa, berilgan nuqtalarning koordinata boshigacha bo'lgan masofalari topilsin va eng yaqin masofada turgan nuqta koordinatasi aniqlansin.

18. Bir metr ipning og'irligi 0,03 g. Gazmol to'qishda 1 m. uchun 1000 ta ip ishlatiladi. Eni 1,4 m. uzunligi 5 m. gazmol og'irligi aniqlansin.

19. Tekislikda $M_1, M_2, \dots, M_n$ nuqtalar to'plami o'z koordinatalari bilan berilgan. Shu nuqtalar to'plami ichida 1 va 3 chi chorakda joylashganlar soni va ular koordinatalari chiqarilsin.

20. Tajribali bichuvchi bir juft poyabzalni 1 soatda, shogirdi 2 soatda bichadi. Agar ular ishni bir vaqtda boshlasalar, qancha vaqtdan so'ng yana birga boshlaydilar.

7-laboratoriya ishi: C++ dasturlash tili va uning asosiy operatorlari.

Ishdan maqsad: C++ dasturlash tili va uning asosiy operatorlari bilan tanishish. Dastur tuzishni o'rganish.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. C++ dasturlash tili bilan tanishish.
3. C++ tilida masalalar yechish.
4. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

C++ tilidagi dastur tuzilishi.

C++ tilida dastur tuzilishi tushuntirish uchun sodda dastur keltiramiz [3, 4-8 b.].

```
#include <iostream.h> // sarlavha faylni qo'shish
int main()           // bosh funksiya tavsifi
{
    // asosiy blok boshlanishi
    cout<<"Kompyuter olami!\n"; // satrni chop etish;
    return 0;              // funksiya qaytaradigan qiymat
}                          // asosiy blok tugashi
```

Dasturning 1-satrida `"#include"` preprosessor ko'rsatmasi bo'lib, dastur kodiga standart oqimli o'qish-yozish funksiyalari va uning o'zgaruvchilari e'loni joylashgan `«iostream.h»` sarlavha faylini qo'shadi (mnemonika: `'i'`(*input*) - kiritish (o'qish); `'o'`(*ouput*) - chiqarish (yozish); `"stream"`- oqim; `'h'`(*head*) – sarlavha). Kelishuv bo'yicha standart oqim ekranga chiqarish hisoblanadi. Keyingi qatorlarda dasturning yagona, asosiy funksiyasi - `main()` funksiyasi tavsifi keltirilgan. Shuni qayd etish kerakki, C++ dasturida, albatta, `main()` funksiyasi bo'lishi shart va dastur shu funksiyani bajarish bilan o'z ishini boshlaydi. Funksiya nomi oldidagi `"int"` kalit so'zi funksiya bajarilishi natijasida qaytaraladigan qiymat turini bildiradi. Bunday holat funksiyaning matematikadagi tavsifiga mos keladi. Keyingi qatordan funksiya tanasi - figurali qavsga olingan amallar ketma-ketligi keladi.

Bizning holda funksiya tanasi ikkita amaldan iborat. Birinchisi, konsol rejimida belgilar ketma-ketligini oqimga chiqarish amali qo'llangan. Buning uchun «iostream.h» sarlavha faylida aniqlangan *cout* obyektidan foydalanilgan. Uning formati quyidagi ko'rinishda:

cout << <ifoda>;

Bu yerda "<<" – ma'lumot uzatish amali («..ga joylashtir»), <ifoda> sifatida o'zgaruvchi yoki sintaksisi to'g'ri yozilgan va qandaydir qiymat qabul qiluvchi til ifodasi kelishi mumkin (keyinchalik, burchak qavs ichiga olingan o'zbekcha satr ostini til tarkibiga kirmaydigan tushuncha deb qabul qilish kerak). Ikkinchisi, funksiya o'z ishini tugatganligini anglatuvchi va undan chiqishni amalga oshiruvchi "**return 0;**" operatoridir. Odatda, bajarilishi normal tugagan funksiyalar operatsion sistemaga 0 qiymatini qaytaradi. Shu qoidagi rioya qilgan holda dastur ham 0 qiymatini qaytaradi.

Bajariluvchi dasturni hosil qilish uchun dastur matni kompilyatsiya qilinishi kerak. Kompilyatsiya jarayonining o'zi ham ikkita bosqichdan tashkil topadi. Boshida preprotsessor ishlaydi, u matndagi kompilyatsiya direktivalarini bajaradi, xususan *#include* direktivasi bo'yicha ko'rsatilgan kutubxonalardan C++ tilida yozilgan modullarni dastur tarkibiga kiritadi. Shundan so'ng kengaytirilgan dastur matni kompilyatorga uzatiladi. Kompilyator o'zi ham dastur bo'lib, uning uchun kiruvchi ma'lumot bo'lib, C++ tilida yozilgan dastur matni hisoblanadi. C++ tilida tuzilgan dastur ob'ektlar, funktsiyalar, o'zgaruvchilar va boshqa elementlardan tashkil topadi.

C++ tilida yozilgan dastur tarkibini o'rganish uchun yana bir oddiygina XAYRLI TONG.CPP dasturi bilan tanishamiz. Bu dastur kichik bo'lishiga qaramasdan bizda qiziqish uyg'otuvchi bir nechta elementdan iboratdir.

XAYRLI TONG.CPP dasturi misolida C++ tilida tuzilgan dastur qismlarini namoyish qilish:

```
#include <iostream.h>
int main( )
{
    cout << "Xayrli tong!\n";
    return 0;
}
```

Natija:

Xayrli tong!

– satrda iostream fayli joriy faylga biriktilyapti. Dasturda birinchi funta (#) belgisi joylashgan. U preprotsessorga signal uzatadi.

Kompilyator dastur matnini leksema (atomar) elementlarga ajratadi va uni leksik, keyinchalik sintaktik tahlil qiladi. Leksik tahlil jarayonida u matnni leksemalarga ajratish uchun «probel ajratuvchisini» ishlatadi. Probel ajratuvchisiga – probel belgisi (' '), '\t' - tabulyatsiya belgisi, '\n' - keyingi qatarga o'tish belgisi, boshqa ajratuvchilar va izohlar kiradi.

Dastur matni tushunarli bo'lishi uchun izohlar ishlatiladi. Izohlar dastur amal qilishiga hech qanday ta'sir qilmaydi.

C++ tilida izohlar ikki ko‘rinishda yozilishi mumkin. Birinchisida “/*” dan boshlanib, “*/” belgilar oralig‘ida joylashgan barcha belgilar ketma-ketligi izoh hisoblanadi, ikkinchisi «satriy izoh» deb nomlanadi va u “//” belgilardan boshlangan va satr oxirigacha yozilgan belgilar ketma-ketligi bo‘ladi. Izohning birinchi ko‘rinishida yozilgan izohlar bir necha satr bo‘lishi va ulardan keyin C++ operatorlari davom etishi mumkin. Berilganlarni standart oqimdan (odatda, klaviaturadan) o‘qish quyidagi formatda amalga oshiriladi:

```
cin >> <o‘zgaruvchi>;
```

Bu yerda <o‘zgaruvchi> oqimdan qiymat qabul qiluvchi o‘zgaruvchining nomi.

```
int n;
cout << " n = ";
cin >> n;
```

Butun turdagi n o‘zgaruvchisi kiritilgan qiymatni o‘zlashtiradi [3, 11-12 b.]. Bir paytning o‘zida probel vositasida bir nechta va har xil turdagi qiymatlarni oqimdan kiritish mumkin. Qiymat kiritish “Enter” tugmasini bosish bilan tugaydi. Agar kiritilgan qiymatlar soni o‘zgaruvchilar sonidan ko‘p bo‘lsa, «ortiqcha» qiymatlar bufer xotirada saqlanib qoladi.

```
int x, y;
float z;
cin >> x >> y >> z;
```

SHuni qayd etish kerakki, oqimga qiymat kiritishda probel ajratuvchi hisoblanadi. Haqiqiy sonning butun va kasr qismlari ‘.’ belgisi bilan ajratiladi.

Taqqoslash amallari.

C++ tilida qiymatlarni solishtirish uchun taqqoslash amallari aniqlangan [3, 22 b.]. Taqqoslash amali binar amal bo‘lib, quyidagi ko‘rinishga ega:

```
<operand1> <taqqoslash amali> <operand2>
```

Taqqoslash amallarining natijasi - taqqoslash o‘rinli bo‘lsa, *true* (rost), aks holda *false* (yolg‘on) qiymat bo‘ladi. Agar taqqoslashda arifmetik ifoda qatnashsa, uning qiymati 0 qiymatidan farqli holatlar uchun 1 deb hisoblanadi. Taqqoslash amallari va ularning qo‘llanishi

Amallar	Qo‘llanishi	Mazmuni (o‘qilishi)
<	$a < b$	“ a kichik b ”
<=	$a \leq b$	“ a kichik yoki teng b ”
>	$a > b$	“ a katta b ”
>=	$a \geq b$	“ a katta yoki teng b ”
==	$a == b$	“ a teng b ”
!=	$a != b$	“ a teng emas b ”

Mantiqiy operatorlar.

Mantiqiy ko‘paytirish operatori ikkita ifodani hisoblaydi, agar ikkala ifoda *true* qiymat qaytarsa, *BA* operatori ham *true* qiymat qaytardi [3, 22 b.].

Operator	Belgi	Namuna
<i>BA</i>	&&	<i>ifoda1 && ifoda2</i>
<i>YOXI</i>	 	<i>ifoda1 ifoda2</i>
<i>INKOR</i>	!	<i>!ifoda</i>

Mantiqiy ko‘paytirish operatori **&&** belgi orqali belgilanadi.

ifoda1	ifoda1	ifoda1 && ifoda2
<i>false</i> (0)	<i>false</i> (0)	<i>false</i> (0)
<i>false</i> (0)	<i>true</i> (0 emas)	<i>false</i> (0)
<i>true</i> (0 emas)	<i>false</i> (0)	<i>false</i> (0)
<i>true</i> (0 emas)	<i>true</i> (0 emas)	<i>true</i> (1)

Mantiqiy qo‘shish operatori ham ikkita ifoda orqali hisoblanadi. Agarda ulardan birortasi *true* bo‘lsa, mantiqiy qo‘shish operatori *true* qiymat qaytaradi. Mantiqiy qo‘shish operatori **||** belgi orqali belgilanadi.

ifoda1	ifoda1	ifoda1 ifoda2
<i>false</i> (0)	<i>false</i> (0)	<i>false</i> (0)
<i>false</i> (0)	<i>true</i> (0 emas)	<i>true</i> (1)
<i>true</i> (0 emas)	<i>false</i> (0)	<i>true</i> (1)
<i>true</i> (0 emas)	<i>true</i> (0 emas)	<i>true</i> (1)

Mantiqiy inkor operatori tekshirilayotgan ifoda *yolg‘on* bo‘lsa, *true* qiymat qaytaradi. Agarda tekshirilayotgan ifoda *rost* bo‘lsa, inkor operatori *false* qiymat qaytaradi. Mantiqiy inkor operatori **!** belgi orqali belgilanadi.

ifoda1	!ifoda1
<i>false</i> (0)	<i>true</i> (1)
<i>true</i> (0 emas)	<i>false</i> (0)

C++ tilining qurilmalari operatorlarni blok ko‘rinishida tashkil qilishga imkon beradi. *Blok* – ‘{‘ va ‘}’ belgilari oralig‘iga olingan operatorlar ketmaketligi bo‘lib, u kompilyator tomonidan yaxlit bir operator deb qabul qilinadi. Blok ichida e‘lon operatorlari ham bo‘lishi mumkin va ularda e‘lon qilingan o‘zgaruvchilar faqat shu blok ichida ko‘rinadi (amal qiladi), blokdan tashqarida ko‘rinmaydi. Blokdan keyin ‘;’ belgisi qo‘yilmasligi mumkin, lekin blok ichidagi har bir ifoda ‘;’ belgisi bilan yakunlanishi shart.

{ifoda_1; ifoda_2; ...ifoda_n;}

Misol tariqasida diskriminantni hisoblash usuli yordamida $ax^2+bx+c=0$ ko‘rinishidagi kvadrat tenglama ildizlarini topish masalasini ko‘raylik:

```
#include <iostream.h>
```

```
#include <smath.h>
```

```
int main()
```

```
{
```

```
float a, b, c;
```

```
float D, x1, x2;
```

```

cout << "ax^2+bx+c=0 tenglama ildizini topish. ";
cout << "\n a - koefitsiyentini kiriting: ";
cin >> a;
cout << "\n b - koefitsiyentini kiriting: ";
cin >> b;
cout << "\n c - koefitsiyentini kiriting: ";
cin >> c;
D = b * b - 4 * a * c;
if (D < 0)
{
cout << "Tenglama haqiqiy ildizga ega emas!";
return 0;
}
if (D == 0)
{
cout << "Tenglama yagona ildizga ega: ";
x1 = -b / (2 * a);
cout << "\nx= " << x1;
}
else
{
cout << "Tenglama ikkita ildizga ega: ";
x1 = (-b + sqrt(D)) / (2 * a);
x2 = (-b - sqrt(D)) / (2 * a);
cout << "\nx1= " << x1;
cout << "\nx2= " << x2;
}
return 0;
}

```

Dastur bajarilganda, birinchi navbatda, tenglama koefitsiyentlari - a , b , c o'zgaruvchilar qiymatlari kiritiladi, keyin diskriminant D o'zgaruvchi qiymati hisoblanadi. Keyin D qiymatining manfiy ekanligi tekshiriladi. Agar shart o'rinli bo'lsa, yaxlit operator sifatida keluvchi '{' va '}' belgilari orasidagi operatorlar bajariladi va ekranga *"Tenglama haqiqiy ildizga ega emas!"* xabari chiqadi va dastur o'z ishini tugatadi (**return 0;** operatorini bajarish orqali). Diskriminant noldan kichik bo'lmasa, navbatdagi shart operatori uni nolga tengligini tekshiradi. Agar shart o'rinli bo'lsa, keyingi qatorlardagi operatorlar bloki bajariladi – ekranga *"Tenglama yagona ildizga ega:"* xabari, x_1 hamda o'zgaruvchi qiymati chop etiladi, aks holda, ya'ni diskriminantning qiymati noldan katta holati uchun **else** kalit so'zidan keyingi operatorlar bloki bajariladi va ekranga *"Tenglama ikkita ildizga ega:"* xabari hamda x_1 va x_2 o'zgaruvchilar qiymatlari chop etiladi. Shu bilan shart operatoridan chiqiladi va asosiy funksiyaning return ko'rsatmasini bajarish orqali dastur o'z ishini tugatadi.

Topshiriqlar. Quyidagi variantlarda berilgan masalalarning C++ da dasturini tuzing.

1. Kvadratning tomoni a berilgan. Uning perimetri aniqlansin: $P = 4 * a$
2. Kvadratning tomoni a berilgan. Uning yuzasi aniqlansin: $S = a^2$
3. To'g'ri to'rtburchakning yuzasi a va b berilgan. Uning yuzasi $S = a * b$ va $P = 2 * (a + b)$ perimetri aniqlansin.
4. Aylananing diametri d berilgan. Uning uzunligi $L = \frac{\pi * d}{2}$, $\pi = 3,14$
5. Kubning yon tomoni a berilgan. Uning hajmini $V = a^3$ va to'la sirti $S = 6 * a^2$ aniqlansin.
6. Parallelepipedning tomonlari a, b, c berilgan. Uning hajmini $V = a * b * c$ va $S = 2 * (a * b + b * c + a * c)$ aniqlansin.
7. Doiraning radiusi R berilgan. Uning uzunligi L va yuzasi S aniqlansin. $L = 2 * \pi * R$, $S = \pi * R^2$
8. Ikkita son a va b berilgan. Ularning o'rta arifmetigi aniqlansin: $(a + b) / 2$
9. Ikkita manfiy bo'lmagan a va b sonlari berilgan. Ularning o'rta geometrigi aniqlansin.
10. Nolga teng bo'lmagan ikkita son berilgan. Ularning yig'indisini, ko'paytmasini va har birining kvadrati aniqlansin.
11. Nolga teng bo'lmagan ikkita son berilgan. Ularning yig'indisini, ko'paytmasini va har birining moduli aniqlansin.
12. To'g'ri uchburchakning katetlari a va b berilgan. Uning gipotenuzasi c va perimetri P aniqlansin. $c = \sqrt{a^2 + b^2}$; $P = a + b + c$

8-laboratoriya ishi. C++ algoritmik tilning tarmoqlanish operatorlari.

Ishdan maqsad: C++ algoritmik tilning tarmoqlanish operatorlari bilan ishlash.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. C++ algoritmik tilning tarmoqlanish operatorlari bilan ishlash.
3. If shart operatori bilan tanishish.
4. Dastur bilan ishlash.
5. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

If va *if else* operatorlari, **?: amali.**

Dastur bajarilishining birorta qadamida qandaydir shartni tekshirish natijasiga ko'ra boshqaruvni dasturning u yoki bu bo'lagiga uzatish mumkin (Tarmoqlanuvchi algoritim). Tarmoqlanishni amalga oshirish uchun shartli operatorlardan foydalaniladi.

If operatori. *If* qandaydir shartni rostlikka tekshirish natijasiga ko'ra dasturda tarmoqlanishni amalga oshiradi:

If ($<shart>$) $<operator>;$

Bu yerda $<shart>$ har qanday ifoda bo'lishi mumkin, odatda u taqqoslash

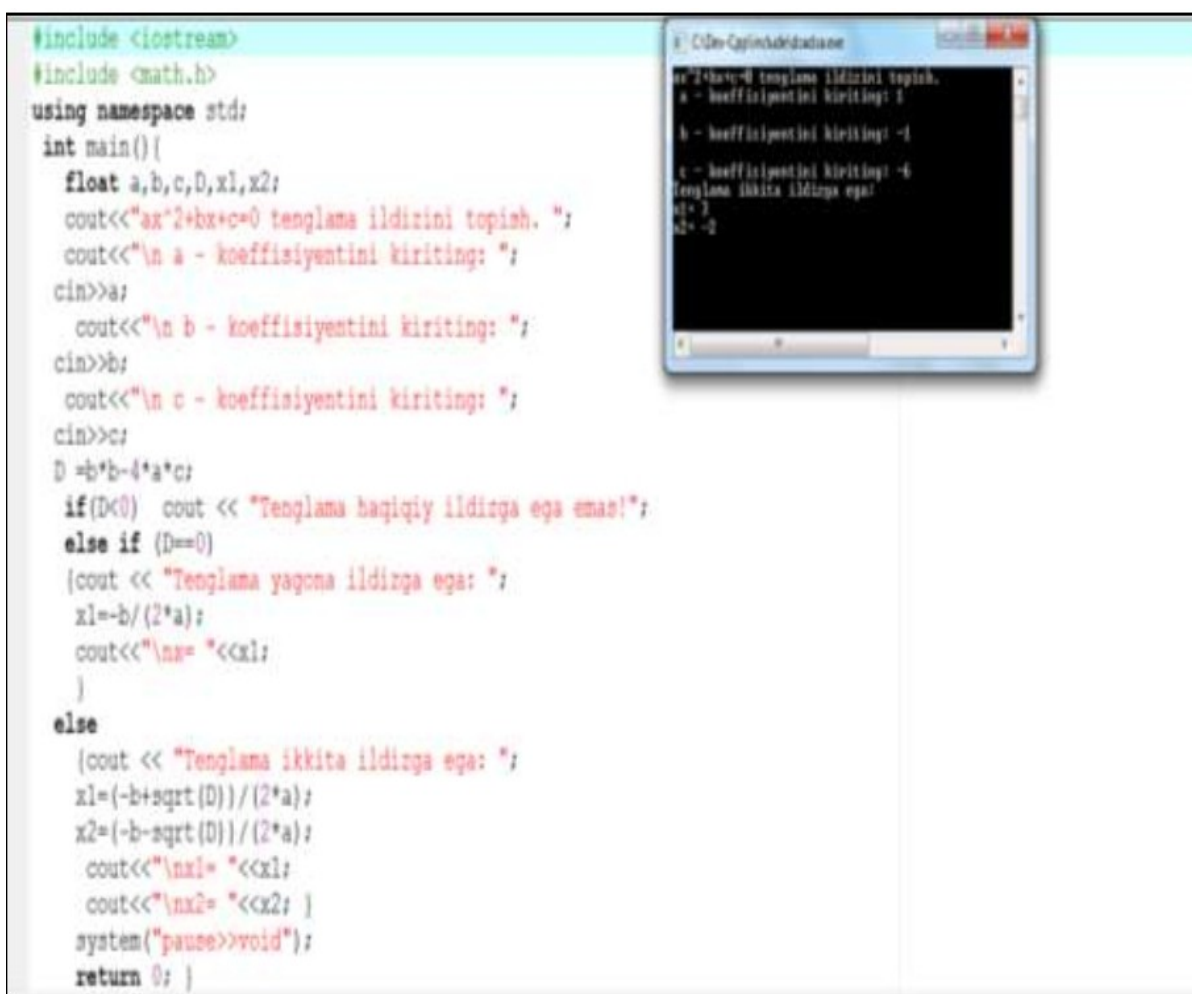
amali bo'ladi.

Agar shart 0 qiymatidan farqli yoki rost (true) bo'lsa, *<operator>* bajariladi, aks holda, ya'ni shart 0 yoki yolg'on (false) bo'lsa, hech qanday amal bajarilmaydi va boshqaruv *if* operatoridan keyingi operatorga o'tadi (agar u mavjud bo'lsa).

C++ tilining qurilmalari operatorlarni blok ko'rinishida tashkil qilishga imkon beradi. Blok - '{' va '}' belgi oralig'iga olingan operatorlar ketma-ketligi bo'lib, u kompilyator tomonidan yaxlit bir operator deb qabul qilinadi. Blok ichida e'lon operatorlari ham bo'lishi mumkin va ularda e'lon qilingan o'zgaruvchilar faqat shu blok ichida ko'rinadi (amal qiladi), blokdan tashqarida ko'rinmaydi. Blokdan keyin ';' belgisi qo'yilmasligi mumkin, lekin blok ichidagi har bir ifoda ';' belgisi bilan yakunlanishi shart. Shart operatorining *if - else* ko'rinishi quyidagicha:

if (<shart-ifoda>) <operator1>; else <operator2>;

Bu yerda *< shart - ifoda >* 0 qiymatidan farqli yoki true bo'lsa, *<operator1>*, aks holda *<operator2>* bajariladi. Misol tariqasida diskriminantni hisoblash usuli yordamida $ax^2 + bx + c = 0$ ko'rinishidagi kvadrat tenglama ildizlarini topish masalasini ko'raylik:



8.1-rasm.Kvadrat tenglamani yechish dasturi

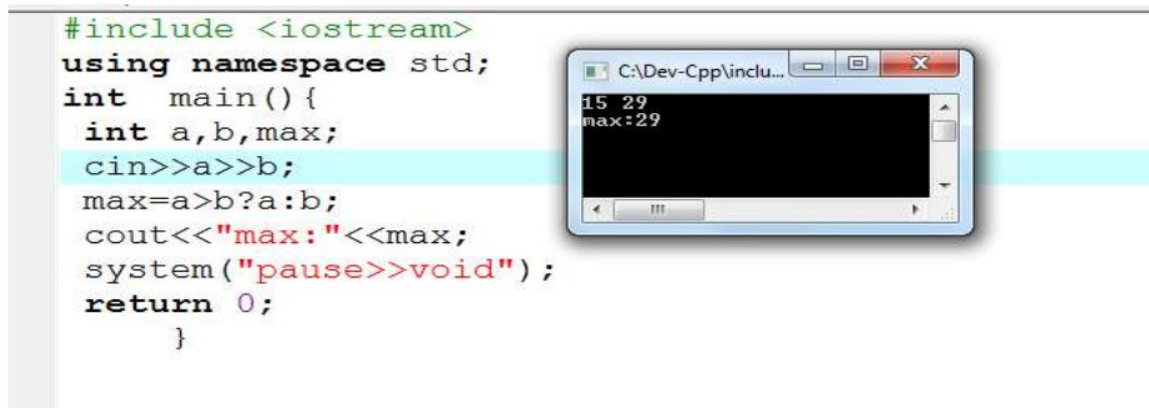
Dastur bajarilganda, birinchi navbatda tenglama koeffitsiyentlari - a,b,c o'zgaruvchilar qiymatlari kiritiladi, keyin diskriminant – D o'zgaruvchi qiymati hisoblanadi. Keyin D qiymatining manfiy ekanligi tekshiriladi. Agar shart o'rinli bo'lsa, yaxlit operator sifatida keluvchi '{' va '}' belgilari orasidagi operatorlar bajariladi va ekranga "Tenglama haqiqiy ildizlarga ega emas" xabari chiqadi va dastur o'z ishini tugatadi ("return 0;" operatorini bajarish orqali). Diskriminant noldan kichik bo'lmasa, navbatdagi shart operatori uni nolga tengligini tekshiradi. Agar shart o'rinli bo'lsa, keyingi qatorlardagi operatorlar bloki bajariladi - ekranga "Tenglama yagona ildizga ega:" xabari, hamda x1 o'zgaruvchi qiymati chop etiladi va dastur shu yerda o'z ishini tugatadi, aks holda, ya'ni D qiymati noldan katta holati uchun else kalit so'zidan keyingi operatorlar bloki bajariladi va ekranga "Tenglama ikkita ildizga ega:" xabari, hamda x1 va x2 o'zgaruvchilar qiymatlari chop etiladi. Shu bilan shart operatoridan chiqiladi va asosiy funksiyaning return ko'rsatmasini bajarish orqali dastur o'z ishini tugatadi.

?: shart amali. Agar tekshirilayotgan shart nisbatan sodda bo'lsa, shart amalining "? :." ko'rinishini ishlatish mumkin:

< shart ifoda > ? < ifoda1 > : < ifoda2 >;

Shart amali *if* shart operatoriga o'xshash holda ishlaydi: agar *<shart ifoda>* 0 qiymatidan farqli yoki *true* bo'lsa, *<ifoda1>*, holda *<ifoda2>* bajariladi. Odatda ifodalar qiymatlari birorta o'zgaruvchiga o'zlashtiriladi.

Misol:



8.2-rasm

Berilgan masala uchun algoritm.

Yuqorida berilgan masalani hisoblash uchun birinchi navbatda uni matematik ko'rinishdan C++ dasturlash muhitiga o'tkazishda *if* shart operatoridan foydalanamiz.

$$Y = \begin{cases} \sqrt{|n_1 \cdot n_2|} & \text{agar } n_1 n_2 < 0.2 \\ \sqrt{|n_1 + n_2|} & \text{agar } n_1 n_2 \geq 0.1 \end{cases}$$

Asosiy dastur matni:

```

#include <iostream>
#include <math.h>
using namespace std;
int main() {
    double n1, n2, Y;
    cin >> n1 >> n2;
    if (n1*n2 < 0.2) {
        Y = sqrt(fabs(n1*n2));
    } else
    if (n1*n2 >= 0.1) Y = sqrt(fabs(n1+n2));
    cout.precision(3);
    cout << fixed << Y;
    system("pause >> void");
    return 0; }

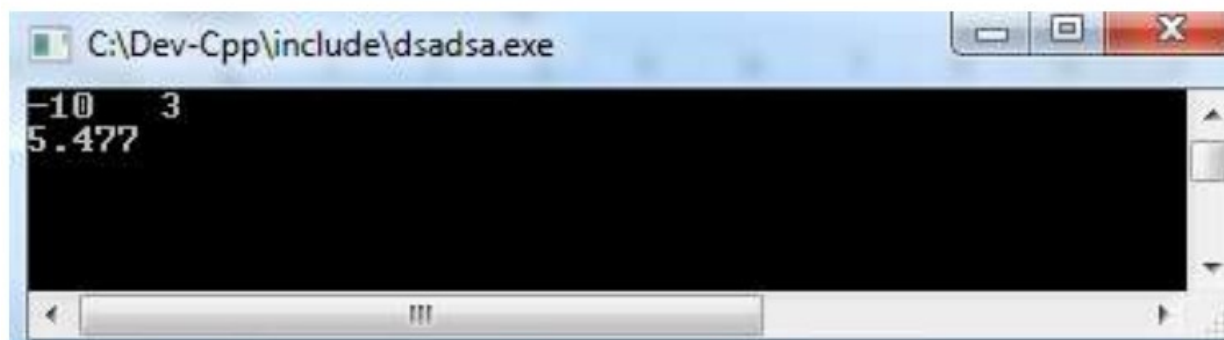
```

8.3-rasm

Berilgan misol ishlanish natijalari tahlili.

n_1 va n_2 haqiqiy sonlari kiritilgandan so'ng ularning ko'paytmasi *if* shart operatori yordamida tekshirilib agarda $n_1 n_2 < 0.2$ shart bajarilsa $Y = \sqrt{|n_1 \cdot n_2|}$ qiymat, aks holda, $n_1 n_2 \geq 0.1$ shart bajarilsa $Y = \sqrt{|n_1 + n_2|}$ qiymatni o'zlashtiradi va natijaga 10^{-3} aniqlikda chiqaradi.

Natija.



8.4-rasm

C++ tilida oddiy dastur. C++ tilida tuzilgan dastur ob'ektlar, funktsiyalar, o'zgaruvchilar va boshqa elementlardan tashkil topadi.

Topshiriqlar. Quyidagi variantlarda berilgan tarmoqlanuvchi algoritmlar dasturini tuzing.

№	Variantlar	№	Kirish	Chiqish
1	$Y = \begin{cases} m^2n + 1 - c, & \text{agar } n + 1 > 0 \\ (m + n)^2 + cm^2, & \text{agar } n + 1 \leq 0 \end{cases}$	1	$m=1$ $n=2$ $c=3$	$Y=0.000$
		2	$m=2$ $n=3$ $c=7$	$Y=6.000$
2	$Y = \begin{cases} \frac{1}{\sqrt{9+x^2}}, & \text{agar } a < 5 \\ b \cdot \sin a, & \text{agar } a \geq 5 \end{cases}$ <i>bu yerda $a = d^2 + \frac{cd}{c^2 - d^2}$;</i>	1	$b=1$ $c=2$ $d=3$ $x=4$	$Y=0.999$
		2	$b=2$ $c=3$ $d=7$ $x=-6$	$Y=-1.952$
3	$Z = \begin{cases} 7x^2 - 3ab - 5ab, & \text{agar } a \geq 0; \\ 15a - 7b, & \text{agar } a < 0 \end{cases};$	1	$a=1$ $b=2$ $x=3$	$Z=47.000$
		2	$a=-2$ $b=9$ $x=7$	$Z=-93.000$
4	$Y = \begin{cases} \frac{a^2+b^2}{c} + \sqrt{a^2+x}, & \text{agar } x \geq 0 \\ \frac{\sin x + b}{a-b}, & \text{agar } x < 0 \end{cases}$ <i>bu yerda $x = \frac{a^2-b^2}{c^2-b}$;</i>	1	$a=1$ $b=2$ $c=3$	$Y=-1.090$
		2	$a=2$ $b=3$ $c=7$	$Y=-2.170$
5	$Y = \begin{cases} (nm^2 + d)^2, & \text{agar } d > 1 \\ \frac{d}{n^2+m^2}, & \text{agar } d \leq 1 \end{cases};$	1	$m=1$ $n=2$ $d=3$	$Y=25.000$
		2	$m=4$ $n=7$ $d=0.2$	$Y=0.003$
6	$Z = \begin{cases} \frac{ax^2}{b-1}, & \text{agar } a < 9 \\ (a+1)^3 + cx^3, & \text{agar } a \geq 9 \end{cases};$	1	$a=1$ $b=2$ $c=3$ $x=4$	$Z=16.000$
		2	$a=12$ $b=3$	$Z=7300.000$

			$c=7$ $x=9$	
7	$X = \begin{cases} \frac{a^3}{3+a}, \text{ agar } a > 0 \\ \sqrt{\left \frac{a^2+2}{1+a}\right }, \text{ agar } a \leq 0 \end{cases} \text{ bunda } a = \frac{b^2-c^2}{d};$	1	$b=4$ $c=-2$ $d=8$	$X=0.750$
		2	$b=4$ $c=-3$ $d=7$	$X=0.250$
8	$Z = \begin{cases} \sqrt{\left \frac{x}{x+1}\right } - \sqrt{x}, \text{ agar } x > 3 \\ \left(\frac{\ln x }{x}\right)^3, \text{ agar } x \leq 3 \end{cases}$	1	$x=2$	$Z=0.042$
		2	$x=3.7$	$Z=1.066$
9	$K = \begin{cases} xe^x \sin x, \text{ agar } x \geq 0 \\ \frac{1}{3} \ln^3 x , \text{ agar } x < 0 \end{cases};$	1	$x=3$	$K=8.503$
		2	$x=-7$	$K=2.456$
10	$Y = \begin{cases} \ln(x + \sqrt{x^2 + 9}), \text{ agar } x \geq 0; \\ tg^3 x, \text{ agar } x < 0 \end{cases};$	1	$x=3$	$Y=1.980$
		2	$x=-7.2$	$Y=0.308$
11	$Y = \begin{cases} \frac{a}{a^2+x}, \text{ agar } a > 5 \\ \frac{1}{2a^5+\sin a} \text{ agar } a \leq 5 \end{cases};$	1	$a=4$ $x=9$	$Y=0.008$
		2	$a=7$ $x=1$	$Y=0.140$
12	$Y = \begin{cases} \sqrt{k}, \text{ agar } \sin k \leq 0.2 \\ \frac{1}{\sqrt{k}}, \text{ agar } \sin k > 0.2 \end{cases}$	1	$k=7$	$Y=0.378$
		2	$k=1.4$	$Y=0.845$
13	$Y = \begin{cases} \sin^2(2x) - \cos^2 x, \text{ agar } x > 0 \\ \frac{1}{x^2 - \sqrt{x}}, \text{ agar } x \leq 0.3 \end{cases};$	1	$x=0.5$	$Y=-0.062$
		2	$x=0.7$	$Y=0.386$
14	$P = \begin{cases} \frac{1}{2}(3x^2 - 1), \text{ agar } x > 0.4 \\ \frac{1}{2}(5x^3 - 3x), \text{ agar } x \leq 0.4 \end{cases};$	1	$x=7.3$	$P=79.435$
		2	$x=0.12$	$P=-0.176$
15	$Y = \begin{cases} x^2 + 4, \text{ agar } x < 10 \\ x^3 - 7, \text{ agar } x \geq 10 \end{cases} \text{ bunda } x = \frac{a^2-b}{c};$	1	$a=1$ $b=2$ $c=3$	$Y=4.11$
		2	$a=-2$ $b=3$ $c=7$	$Y=4.02$
16	$Y = \begin{cases} \sqrt{15a^2 + 21b^2}, \text{ agar } a > b \\ \sqrt{15b^2 + 21a^2}, \text{ agar } a \leq b \end{cases};$	1	$a=7$ $b=3$	$Y=30.397$
		2	$a=1.2$ $b=3.8$	$Y=15.711$
17	$Y = \begin{cases} \ln 2x - 3z^2 , \text{ agar } x < 5z \\ \ln 2x^2 - 3z , \text{ agar } x > 5z \end{cases};$	1	$x=-10$ $z=3$	$Y=3.850$
		2	$x=6.3$ $z=-7$	$Y=4.609$
18	$P = \begin{cases} \sin(5k + 3m), \text{ agar } k > m \\ \cos(5k + 3m), \text{ agar } k \leq m \end{cases};$	1	$k=5$ $m=-6$	$Y=0.754$
		2	$k=-4.2$ $m=3.7$	$Y=0.458$
19	$Y = \begin{cases} \sqrt{2k_1 - 7k_2}, \text{ agar } k_1 k_2 < 1 \\ \sqrt{2k_1 + 7k_2}, \text{ agar } k_1 k_2 \geq 1 \end{cases};$	1	$k_1 = 0.02$ $k_2 = 4$	$Y=5.288$
		2	$k_1 = 9$ $k_2 = 7$	$Y=8.185$
20	$Y = \begin{cases} \frac{4r+3m}{r^2+m^2}, \text{ agar } r > m + \frac{1}{2} \\ r-m , \text{ agar } r \leq m + \frac{1}{2} \end{cases};$	1	$m=3$ $r=5$	$Y=0.853$
		2	$m=6.5$ $r=7$	$Y=0.500$

9-laboratoriya ishi. C++ algoritmik tilning takrorlanish operatorlar.

Ishdan maqsad. Takrorlanuvchi algoritmlar va ular ustida amallar bajarish ko'nikmalarini oshirish. Dasturini tuzish.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. C++ tillarida for, while standart sikl operatorlari bilan ishlash.
3. For siklining while va do/while sikllaridan asosiy farqini o'rganish.
4. do...while konstruktsiyasi yordamida sikl tashkil etish.
5. Tajriba ishinings hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

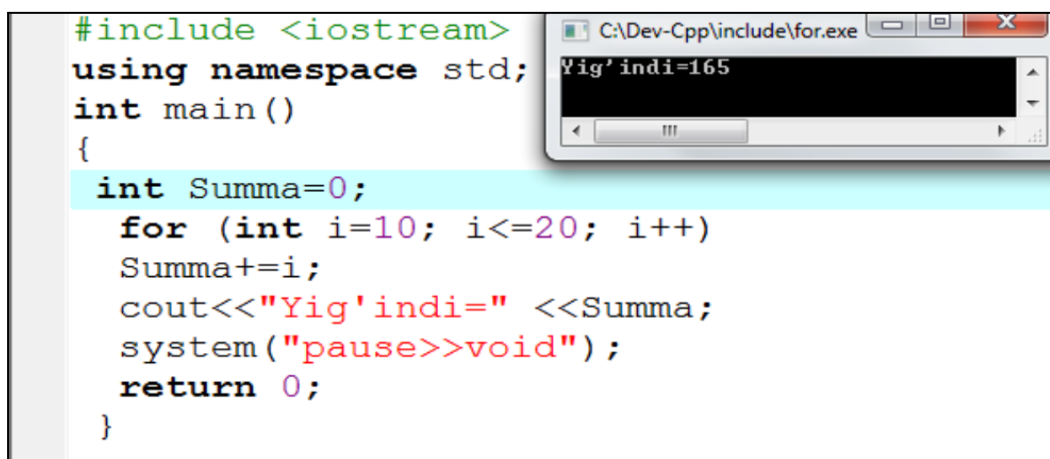
For, while va do while takrorlash operatorlari, **break, continue, goto** operatorlari.

Dastur bajarilishini boshqarishning boshqa bir kuchli mexanizmlaridan biri - takrorlash operatorlari hisoblanadi. Takrorlash operatori «takrorlash sharti» deb nomlanuvchi ifodaning rost qiymatida dasturning ma'lum bir qismidagi operatorlarni (takrorlash tanasini) ko'p marta takror ravishda bajaradi (itarativ jarayon). Takrorlash shartini tekshirish takrorlash tanasidagi operatorlarni bajarishdan oldin tekshirilishi mumkin (**for, while** takrorlashlari) yoki takrorlash tanasidagi operatorlari bir marta bajarilgandan keyin tekshirilishi mumkin (**do-while**). Takrorlash operatorlari ichma-ich joylashgan bo'lishi mumkin,.

for takrorlash operatori

for takrorlash operatorining sintaksisi qo'yidagi ko'rinishga ega:
for (<ifoda1>; <ifoda2>;<ifoda3>) <operator yoki blok>;
Bu operator o'z ishini <ifoda1> ifodasini bajarishdan boshlaydi. Keyin takrorlash qadamlari boshlanadi. Har bir qadamda <ifoda2> bajariladi, agar natija 0 qiymatidan farqli yoki *true* bo'lsa, takrorlash tanasi - <operator yoki blok> bajariladi va oxirida <ifoda3> bajariladi. Agar <ifoda2> qiymati 0 (*false*) bo'lsa, takrorlash jarayoni to'xtaydi va boshqaruv takrorlash operatoridan keyingi operatorga o'tadi. Shuni qayd qilish kerakki, <ifoda2> ifodasi vergul bilan ajratilgan bir nechta ifodalar birlashmasidan iborat bo'lishi mumkin, bu holda oxirgi ifoda qiymati takrorlash sharti hisoblanadi. Takrorlash tanasi sifatida bitta operator, jumladan bo'sh operator bo'lishi yoki operatorlar bloki kelishi mumkin.

Misol uchun 10 dan 20 gacha bo'lgan butun sonlar yig'indisini hisoblash masalasini ko'raylik.



9.1-rasm. 10 dan 20 gacha bo'lgan butun sonlar yig'indisini hisoblovchi dastur tuzilishi

Dasturdagi takrorlash operatori o'z ishini i takrorlash parametriga (takrorlash sanagichiga) boshlang'ich qiymat 10 sonini berishdan boshlaydi va har bir takrorlash qadamidan (itirasiyadan) keyin qavs ichidagi uchinchi operator bajarilishi hisobiga uning qiymati bittaga oshadi. Har bir takrorlash qadamida takrorlash tanasidagi operator bajariladi, ya'ni Summa o'zgaruvchisiga i qiymati qo'shiladi. Takrorlash sanagichi i qiymati 21 bo'lganda "i<=20" takrorlash sharti *false* (0-qiymati) bo'ladi va takrorlash tugaydi. Natijada boshqaruv takrorlash operatoridan keyingi *cout* operatoriga o'tadi va ekranga yig'indi chop etiladi. Yuqorida keltirilgan misolga qarab takrorlash operatorlarining qavs ichidagi ifodalariga izoh berish mumkin:

<ifoda1> - takrorlash sanagichi vazifasini bajaruvchi o'zgaruvchiga boshlang'ich qiymat berishga xizmat qiladi va u takrorlash jarayoni boshida faqat bir marta hisoblanadi. Ifodada o'zgaruvchi e'loni uchrashi mumkin va bu o'zgaruvchi takrorlash operatori tanasida amal qiladi va takrorlash operatoridan tashqarida «ko'rinmaydi».

<ifoda2> - takrorlashni bajarish yoki yo'qligini aniqlab beruvchi mantiqiy ifoda, agar shart rost bo'lsa, takrorlash davom etadi, aks holda yo'q. Agar bu ifoda bo'sh bo'lsa, shart doimo rost deb hisoblanadi;

<ifoda3> - odatda takrorlash sanagichining qiymatini oshirish (kamaytirish) uchun xizmat qiladi yoki unda takrorlash shartiga ta'sir qiluvchi boshqa amallar bo'lishi mumkin. Takrorlash operatorida qavs ichidagi ifodalar bo'lmasligi mumkin, lekin sintaksis ';' bo'lmasligiga ruxsat bermaydi. Shu sababli, eng sodda ko'rinishdagi takrorlash operatori quyidagicha bo'ladi:

for (;);cout <<"Cheksiz takrorlash...";

Agar takrorlash jarayonida bir nechta o'zgaruvchilarning qiymati sinxron ravishda o'zgarishi kerak bo'lsa, takrorlash ifodalarida zarur operatorlarni ',' bilan yozish orqali bunga erishish mumkin:

for(int i=10;j=2;i<=20;i++,j=i+10) {...};

Takrorlash operatorining har bir qadamida j va i o'zgaruvchilarning qiymatlari mos ravishda o'zgarib boradi. for operatorida takrorlash tanasi

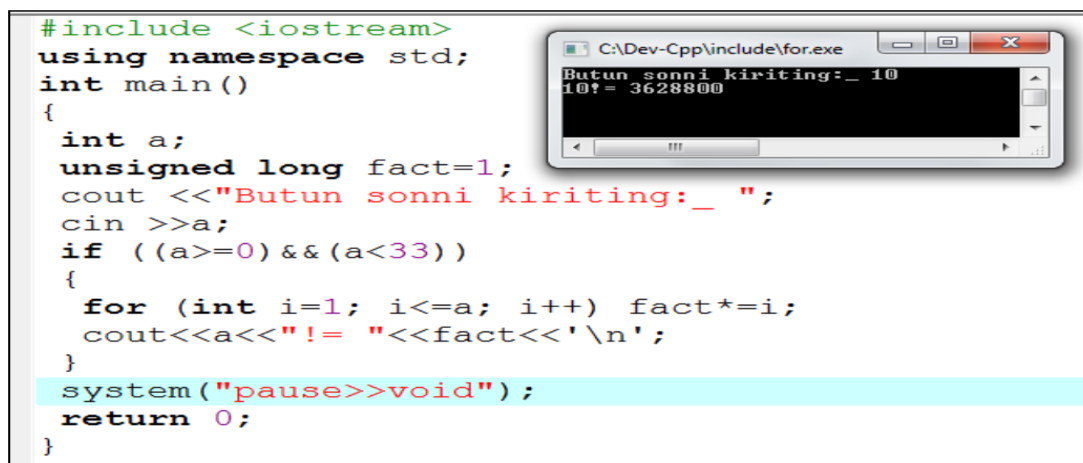
bo'lmasligi ham mumkin. Masalan, dastur bajarilishini ma'lum bir muddatga «to'xtab» turish zarur bo'lsa, bunga takrorlashni hech qanday qo'shimcha ishlarni bajarmasdan amal qilishi orqali erishish mumkin:

```
#include <iostream.h>
int main()
{int delay;
...
for(delay=5000;delay>0;delay--) ;//bo' shoperator
...
return 0;}
```

Yuqorida keltirilgan 10 dan 20 gacha bo'lgan sonlar yig'indisini bo'sh tanali takrorlash operatori orqali hisoblash mumkin:

```
...
for (int i=10; i<=20; Summa+=i++);
...
```

Takrorlash operatori tanasi sifatida operatorlar bloki ishlatilishini faktorialni hisoblash misolida ko'rsatish mumkin:



9.2-rasm. Takrorlash operatorining ishlatilishi

Dastur foydalanuvchi tomonidan 0 dan 33 gacha oraliqdagi son kiritilganda amal qiladi, chunki 34! qiymati unsigned long uchun ajratilgan razryadlarga sig'maydi.

while takrorlash operatori

while takrorlash operatori, operator yoki blokni takrorlash sharti yolg'on (false yoki 0) bo'lguncha takror bajaradi. U quyidagi sintaksisga ega:

while (<ifoda>) <operator yoki blok>;

Agar <ifoda> rost qiymatli o'zgarmas ifoda bo'lsa, takrorlash cheksiz bo'ladi. Xuddi shunday, <ifoda> takrorlash boshlanishida rost bo'lib, uning qiymatiga takrorlash tanasidagi hisoblash ta'sir etmasa, ya'ni uning qiymati o'zgarmasa, takrorlash cheksiz bo'ladi. while takrorlash shartini oldindan tekshiruvchi takrorlash operatori hisoblanadi. Agar takrorlash boshida <ifoda> yolg'on bo'lsa, while operatori tarkibidagi <operator yoki blok> qismi bajarilmasdan cheklab o'tiladi. Ayrim hollarda <ifoda> qiymat berish operatori

ko'rinishida kelishi mumkin. Bunda qiymat berish amali bajariladi va natija 0 bilan solishtiriladi.

Natija noldan farqli bo'lsa, takrorlash davom ettiriladi. Agar rost ifodaning qiymati noldan farqli o'zgarmas bo'lsa, cheksiz takrorlash ro'y beradi. Masalan:

while(1); // cheksiz takrorlash Xuddi **for** operatoridek, ',' yordamida <ifoda> da bir nechta amallar sinxron ravishda bajarish mumkin. Masalan, son va uning kvadratlari chop qiladigan dasturda ushbu holat ko'rsatilgan:

The screenshot shows a C++ IDE with two windows. The left window displays the source code of a program that uses a `while` loop to calculate and print the squares of numbers from 1 to 20. The right window shows the program's output, displaying each number and its square on a new line.

```
#include <iostream.h>
int main()
{
    int n,n2;
    cout<<"Sonni kiriting(1..10):_";
    cin>>n;
    n++;
    while(n--,n2=n*n,n>0)
    cout<<" n="<<n<<" n^2="<<n2<<endl;
    system("pause">>void");
    return 0;
}
```

```
Sonni kiriting(1..10):_20
n=20 n^2=400
n=19 n^2=361
n=18 n^2=324
n=17 n^2=289
n=16 n^2=256
n=15 n^2=225
n=14 n^2=196
n=13 n^2=169
n=12 n^2=144
n=11 n^2=121
n=10 n^2=100
n=9 n^2=81
n=8 n^2=64
n=7 n^2=49
n=6 n^2=36
n=5 n^2=25
n=4 n^2=16
n=3 n^2=9
n=2 n^2=4
n=1 n^2=1
```

9.3-rasm. while operatorining ishlatilishi

Dasturdagi takrorlash operatori bajarilishida `n` soni 1 gacha kamayib boradi. Har bir qadamda `n` va uning kvadrati chop qilinadi. Shunga e'tibor berish kerakki, shart ifodasida operatorlarni yozilish ketma-ketligining ahamiyati bor, chunki eng oxirgi operator takrorlash sharti sifatida qaraladi va `n` qiymati 0 bo'lganda takrorlash tugaydi. Keyingi dasturda berilgan o'nlik sonning ikkilik ko'rinishini chop qilish masalasini yechishda `while` operatorini qo'llash ko'rsatilgan.

The screenshot shows a C++ IDE with two windows. The left window displays the source code of a program that converts a decimal number to binary using a `while` loop. The right window shows the program's output, displaying the decimal number and its binary representation.

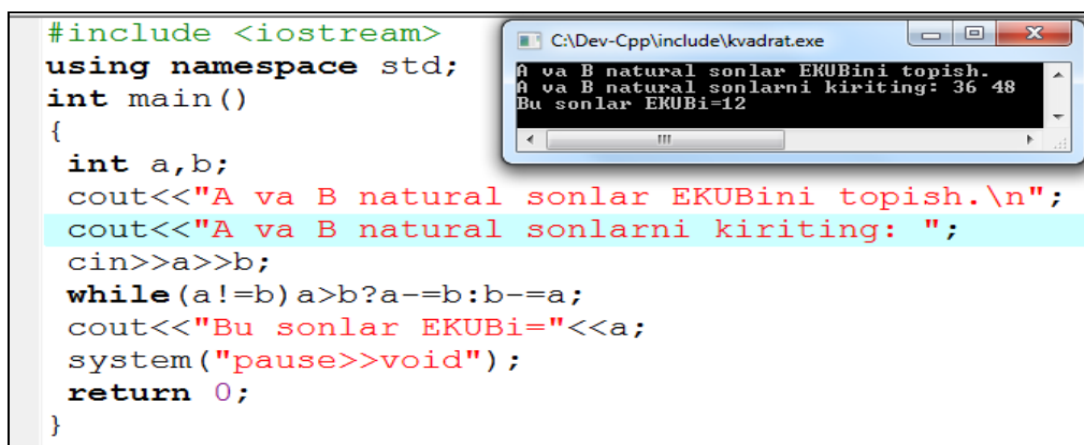
```
#include <iostream.h>
int main()
{
    int sanagich=4;
    short son10,jarayon=1;
    while (jarayon) // cheksiz takrorlash
    {cout <<"O'nlik sonni kiriting(0..15):_";
    cin >>son10;
    cout<<"\n"<<son10<<" Sonining ikkilik ko'rinishi: ";
    while (sanagich)
    {if(son10 & 8) //son10 & 00001000
    cout<<'1';
    else cout<<'0';
    son10=son10<<1; //razryadlarni 1 o'rin chapga surish
    sanagich--;}
    cout<<"\n";
    cout<<"Jarayonni to'xtasin(0),davom etsin(1):_ ";
    cin >> jarayon;
    sanagich=4;
    }
    return 0;
}
```

```
O'nlik sonni kiriting(0..15):_14
14 Sonining ikkilik ko'rinishi: 1110
Jarayonni to'xtasin(0),davom etsin(1):_ 1
O'nlik sonni kiriting(0..15):_10
10 Sonining ikkilik ko'rinishi: 1010
Jarayonni to'xtasin(0),davom etsin(1):_
```

9.4-rasm. while operatorining ishlatilishi

Dasturda ichma-ich joylashgan takrorlash operatorlari ishlatilgan. Birinchisi, sonning ikkilik ko'rinishini chop qilish jarayonini davom ettirish sharti bo'yicha amal qiladi. Ichki joylashgan ikkinchi takrorlash operatoridagi amallar - har qanday, 0 dan 15 gacha bo'lgan sonlar to'rtta razryadli ikkilik son ko'rinishida bo'lishiga asoslangan. Unda

kiritilgan sonning ichki, ikkilik ko'rinishida uchinchi razryadida 0 yoki 1 turganligi aniqlanadi ("son10&8"). Shart natijasi natija 1 (rost) bo'lsa, ekranga '1', aks holda '0' belgisi chop etiladi. Keyingi qadamda son razryadlari chapga bittaga suriladi va yana uchinchi razryaddagi raqam chop etiladi. Takrorlash sanagich qiymati 0 bo'lguncha ya'ni to'rt marta bajariladi va boshqaruv ichki takrorlash operatoridan chiqadi. *while* ta krorlash operatori yordamida samarali dastur kodi yozishga yana bir misol bu - ikkita natural sonlarning eng katta umumiy bo'luvchisini (EKUB) Evklid algoritmi bilan topish masalasini keltirishimiz mumkin:



9.5-rasm. Ichma-ich joylashgan takrorlash operatorining ishlatilishi

Butun turdagi *a* va *b* qiymatlari oqimdan o'qilgandan keyin toki ularning qiymatlari o'zaro teng bo'lmaguncha takrorlash jarayoni ro'y beradi. Takrorlashning har bir qadamida *a* va *b* sonlarning kattasidan kichigi ayriladi. Takrorlashdan keyingi ko'rsatma vositasida *a* o'zgaruvchisining qiymati natija sifatida chop etiladi.

***do-while* takrorlash operatori**

do-while takrorlash operatori *while* operatoridan farqli ravishda oldin operator yoki blokni bajaradi, keyin takrorlash shartini tekshiradi. Bu qurilma takrorlash tanasini kamida bir marta bajarilishini ta'minlaydi. *do-while* takrorlash operatori quyidagi sintaksisga ega:

do <operator yoki blok>; *while* (<ifoda>);

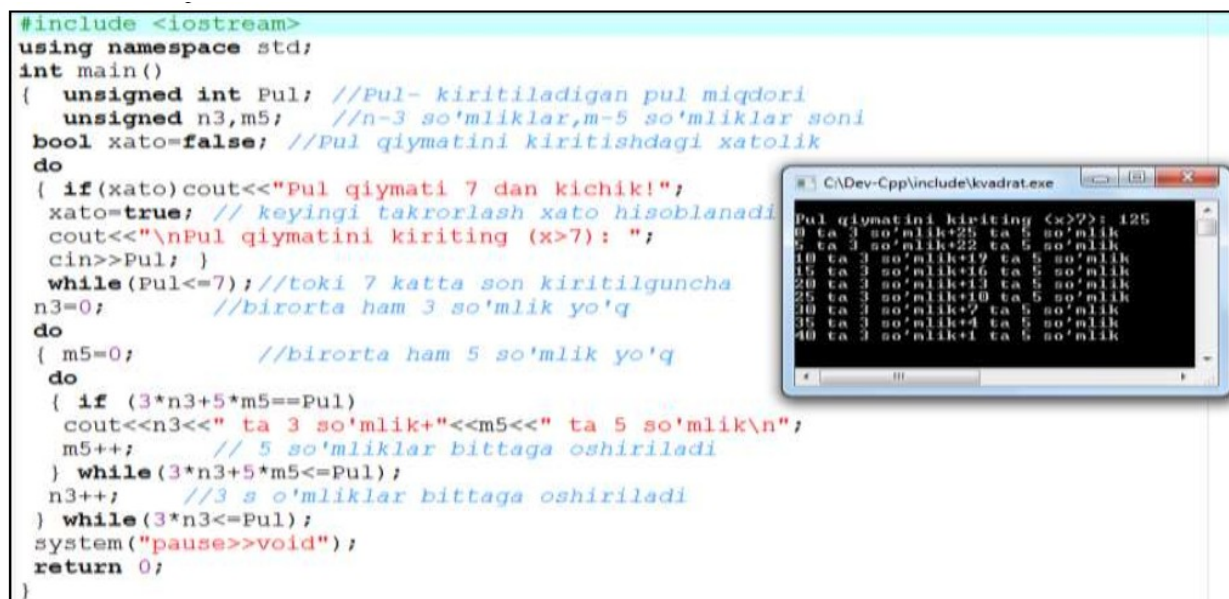
Bunday takrorlash operatorining keng qo'llaniladigan holatlari - bu takrorlashni boshlamasdan turib, takrorlash shartini tekshirishning iloji bo'lmagan holatlar hisoblanadi. Masalan, birorta jarayonni davom ettirish yoki to'xtatish haqidagi so'rovga javob olish va uni tekshirish zarur bo'lsin. Ko'rinib turibdiki, jarayonni boshlamasdan oldin bu so'rovni berishning ma'nosi yo'q. Hech bo'lmaganda takrorlash jarayonining bitta qadami amalga oshirilgan bo'lishi kerak:

```
#include <iostream.h>
int main()
{
    char javob;
    do
    {
        ... // dastur tanasi
        cout<<"Jarayonni to'xtatish (N):_ ";
        cin>>javob;
    } while(javob !='N')
    return 0;
}
```

Dastur toki "Jarayonni to'xtatish (N):_ " so'roviga 'N' javobi kiritilmaguncha davom etadi. Bu operator ham cheksiz takrorlanishi mumkin:

do; while(1);

Masala. Har qanday 7 dan katta butun sondagi pul miqdorini 3 va 5 so'mliklarda berish mumkinligi isbotlansin. Qo'yilgan masala $p=3n+5m$ tenglamasi qanoatlantiruvchi m , n sonlar juftliklarini topish masalasidir (p -pul miqdori). Bu shartning bajarilishini m va n o'zgaruvchilarining mumkin bo'lgan qiymatlarining barcha kombinasiyalarida tekshirish zarur bo'ladi.



9.6-rasm. do while operatorining ishlatilishi

Dastur pul qiymatini kiritishni so'raydi (Pul o'zgaruvchisiga). Agar pul qiymati 7 sonidan kichik bo'lsa, bu haqda xabar beriladi va takror ravishda qiymat kiritish talab qilinadi. Pul qiymati 7 dan katta bo'lganda 3 va 5 so'mliklarning mumkin bo'lgan to'la kombinasiyasi amalga oshirish uchun ichma-ich takrorlashlar amalga oshiriladi. Tashqi takrorlash $n3$ (3 so'mliklar miqdori) bo'yicha ichki takrorlash esa $m5$ (5 so'mliklar miqdori) bo'yicha, toki bu miqdordagi pullar qiymati Pul qiymatidan oshib ketmaguncha davom etadi. Ichki takrorlashda $m5$ o'zgaruvchisining har bir qiymatida « $3*n3+5*m5==Pul$ » sharti

tekshiriladi, agar u o'rinli bo'lsa, yechim variant sifatida n3 va m5 o'zgaruvchilar qiymatlari chop etiladi.

Break operatori. Takrorlash operatorlarining bajarilishida shunday holatlar yuzaga kelishi mumkinki, unda qaysidir qadamda takrorlashni yakuniga yetkazmasdan takrorlashdan chiqish zarurati bo'lishi mumkin.

Boshqacha aytganda, takrorlashni «uzish» kerak bo'lishi mumkin. Bunda break operatoridan foydalaniladi. break operatorini takrorlash operatori tanasining ixtiyoriy (zarur) joylariga qo'yish orqali shu joylardan takrorlashdan chiqishni amalga oshirish mumkin. E'tibor beradigan bo'lsak, switch-case operatorining tub mohiyatiga ham break operatorini qo'llash orqali erishilgan. Ichma - ich joylashgan takrorlash va switch operatorlarida break operatori faqat o'zi joylashgan blokdan chiqish imkoniyatini beradi.

Quyidagi programmada ikkita ichma-ich joylashgan takrorlash operatoridan foydalangan holda foydalanuvchi tomonidan kiritilgan qandaydir sonni 3 va 7 sonlariga nisbatan qanday oraliqqa tushishi aniqlanadi. Tashqi takrorlashda “Son kiriting (0- to'xtash):_” so'rovi beriladi va javob javob_son o'zgaruvchisiga o'qiladi. Agar son noldan farqli bo'lsa, ichki takrorlash operatorida bu sonning qandaydir oraliqqa tushishi aniqlanib, shu haqida xabar beriladi va ichki takrorlash operatoridan chiqiladi. Tashqi takrorlashdagi so'rovga javob tariqasida 0 kiritilsa, programma o'z ishini tugatadi.

```
#include <iostream.h>
int main ()
{
    int javob_son=0;
    do
    {
        while (javob_son)
        {
            if (javob_son<3)
            {cout<<"3 kichik ! "; break;}
            if ( 3<=javob_son && javob_son <=7)
            {cout<<"3 va 7 oralig'ida ! "; break;}
            if (javob_son >7 )
            {cout<<"7 dan katta ! "; break;}
        }
        cout<<"\nSon kiriting ( 0 – to'xtash) :_";

        cin >> javob_son;
    }
    while (javob_son !=0)
    return 0;
}
```

Amaliyotda break operatoridan cheksiz takrorlashdan chiqishda foydalaniladi.

```
for ( ; ; )
{
// 1 – shart
if (....)
{
....
break;
}
// 2-shart
if ( ... )
{
....
break;
}
...
}
```

Bu misolda cheksiz for takrorlashidan 1 yoki 2 - shart bajarilganda chiqiladi.

Masala. Ishorasiz butun sonlar ketma-ketligi 0 qiymati bilan tugaydi, 0 ketma-ketlik hadi hisoblanmaydi. Ketma-ketlikni kamaymaydigan holda tartiblangan yoki yo'qligi aniqlansin.

```
#include <iostream.h>
int main()
{
    unsigned int Ai_1=0,Ai;
    cout<<"Sonlar ketma-ketligini kiriting"
    cout<<(0-tugash alomati):\n ";
    cin>>Ai;        // ketma-ketlikning birinchi hadi
    while (Ai)
    {
        Ai_1=Ai;
        cin>>Ai;        // navbatdagi had
        if (Ai_1>Ai) break;
    }
    if (Ai_1)
    {
        cout<<"Ketma-ketlik tartiblangan";
        if (!Ai) cout<<" emas!";
        else cout<<"!";
    }
    else cout<<"Ketma-ketlik bo'sh!";
    return 0;
}
```

Dastur ishga tushganda, boshda ketma-ketlikning birinchi hadi alohida o'qib olinadi (A_i o'zgaruvchisiga). Keyin A_i qiymati nolga teng bo'lmaguncha takrorlash operatori amal qiladi. Takrorlash tanasida A_i qiymati oldingi qiymat sifatida A_{i-1} o'zgaruvchisida eslab qolinadi va navbatdagi had A_i o'zgaruvchisiga o'qiladi. Agar oldingi had navbatdagi haddan katta bo'lsa, break operatori yordamida takrorlash jarayoni uziladi va boshqaruv takrorlashdan keyingi shart operatoriga o'tadi. Bu yerdagi shart operatorlari mazmuni quyidagicha: agar A_{i-1} noldan farqli bo'lsa, ketma-ketlikning kamida bitta hadi kiritilgan bo'ladi (ketma-ketlik mavjud) va oxirgi kiritilgan had tekshiriladi. O'z navbatida agar A_i noldan farqli bo'lsa, bu holat hadlar o'rtasida kamaymaslik sharti bajarilmaganligi sababli hadlarni kiritish jarayoni uzilganini bildiradi va bu haqda xabar chop etiladi. Aks holda ketma-ketlikni kamaymaydigan holda tartiblangan bo'ladi.

continue operatori

continue operatori xuddi break operatoridek takrorlash operatori tanasini bajarishni to'xtatadi, lekin takrorlashdan chiqib ketmasdan keyingi qadamiga «sakrab» o'tishini tayinlaydi. continue operatorini qo'llanishiga misol tariqasida 2 va 50 sonlar oralig'idagi tub sonlarni topadigan programma matnini keltiramiz.

```
#include <iostream.h>
int main()
{
    bool bulinadi=false;
    for (int i=2; i<50; i++)
    {
        for (int j=2; j<i/2; j++)
        {
            if (i%j) continue;
            bulinadi=true;
            break;
        }
    }
    // break bajarilganda boshqaruv o'tadigan joy

    if (bulinadi) cout <<i>i<<" ";
}
return 0;
}
```

Keltirilgan programmada qo'yilgan masala ichma-ich joylashgan ikkita takrorlash operatorlari yordamida yechilgan. Birinchi takrorlash operatori 2 dan 50 gacha sonlarni hosil qilishga xizmat qiladi. Ichki takrorlash esa har bir hosil qilinayotgan sonni 2 sonidan toki shu sonning yarmigacha bo'lgan sonlarga bo'lib, qoldig'ini tekshiradi, agar qoldiq 0 sonidan farqli bo'lsa, navbatdagi songa bo'lish davom etadi, aks holda bulinadi o'zgaruvchisiga true qiymat berib, ichki takrorlash uziladi (son o'zining yarmigacha bo'lgan qandaydir songa bo'linar

ekan, demak u tub emas va keyingi sonlarga bo'lib tekshirishga hojat yo'q). Ichki j bo'yicha takrorlashdan chiqqandan keyin bulinadi qiymati false bo'lsa (!bulinadi), i soni tub bo'ladi va u chop qilinadi.

goto operatori va nishonlar

Nishon - bu davomida ikkita nuqta (':') qo'yilgan identifikator. Nishon bilan qandaydir operator belgilanadi va keyinchalik, programmaning boshqa bir qismidan unga shartsiz o'tish amalga oshiriladi. Nishon bilan har qanday operator belgilanishi mumkin, shu jumladan e'lon operatori va bo'sh operatori ham. Nishon faqat funksiyalar ichida amal qiladi. Nishonga shartsiz o'tish goto operatori yordamida bajariladi. goto operatori orqali faqat uning o'zi joylashgan funksiya ichidagi operatorlarga o'tish mumkin. goto operatorining sintaksisi quyidagicha:

`goto <nishon>;`

Ayrim hollarda, goto operatorining «sakrab o'tishi» hisobiga xatoliklar yuzaga kelishi mumkin. Masalan,

```
int i=0;  
i++; if (i) goto m;  
int j;  
m: j+=i;
```

operatorlarining bajarilishi xatolikka olib keladi, chunki j e'lon qilinmay qoladi. Shartsiz o'tish operatori programmani tuzishdagi kuchli va shu bilan birgalikda xavfli vositalardan biri hisoblanadi. Kuchliligi shundaki, uning yordamida algoritmning «boshi berk» joylaridan chiqib ketish mumkin. Ikkinchi tomondan, bloklarning ichiga o'tish, masalan, takrorlash operatorlarini ichiga «sakrab» kirish kutilmagan holatlarni yuzaga keltirishi mumkin. Shu sababli, imkon qadar goto operatoridan foydalanmaslik kerak, ishlatilgan taqdirda ham quyidagi qoidaga amal qilish zarur: «blok ichiga, if...else va switch operatorlari ichiga, hamda takrorlash operatorlari tanasiga tashqaridan kirish mumkin emas».

Garchi, nishon yordamida programmaning ixtiyoriy joyiga o'tish mumkin bo'lsa ham, boshlang'ich qiymat berish e'lonlaridan sakrab o'tish man etiladi, lekin bloklardan sakrab o'tish mumkin.

Masalan:

```
....  
goto B;      \\ xato  
float x=0.0;  
goto B;      \\ to'g'ri  
{ int n=10; x=n*x+x; }  
B: cout << "x="<<x;  
....
```

Xususan, nishon yordamida ichki blokdan tashqi blokka va tashqi blokdan ichki blokka o'tishga C++ tili ruxsat beradi:

```
{....
    goto ABC:
    ....
    { int i=15;
    ....
    ABC:
    ....
    goto XYZ;
    int y=10;
    ...
    XYZ:
    ....

    goto KLM;
    ...}
    ...
    int k=0;
    ...
    KLM:
    ...
}
```

Lekin, yuqorida keltirilgan misoldagi barcha o'tishlar mazmunan xato hisoblanadi. Quyidagi programmada ikkita natural sonlar eng katta umumiy bo'luvchini (EKUB) topish masalasidagi takrorlash jarayonini nishon va goto operatori vositasida amalga oshirish ko'rsatilgan:

```
int main()
{
    int a,b;
    cout<<"A va B natural sonlar EKUBini
topish.\n";
    cout<<"A va B natural sonlarni kiriting: "
    cin>>a>>b;
    nishon:
    if(a==b)
    {
        cout<<"Bu sonlar EKUBi: "<<a;
        return 0;
    }
    a>b?a-=b:b-=a;
    goto nishon;
}
```

dasturdagi nishon bilan belgilangan operatorlarda a va b sonlarni tengligi tekshiriladi. Agar ular teng bo'lsa, ixtiyoriy bittasi, masalan a soni

EKUB bo'ladi va funksiyadan chiqiladi. Aks holda, bu sonlarning kattasidan kichigi ayriladi va goto orqali ularning tengligi tekshiriladi. Takrorlash jarayoni a va b sonlar o'zaro teng bo'lguncha davom etadi.

Shuni qayd etish kerakki, bu masalani takrorlash operatorlari yordamida bajarish ancha samarali hisoblanadi.

Berilgan masala uchun algoritm.

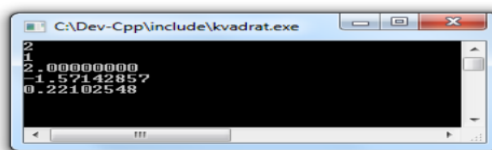
Berilgan masalada uchta A, B va C misollar mavjud. A misol: k! hisoblash, ya'ni $1 \cdot 2 \cdot \dots \cdot k$ ko'paytma natijasini olish lozim. Bu yerda A=1 haqiqiy o'zgaruvchi olamiz va k ta takrorlash natijasida 1 dan k gacha bo'lgan sonlarni A ga ko'paytiramiz.

B misol: B=0 haqiqiy o'zgaruvchi olamiz. $\frac{10n-30}{10n^2-3n+8}$ ifodaning natijasini n=1 dan toki n=k gacha almashtirib hisoblaymiz va har birining natijasini B ga qo'shamiz.

C misol: Ushbu misol ichma-ich sikl bo'lganligi sababli C=0 va C1=0 haqiqiy qiymatli o'zgaruvchi olamiz. Tashqi siklning har bir holati uchun ichki sikl hisoblanib C1 ga qo'shiladi va natija C ga qo'shiladi.

Asosiy dastur matni.

```
#include <iostream>
#include <math.h>
using namespace std;
int main()
{
    double A=1,B=0,C=0,C1=0;
    __int64 i,k,p,n,m,r;
    cin>>k>>p;
    for (i=1;i<=k;i++)
        A*=i;
    for (n=1;n<=k;n++)
        B+=(10*n-30)*1.0/(10*n*n-3*n+8);
    for (r=1;r<=k;r++)
    {C1=0;
    for (m=1;m<=p;m++)
        C1+=sqrt(pow(r+m,2)+5*r)/(r+pow(m,3)+pow(2,m-r)+30);
        C+=C1;
    }
    cout.precision(8);
    cout<<fixed<<A<<"\n"<<B<<"\n"<<C;
    system("pause");
    return 0;
}
```



9.7-rasm. ichma-ich siklar ishlatilishi

Berilgan misol ishlanish natijalari tahlili.

k! faktorialni hisoblash **for (i=1;i<=k;i++) A*=i;** bu yerda **i=1** siklning initsializatsiya qismi, ya'ni i o'zgaruvchi 1 qiymatini o'zlashtiradi. **i<=k** siklning shart qismi, ushbu shart bajarilmaguncha davom etadi. **i++** siklning qadamlar qismi, ya'ni har safar A ga ko'paytirilib bir birlikka oshiriladi. Ikkinchi va uchinchi misolni hisoblashda ham sikldan huddi shu uslubda foydalanamiz.

Natija

```

C:\Dev-Cpp\include\kvadrat.exe
2 1
2.000000000
-1.57142857
0.22102548_
  
```

Topshiriqlar. Quyidagi variantlarda berilgan takrorlanuvchi algoritmlar dasturini tuzing.

N _o	Variantlar	N _o	Kirish	Chiqish
1	a) $\sum_{n=1}^m \frac{1}{n^3}$; b) $\sum_{R=1}^l \frac{R^3}{R^4 + 3R^2 + e^{-R}}$; c) $\prod_{R=1}^l \prod_{i=1}^m \frac{R^i + 1}{R^4 + 3^i - R + e^{-R}}$;	1	$m=2$ $l=1$	a)1.12500000 b)0.22894405 c)0.08832821
		2	$m=5$ $l=8$	a)1.18566204 b)1.56011635 c)0.00000000
2	a) $\sum_{n=1}^k \frac{2}{n^3(n+1)}$; b) $\sum_{R=1}^y \frac{R^2 + R-2 }{\ln R + 3R}$; c) $\sum_{R=1}^k \sum_{m=1}^p \frac{R^2 + R-2 }{\ln R + 3R}$;	1	$k=2$ $p=1$	a)1.08333333 b)0.66666667 c)2.55180630
		2	$k=5$ $p=8$	a)1.11076852 b)12.34348563 c)44.68083491
3	a) $\sum_{n=1}^k \frac{3}{(2n+1)^3}$; b) $\sum_{R=1}^p \frac{R+1}{R + e^{-R} + 1}$; c) $\prod_{R=1}^k \sum_{i=1}^p \frac{R+3}{R^3 + 3R + i^3}$;	1	$k=2$ $p=1$	a)0.13511111 b)0.90524345 c)0.26666667
		2	$k=5$ $p=8$	a)0.15022664 b)169.36763033 c)0.06609173
4	a) $\sum_{R=1}^k \frac{4}{R(R+1)}$; b) $\sum_{R=1}^p \frac{R^{R+1}}{2^{R+1} + (R+1)^4}$; c) $\sum_{R=1}^k \prod_{i=1}^p \frac{(R+1)^i + 4}{(-1)^R + 3(-1)^i + i^R}$;	1	$k=2$ $p=1$	a)2.66666667 b)0.05000000 c)-9.00000000
		2	$k=5$ $p=8$	a)3.33333333 b)20424.87608993 c)-204508918.09011152
5	a) $\sum_{m=1}^k \frac{5}{m^2 + m + 4}$; b) $\sum_{R=1}^p \frac{(100-R)^2}{\lg R + 5^{-R}}$; c) $\sum_{i=1}^k \sum_{R=1}^p \frac{(-1)^i \cos(i+R) + 5}{5i + 7^{-R} + i^{-R}}$;	1	$k=2$ $p=1$	a)1.33333333 b)49005.00000000 c)1.25847755
		2	$k=5$ $p=8$	a)2.00122549 b)155695.30188719 c)16.88785843
6	a) $\prod_{n=1}^k \frac{n+6}{n^2 + 4n + 1}$; b) $\sum_{i=1}^p \frac{i+6}{i^4 + 27i + 7}$;	1	$k=2$ $p=1$	a)0.71794872 b)0.20000000 c)15.21598260

	$c) \prod_{R=1}^k \prod_{i=1}^p (-1)^i \frac{\sqrt{5i^4 + e^{-R} + 6}}{\cos(i+1)^3 - R^{-i}};$	2	$k=5$ $p=4$	a)0.02128306 b)0.38410472 c)5851732838696673300000 000000.00000000
7	$a) k! + 7; \quad b) \sum_{i=1}^p \frac{(-1)^i \cdot 7^{-i}}{1+i+i^2};$ $c) \sum_{i=1}^k \sum_{R=1}^p \frac{(-1)^i \cos(i+R) + 5}{5i + 7^{-R} + i^{-R}};$	1	$k=2$ $p=1$	a)9.00000000 b)-0.0476190 c)1.2584776
		2	$k=5$ $p=8$	a)127.00000000 b)-0.0449098 c)16.8878584
8	$a) (k+1)!; \quad b) \sum_{n=1}^p \frac{10n-8}{10n^2-3n+8};$ $c) \sum_{i=1}^k \sum_{m=1}^p \left[\frac{i^m + 4m + e^m}{m^i} \right];$	1	$k=2$ $p=1$	a)6.00000000 b)0.13333333 c)16.43656366
		2	$k=5$ $p=8$	a)720.00000000 b)1.43814142 c)942.46222819
9	$a) \prod_{n=1}^k \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-2}}}; \quad b) \sum_{i=1}^p (2i + 5i + 9);$ $c) \sum_{R=1}^k \prod_{m=1}^p \sqrt{\left \frac{R + m^3 + e^{-m} + 9}{\log_2 R + (mR)^3} \right };$	1	$k=2$ $p=1$	a)16.47419724 b)16.00000000 c)4.54389514
		2	$k=5$ $p=8$	a)9644244424.99799920 b)324.00000000 c)6.96548965
10	$a) \prod_{n=1}^k \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-n}}}; \quad b) \prod_{n=1}^p \frac{n^2 + 3n + 10}{n^2 + 7n + 91};$ $c) \sum_{R=1}^k \sum_{m=1}^p \frac{\sqrt{tg(R+m)^2 + 10R}}{R + m^{-R} + e^{m-R}};$	1	$k=2$ $p=1$	a)16.47419724 b)3.02633015 c)2.42622166
		2	$k=5$ $p=8$	a)9644244424.99799920 b)13252651.75703296 c)28.57163344
11	$a) \prod_{n=1}^k \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-n}}}; \quad b) \prod_{n=1}^p \frac{1}{n^4 + 1};$ $c) \prod_{i=1}^k \sum_{m=1}^p \left[e^{\sqrt{i^2+m-i}} + \frac{i^2 + 11}{m^4 + i^{-m}} \right];$	1	$k=2$ $p=1$	a)16.7419724 b)0.50000000 c)136.46058442
		2	$k=5$ $p=8$	a)9644244424.99799920 b)0.00000000 c)1293947622309.52320000
12	$a) \prod_{i=1}^k \frac{i^4 + i^2 + 12}{\sqrt{i^3 + e^{-i}}}; \quad b) \sum_{R=1}^p \frac{R + 1.125}{R^5 + 5R + 1.2}$ $c) \sum_{m=1}^k \prod_{n=1}^p \sqrt{\left \frac{m^3 - n^2 + 3.4}{m^{-n} - n^{-m} + 12} \right };$	1	$k=2$ $p=1$	a)134.29701962 b)0.29513889 c)1.48326269
		2	$k=5$ $p=8$	a)5535312.16043921 b)0.39191141 c)5141.72021009
13	$a) \prod_{n=1}^k \frac{13}{n^3 + 5n + 7}; \quad b) \sum_{m=1}^p \frac{(-1)^m \sqrt{m}}{2^{-2m}};$	1	$k=2$ $p=1$	a)0.52000000 b)-4.00000000 c)-0.00258075
		2	$k=5$	a)0.00163191

	$c) \prod_{i=1}^k \prod_{m=1}^p \operatorname{tg} \frac{i^{-m} - i^{-3} - i^2 + 1.3}{m^{-i} + m^{-6} + im + 13}$		$p=8$	$b)150178.96310968$ $c)0.00000000$
14	$a) \sum_{R=1}^k \frac{R^2 + 14}{\sqrt[3]{3^{-R} + R^3}}; \quad b) \prod_{n=1}^p \frac{n + 1.23}{n + \frac{1}{n}};$ $c) \sum_{i=1}^k \sum_{m=1}^p \lg \frac{\sqrt{m^2 + e^{m-i}}}{i^2 + 2^{m-1}};$	1	$k=2$ $p=1$	$a)19.31060301$ $b)1.11500000$ $c)-0.78146109$
		2	$k=5$ $p=8$	$a)30.97176927$ $b)4.13609486$ $c)-28.76105925$
15	$a) \prod_{i=1}^k \frac{ i - 15 + i^3}{\ln i + 7i}; \quad b) \sum_{R=1}^p \frac{(-1)^R \cdot (R + 1)}{R^3 + R^2 + 1}$ $c) \prod_{n=1}^k \sum_{m=1}^p (-1) \frac{m \ln(n + 5) + 1.5}{2^{m+3} + (n + 3)^{-m} + nm};$	1	$k=2$ $p=1$	$a)3.06265223$ $b)-0.66666667$ $c)0.03451403$
		2	$k=5$ $p=8$	$a)50.86914472$ $b)-0.49910258$ $c)-0.07829932$
16	$a) \sum_{i=-22}^k \frac{\sqrt{ i - 2i^3 } + 16}{\ln i + 3 + 1.6};$ $b) \sum_{n=1}^p (-1)^n \frac{n + 1.258974}{2n^4 + 1};$ $c) \prod_{R=4}^k \prod_{m=1}^p \frac{\sqrt{ R^m + 4R - m + 1.6 }}{\sin(m + R) - m};$	1	$k=2$ $p=1$	$a)462.95921847$ $b)-0.75299133$ $c)1.00000000$
		2	$k=5$ $p=8$	$a)9644244424.99799920$ $b)0.00000000$ $c)1293947622309.52320000$
17	$a) \prod_{R=1}^k \frac{R + 17}{2R^2 + 9};$ $b) \sum_{R=1}^p (-1) \cdot \frac{R \cdot \sqrt[R]{R + 1} + R^2}{2R^2 + 4R + 11};$ $c) \sum_{m=1}^k \prod_{n=1}^p \sqrt{\frac{m^3 - n^2 + 1.7}{m^n + m^m + 12}};$	1	$k=2$ $p=1$	$a)1.82887701$ $b)-0.17647059$ $c)1.04368781$
		2	$k=5$ $p=8$	$a)0.25873599$ $b)-2.83439919$ $c)0.51638448$
18	$a) \sum_{n=1}^k \frac{18}{5 - 17n + n^3};$ $b) \prod_{m=-12}^p \frac{m^2 \sqrt{ m + 1.8}}{m^2 + 4m + (-1)^m};$ $c) \sum_{i=1}^k \prod_{R=1}^p \frac{\sqrt{e^{i+R} (i + R)^R}}{ 4i^3 + R^4 };$	1	$k=2$ $p=1$	$a)-2.49350649$ $b)0.41833001$ $c)1.00407388$
		2	$k=5$ $p=8$	$a)14.95912509$ $b)2.78668250$ $c)24681022165.89471400$
19	$a) \sum_{n=1}^k \frac{19n}{3 + n + n^2};$ $b) \sum_{t=3}^p \frac{\operatorname{tg}(t + 3)}{t^3 + 2t + e^{t-1}};$	1	$k=2$ $p=1$	$a)8.02222222$ $b)0.00000000$ $c)1.00000000$
		2	$k=5$ $p=8$	$a)18.00535793$ $b)-0.17303251$ $c)9737209203758591500000$

	$c) \prod_{i=1}^k \prod_{m=2}^p (i \cdot m)^{\frac{i\sqrt{ i } + m}{(i+m)}}$			000000000000.00000000
20	$a) \prod_{n=1}^k (-1)^n \frac{1+n^2}{1+n^3};$ $b) \sum_{m=10}^p \frac{\sin(m)}{\sqrt[4]{m^2 + e^{m+13}}};$ $c) \prod_{n=1}^k \prod_{R=1}^p \frac{n^3 - R^2 + 20}{(n-R + n)^R};$	1	$k=2$ $p=1$	a)-0.55555556 b)0.00000000 c)180.00000000
		2	$k=5$ $p=18$	a)-0.25360195 b)-0.00392117 c)0.00000000

10-laboratoriya ishi. C++ algoritmik tilning bir o'lchamli va ko'p o'lchamli massivlari.

Ishdan maqsad: C++ algoritmik tilning bir o'lchamli va ko'p o'lchamli massivlari bilan ishlash.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. Massivlarning xossalarini o'rganish.
3. Massivlarni e'lon qilish.
4. Massivlarni ta'riflash.
5. Bir o'lchovli massivlar bilan ishlash.
6. Tajriba ishining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Bir o'lchovli massivlarni e'lon qilish va ular ustida amallar.

Xotirada ketma-ket (regulyar) joylashgan bir xil turdagi qiymatlarga massiv deyiladi. Odatda massivlarga zarurat, katta hajmdagi, lekin cheklangan miqdordagi va tartiblangan qiymatlarni qayta ishlash bilan bog'liq masalalarni yechishda yuzaga keladi. Faraz qilaylik, talabalar guruhining reyting ballari bilan ishlash masalasi qo'yilgan. Unda guruhning o'rtacha reytingini aniqlash, reytinglarni kamayishi bo'yicha tartiblash, konkret talabaning reytingi haqida ma'lumot berish va boshqa masala ostilarini yechish zarur bo'lsin. Qayd etilgan masalalarni yechish uchun berilganlarning (reytinglarning) tartiblangan ketma-ketligi zarur bo'ladi. Bu yerda tartiblanganlik ma'nosi shundaki, ketma-ketlikning har bir qiymati o'z o'rniga ega bo'ladi (birinchi talabaning reytingi massivda birinchi o'rinda, ikkinchi talabaniki - ikkinchi o'rinda va hakoza). Berilganlar ketmaketligini ikki xil usulda hosil qilish mumkin. Birinchi yo'l - har bir reyting uchun alohida o'zgaruvchi aniqlash: $Reyting_1, \dots, Reyting_N$.

Lekin, guruhdagi talabalar soni yetarlicha katta bo'lganda, bu o'zgaruvchilar qatnashgan programmani tuzish katta qiyinchiliklarni yuzaga keltiradi. Ikkinchi yo'l - berilganlar ketma-ketligini yagona nom bilan aniqlab, uning qiymatlariga murojaatni, shu qiymatlarning

ketma-ketlikda joylashgan o'rnining nomeri (indeksi) orqali amalga oshirishdir. Reytinglar ketma-ketligini Reyting deb nomlab, undagi qiymatlariga Reyting1,...,ReytingN ko'rinishida murojaat qilish mumkin. Odatda berilganlarning bunday ko'rinishiga massivlar deyiladi. Massivlarni matematikadagi sonlar vektoriga o'xshatish mumkin, chunki vektor ham o'zining individual nomiga ega va u fiksirlangan miqdordagi bir turdagi qiymatlardan - sonlardan iboratdir.

Demak, massiv - bu fiksirlangan miqdordagi ayrim qiymatlarning (massiv elementlarining) tartiblangan majmuasidir. Barcha elementlar bir xil turda bo'lishi kerak va bu tur element turi yoki massiv uchun tayanch tur deb nomlanadi. Yuqoridagi keltirilgan misolda Reyting - haqiqiy turdagi vektor deb nomlanadi.

Programmada ishlatiladigan har bir konkret massiv o'zining individual nomiga ega bo'lishi kerak. Bu nomni to'liq o'zgaruvchi deyiladi, chunki uning qiymati massivning o'zi bo'ladi. Massivning har bir elementi massiv nomi, hamda kvadrat qavsga olingan va element selektori deb nomlanuvchi indeksni ko'rsatish orqali oshkor ravishda belgilanadi. Murojaat sintaksisi:

< massiv nomi > [< indeks >]

Bu ko'rinishga xususiy o'zgaruvchi deyiladi, chunki uning qiymati massivning alohida elementidir. Bizning misolda Reyting massivining alohida komponentalariga Reyting[1],...,Reyting[N] xususiy o'zgaruvchilar orqali murojaat qilish mumkin. Boshqacha bu o'zgaruvchilar indeksli o'zgaruvchilar deyiladi. Massiv indeks sifatida butun son qo'llaniladi. Umuman olganda indeks sifatida butun son qiymatini qabul qiladigan ixtiyoriy ifoda ishlatilishi mumkin va uning qiymati massiv elementi nomerini aniqlaydi. Ifoda sifatida o'zgaruvchi ham olinishi mumkinki, o'zgaruvchining qiymati o'zgarishi bilan murojaat qilinayotgan massiv elementini aniqlovchi indeks ham o'zgaradi.

Shunday qilib, programmada bitta indeksli o'zgaruvchi orqali massivning barcha elementlarini belgilash (aniqlash) mumkin bo'ladi. Masalan, Reyting[I] o'zgaruvchisi orqali I o'zgaruvchining qiymatiga bog'liq ravishda Reyting massivining ixtiyoriy elementiga murojaat qilish mavjud.

Haqiqiy turdagi (float, double) qiymatlar to'plami cheksiz bo'lganligi sababli ular indeks sifatida ishlatilmaydi. C++ tilida indeks doimo 0 dan boshlanadi va uning eng katta qiymati massiv e'lonidagi uzunlikdan bittaga kam bo'ladi.

Massiv e'loni quyidagicha bo'ladi:

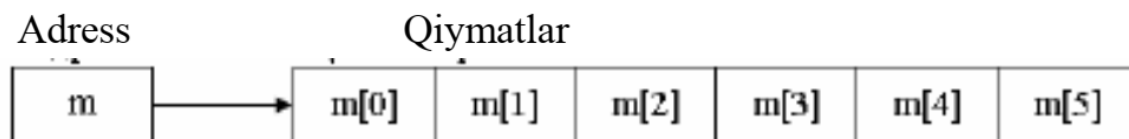
< tur > < nom > [< uzunlik >] = {boshlang'ich qiymatlar}.

Bu yerda <uzunlik> - o'zgarmas ifoda. Misollar:

int m[6]={1,4,-5,2,10,3}; float a[4];

Massiv statik va dinamik bo'lishi mumkin. Statik massivning uzunligi oldindan ma'lum bo'lib, u xotirada ma'lum adresdan boshlanib ketma-ket joylashadi. Dinamik massivni uzunligi programma bajarilish jarayonida aniqlanib,

u dinamik xotiradagi ayni paytda bo'sh bo'lgan adreslarga joylashadi. Masalan, **int m[6]**; ko'rinishida e'lon qilingan bir o'lchamli massiv elementlari xotirada quyidagicha joylashadi.



Bir o'lchamli massivning xotirada joylashuvi

Massivning i - elementiga $m[i]$ yoki $*(m+i)$ - vositali murojaat qilish mumkin. Massiv uzunligini `sizeof(m)` amali orqali aniqladi. Massiv e'lonida uning elementlariga boshlang'ich qiymatlar berish mumkin va uning bir nechta variantlari mavjud.

1) o'lchami ko'rsatilgan massiv elementlarini to'liq inisializasiyalash:

int t[5]={-10,5,15,4,3};

Bunda 5 ta elementdan iborat bo'lgan t nomli butun turdagi bir o'lchamli massiv e'lon qilingan va uning barcha elementlariga boshlang'ich qiymatlar berilgan. Bu e'lon quyidagi e'lon bilan ekvivalent:

int t[5]; t[0]=-10; t[1]=5; t[2]=15; t[3]=4; t[4]=3;

2) o'lchami ko'rsatilgan massiv elementlarini to'liqmas inisializasiyalash:

int t[5]={-10,5,15};

Bu yerda faqat massiv boshidagi uchta elementga boshlang'ich qiymatlar berilgan. Shuni aytib o'tish kerakki, massivning boshidagi yoki o'rtasidagi elementlariga qiymatlar bermasdan, uning oxiridagi elementlarga boshlang'ich qiymat berish mumkin emas. Agarda massiv elementlariga boshlang'ich qiymat berilmasa, unda kelishuv bo'yicha static va extern modifikatori bilan e'lon qilingan massiv uchun elementlarning qiymati 0 soniga teng deb, automatic massivlar elementlarining boshlang'ich qiymatlari noma'lum hisoblanadi.

3) o'lchami ko'rsatilmagan massiv elementlarini to'liq inisializasiyalash:

int t[]={-10,5,15,4,3};

Bu misolda massivni barcha elementlariga qiymatlar berilgan hisoblanadi, massiv uzunligi kompilyator tomonidan boshlang'ich qiymatlar soniga qarab aniqlanadi. Agarda massiv uzunligi berilmasa, boshlang'ich qiymati berilishi shart.

Massivni e'lon qilishga misollar:

char ch[4]={'a','b','c','d'}; //belgilar massivi

int in[6]={10,20,30,40}; // butun sonlar massivi

char str[]="abcd"; //satr uzunligi 5 ga teng, chunki uning oxiriga '\0' belgisi qo'shiladi

char str[]={ 'a','b','c','d' }; // yuqoridagi satrning boshqacha yozilishi

Masala. Berilgan massiv elementlarining har xil qiymatlilari sonini

toping. N – massiv elementlari soni ($1 \leq N \leq 100000$). ($-1000 \leq a[i] \leq 1000$).

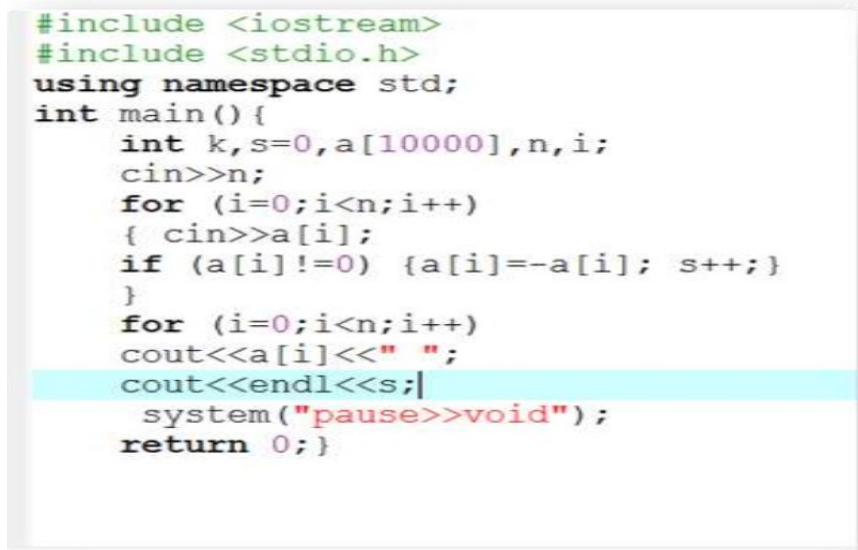
Dastur matni:



10.1-rasm. Massiv elementlari sonini topish.

Berilgan masala uchun algoritm. Berilgan masalada massiv elementlari qiymatlari ishoralarini almashtirish va almashtirishlar sonini aniqlash uchun bizga massiv elementlarining nechitasi 0 ga teng ekanligini aniqlash lozim bo'ladi. Chunki 0 qiymatli sonning ishorasi yo'q.

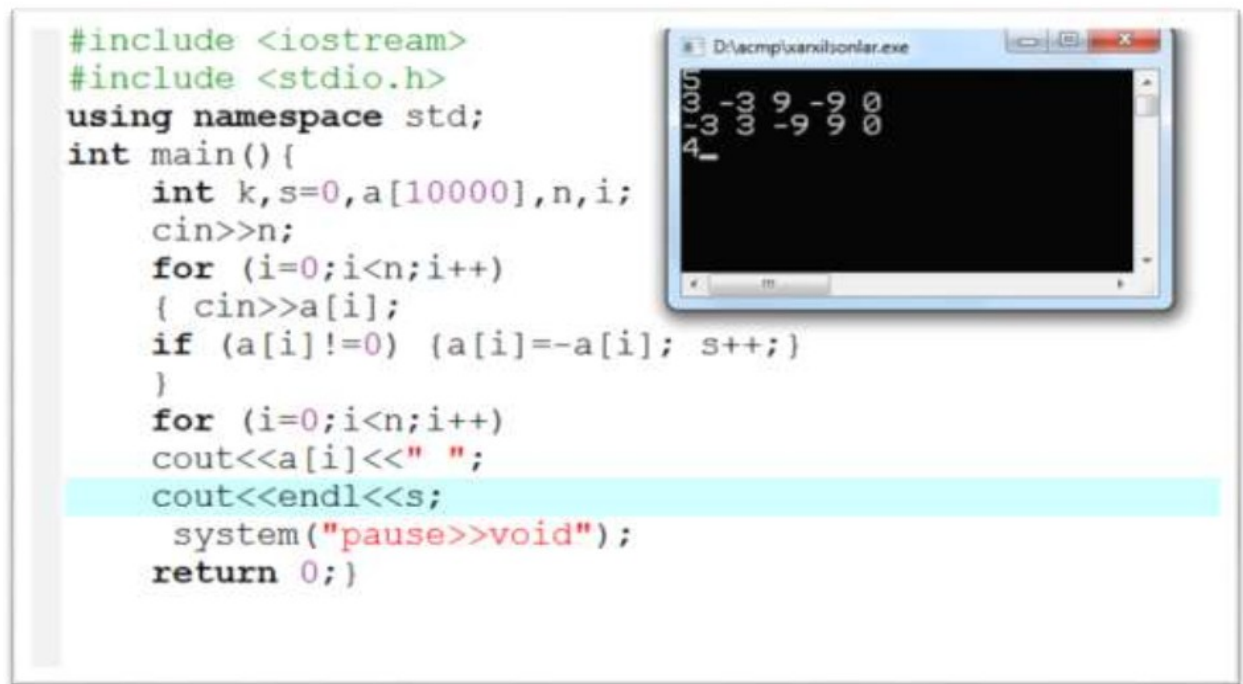
Asosiy dastur matni.



10.2-rasm. Berilgan masala uchun dastur

Berilgan misol ishlanish natijalari tahlili. Berilgan misolda massiv elementlarini kiritish jarayonida ushbu massiv elementlarining qiymati 0 dan farqli bo'lsa uning qiymatini qaramaqarshisiga almashtirilyapti va ushbu almashtirishlar shu joyda sanalayapti.

Natija



10.3-rasm. Berilgan dastur natijasi

Massiv chegarasidan tashqariga chiqish (ya'ni mavjud bo'lmagan elementni o'qish/yozishga urinish) dastur bajarilishida kutilmagan natijalarga olib kelishi mumkin. SHuni ta'kidlab o'tamizki, bu eng ko'p tarqalgan xatolardan biridir.

Agar massiv initsializatsiya qilinganda elementlar chegarasi ko'rsatilgan bo'lsa, ro'yxatdagi elementlar soni bu chegaradan kam bo'lishi mumkin, lekin ortiq bo'lishi mumkin emas.

Misol uchun `int a[5]={2,-2}`. Bu holda `a[0]` va `a[1]` qiymatlari aniqlangan bo'lib, mos holda 2 va -2 ga teng. Agar massiv uzunligiga qaraganda kamroq element berilgan bo'lsa, qolgan elementlar 0 hisoblanadi:

```
int a10[10]={1, 2, 3, 4}; //va 6 ta nol
```

Agar nomlangan massivning tavsifida uning o'lchamlari ko'rsatilmagan bo'lsa, kompilyator tomonidan massiv chegarasi avtomatik aniqlanadi:

```
int a3[]={1, 2, 3};
```

Massivda musbat elementlar soni va summasini hisoblash.

```

#include <iostream>
using namespace std;
int main()
int s=0,k=0;
int x[]={-1,2,5,-4,8,9};
for(int i=0; i<6; i++)
{
if (x[i]<=0) continue;
k++;
s+=x[i];
};

```

```

cout<<k<<endl;
cout<<s;
return 0;
};

```

Massivning eng katta, eng kichik elementi va o'rta qiymatini aniqlash:

```

#include <iostream>
using namespace std;
int main()
{
    int i,j,n;
    int min,max,s;
    float a,b,d,x[100];
    while(1)
    {
        cout<<("\n n=");
        cin>>n;
        if ( n>0 && n<=100 ) break;
        cout<<"\n Hato 0<n<101 bo'lishi kerak";
    }
    cout<<"\n elementlar qiymatlarini kiriting:\n";
    for (i=0;i<n;i++)
    {
        cout<<"x["<<i<<"]="";
        cin>>x[i];
    }
    max=x[0];
    min=x[0];
    for (s=0,i=0;i<n;i++)
    {
        s+=x[i];
        if (max<x[i]) max=x[i];
        if (min>x[i]) min=x[i];
    };
    s/=n;
    cout<<"\n max="<<max;
    cout<<"\n min="<<min;
    cout<<"\n o'rta qiymat="<<s;
    return 0;
}

```

Massivlarni navlarga ajratish. Navlarga ajratish - bu berilgan ko'plab ob'ektlarni biron-bir belgilangan tartibda qaytadan guruhlash jarayoni.

Massivlarning navlarga ajratilishi tez bajarilishiga ko'ra farqlanadi. Navlarga ajratishning $n \cdot n$ ta qiyoslashni talab qilgan oddiy usuli va $n \cdot \log(n)$ ta qiyoslashni talab qilgan tez usuli mavjud. Oddiy usullar navlarga ajratish tamoyillarini tushuntirishda qulay hisoblanadi, chunki sodda va kalta algoritmlarga ega.

Murakkablashtirilgan usullar kamroq sonli operatsiyalarni talab qiladi, biroq operatsiyalarning o'zi murakkabroq, shuning uchun uncha katta bo'lmagan massivlar uchun oddiy usullar ko'proq samara beradi.

Oddiy usullar uchta asosiy kategoriyaga bo'linadi:

- oddiy kiritish usuli bilan navlarga ajratish;
- oddiy ajratish usuli bilan navlarga ajratish;
- oddiy almashtirish usuli bilan navlarga ajratish

Oddiy kiritish usuli bilan navlarga ajratish

Massiv elementlari avvaldan tayyor berilgan va dastlabki ketma-ketliklarga bo'linadi. $i=2$ dan boshlab, har bir qadamda dastlabki ketma-ketlikdan i -nchi element chiqarib olinadi hamda tayyor ketma-ketlikning kerakli o'rniga kiritib qo'yiladi. Keyin i bittaga ko'payadi va h.k.

Kerakli joyni izlash jarayonida, ko'proq o'ngdan bitta pozitsiyadan tanlab olingan elementni uzatish amalga oshiriladi, ya'ni tanlab olingan element, $j:=i-1$ dan boshlab, navlarga ajratib bo'lingan qismning navbatdagi elementi bilan qiyoslanadi. Agar tanlab olingan element $a[i]$ dan katta bo'lsa, uni navlarga ajratish qismiga qo'shadilar, aks holda $a[j]$ bitta pozitsiyaga suriladi, tanlab olingan elementni esa navlarga ajratilgan ketma-ketlikning navbatdagi elementi bilan qiyoslaydilar. To'g'ri keladigan joyni qidirish jarayoni ikkita turlicha shart bilan tugallanadi:

- agar $a[j]>a[i]$ elementi topilgan bo'lsa;
- agar tayyor ketma-ketlikning chap uchiga yetilgan bo'lsa.

int i, j, x;

for(i=1; i<n; i++)

{

x= $a[i]$; // kiritib qo'ishimiz lozim bo'lgan elementni esda saqlab qolamiz

j=i-1;

while(x< $a[j]$ && j>=0) // to'g'ri keladigan joyni qidirish

}

$a[j+1]=a[j]$ / o'nga surilish

j--;

}

$a[j+1]=x$; // elementni kiritish

}

Massivning minimal elementi tanlanadi hamda massivning birinchi elementi bilan joy almashtiriladi. Keyin jarayon qolgan elementlar bilan takrorlanadi va h.k.

int i, min, n_min, j;

for(i=0; i<n-1; i++)

{

min= $a[i]$; n_min=i; // minimal qiymatni qidirish

for(j=i+1; j<n; j++)

if($a[j]<min$) { min= $a[j]$; n_min=j; }

$a[n_min]=a[i]$; // almashtirish

$a[i]=min$;

}

Elementlar juftlari oxirgisidan boshlab qiyoslanadi va o'rin almashinadi. Natijada massivning eng kichik elementi uning eng chapki elementiga aylanadi. Jarayon massivning qolgan elementlari bilan davom ettiriladi.

```
for(int i=1;i<n;i++)
for(int j=n-1;j>=i;j--)
if(a[j]<a[j-1])
{int r=a[j];a[j]=a[j-1];a[j-1]=r;}
}
```

Bir o'lchamli massivlarni funktsiya parametrlari sifatida uzatish. Massivdan funktsiya parametri sifatida foylalanganda, funktsiyaning birinchi elementiga ko'rsatkich uzatiladi, ya'ni massiv hamma vaqt adres bo'yicha uzatiladi. Bunda massivdagi elementlarning miqdori haqidagi axborot yo'qotiladi, shuning uchun massivning o'lchamlari haqidagi ma'lumotni alohida parametr sifatida uzatish kerak.

Funktsiyaga massiv boshlanishi uchun ko'rsatkich uzatilgani tufayli (adres bo'yicha uzatish), funktsiya tanasining operatorlari hisobiga massiv o'zgarishi mumkin.

Funktsiyalarda bir o'lchovli sonli massivlar argument sifatida ishlatilganda ularning chegarasini ko'rsatish shart emas.

Topshiriqlar. Quyidagi variantlarda berilgan massivlarga oid misollar dasturini tuzing.

№	Variantlar	№	Kirish	Chiqish
1	Bir o'lchamli sonli massivning o'rtacha qiymatidan kichik elementlarining o'rtacha qiymati hisoblansin.	1	6 58 22 17 84 50 53	19.50
		2	13 8 37 42 64 16 7 40 12 13 21 24 11 8	12.00
2	Bir o'lchamli sonli massiv [a,b] qismidagi elementlari massivning eng kichik elementiga bo'lib chiqilsin qolganlari o'zgartirishsiz qoldirilsin.	1	4 44 99 55 12 1 3	3.7 8.3 4.6 12.0
		2	2 15 8 1 1	1.9 8.0
3	Bir o'lchamli sonli massiv k – elementidan 1 – elementigacha bo'lgan elementlarining o'rtacha qiymati hisoblansin.	1	3 38 39 41 1 2	38.5
		2	10 6 93 73 62 26 28 65 74 53 59 8 9	63.5
4	Bir o'lchamli sonli massivning minimum elementini massivning oxirgi elementi bilan o'rin almashtirilsin.	1	4 74 0 1 33	74 33 1 0
		2	7 8 37 42 64 16 7 40	8 37 42 64 16 40 7
5	Bir o'lchamli sonli massiv [a,b] qismda yotmaydigan elementlarining o'rtacha qiymati hisoblansin.	1	6 51 49 9 76 56 78 1 4	67.00
		2	17 60 66 34 23 40 68 4 31 36 86 61 59 84 10 28 1 30 11 13	36.93
6	Bir o'lchamli sonli massiv elementlari kvadrlarining yig'indisi hisoblansin.	1	5 24 50 72 96 95	26501
		2	1 43	1849
7	Bir o'lchamli sonli massivning barcha elementlari massiv eng katta elementiga bo'lib chiqilsin.	1	9 37 23 51 47 12 23 90 85 98	0.38 0.23 0.52 0.48 0.12 0.23 0.92 0.87 1.00
		2	3 60 19 27	1.00 0.32 0.45
8	Bir o'lchamli sonli massivni barcha elementlari massiv eng kichik elementiga bo'lib chiqilsin.	1	9 69 48 58 45 57 55 88 89 85	1.53 1.07 1.29 1.00 1.27 1.22 1.96 1.98 1.89
		2	3 4 84 79	1.00 21.00 19.75
9	Bir o'lchamli sonli massiv M dan kattta elementlarini ko'paytmalarini logarifmi hisoblansin.	1	5 26 18 32 68 81 6	18.23
		2	10 49 39 42 12 53 35 94 21 35 12 12	29.96
10	Bir o'lchamli sonli massiv K yoki M ga teng elementlari ko'paytmasi	1	5 7 11 83 18 31 31 3	31

	hisoblansin.	2	7 44 64 23 84 13 6 22 22 6	132
11	Bir o'lchamli sonli massiv M dan katta elementlari yig'indisi hisoblansin.	1	4 12 88 30 87 94	0
		2	9 9 72 18 48 75 32 29 78 25 76	78
12	Bir o'lchamli sonli massivni toq o'rindagi elementlari ko'paytmasini juft o'rindagi elementlari yig'indisiga bo'linsin.	1	2 32 8	4.00
		2	4 38 34 13 48	6.02
13	Bir o'lchamli sonli massiv manfiy elementlarini o'rtacha qiymati hisoblansin.	1	9 93 64 -90 74 62 -83 58 15 -37	-70.00
		2	10 63 89 -6 48 77 -19 16 73 -72 34	-32.33
14	Bir o'lchamli sonli massivni 2 ga va 5 ga bo'linadigan elementlarini ko'paytmasining sinusi topilsin.	1	6 44 34 42 83 43 64	0.02
		2	15 62 54 24 95 67 62 25 17 77 50 38 12 90 59 7	0.64
15	Bir o'lchamli sonli massivni M dan kichik elementlarini kvadratlari yig'indisi hisoblansin.	1	1 38 9	0
		2	14 85 15 57 68 18 67 7 45 69 21 1 5 98 34 92	28594
16	Bir o'lchamli sonli massiv elementlari massivning eng katta elementiga bo'lib chiqilsin.	1	5 91 51 75 85 29	1.00 0.56 0.82 0.93 0.32
		2	8 30 31 30 94 61 41 74 78	0.32 0.33 0.32 1.00 0.65 0.44 0.79 0.83
17	Bir o'lchamli sonli massiv toq o'rindagi elementlarini yig'indisi hisoblansin.	1	10 93 64 63 8 50 24 32 80 3 76	241
		2	2 85 88	85
18	Bir o'lchamli sonli massivning toq qiymatli elementlarini o'rtacha qiymati hisoblansin.	1	5 76 12 51 50 98	51.00
		2	13 23 87 77 4 14 57 91 16 80 7 45 78 46	55.29
19	Elementlari soni N ($1 \leq N \leq 10000$) ta bo'lgan butun sonli massiv berilgan. Ushbu massivning elementlarining qarama-qarshisini hamda necha marta ishora almashish amalga oshganligini aniqlovchi dastur tuzing. Massiv elementlarini qiymatlari $[-1000,1000]$ oraliqda joylashgan.1	1	5 3 -3 9 -9 0	-3 3 -9 9 0 4
		2	10 0 1 2 3 4 5 0 7 8 -9	0 -1 -2 -3 -4 -5 0 -7 -8 9 8
20	Bir o'lchamli sonli massiv elementlarini qiymati $[x,y]$ oraliqda yotmaydigan elementlari soni aniqlansin.	1	10 14 51 -83 42 85 -77 91 70 -98 54 50 99	5
		2	3 1 77 -58 20 97	2

11-laboratoriya ishi. C++ algoritmik tilida funksiyalarning grafigini qurish.

Ishdan maqsad: C++ algoritmik tilning bir va ikki o'lchamli funksiyalarning grafigini qurishni o'rganish.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. C++da grafik rejimida ishlash.
3. O'zgarmlar, o'zgaruvchilar va turli qism dasturlar.
4. Graphics.h fayli bilan ishlash
5. Grafik rejimida ishlash.
6. Tajriba ishinining hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Lokal o'zgaruvchilar. Funksiyaga qiymatlar uzatish bilan birga uning tanasida o'zgaruvchilarni e'lon qilish ham mumkin. Bu lokal o'zgaruvchilar orqali amalga oshiriladi. Qachonki dasturni bajarilishi funktsiyadan asosiy qismga qaytsa, bu funktsiyadagi lokal o'zgaruvchilar xotiradan o'chiriladi.

Lokal o'zgaruvchilar xuddi boshqa o'zgaruvchilar kabi aniqlanadi. Funksiyaga beriladigan parametrlarni ham lokal o'zgaruvchilar deb atash mumkin va ularni funktsiya tanasida aniqlangan o'zgaruvchilar kabi ishlatish mumkin.

Funktsiya lokal o'zgaruvchilari va parametrlarining qo'llanilishi:

```
#include <iostream>
using namespace std;
float Almashtirish(float);
int main()
{
    float TempFer;
    float TempCel;
    cout << "Feringait bo'yicha temperaturani";
    cout << "kiriting:";
    cin >> TempFer;
    TempCel = Almashtirish(TempFer);
    cout << "\n Bu temperatura selziy shkalasi";
    cout << "bo'yicha: ";
    cout << TempCel << endl;
    return 0 ;
}
float Almashtirish(float TempFer)
{
    float TempCel;
    TempCel=((TempFer-32)*5)/9;
    return TempCel;
};
Natija:
```

Faringait bo'yicha temperaturani kiriting: 50

Bu temperatura selziy shkalasi bo'yicha: 10

Global o'zgaruvchilar. main() funktsiyasida aniqlangan o'zgaruvchilar dasturdagi barcha funktsiyalar uchun murojaat qilishga imkonli va ko'rinish sohasiga ega hisoblanadi. Bunday o'zgaruvchilar dasturdagi funktsiyalar uchun global o'zgaruvchilar deyiladi.

Global o'zgaruvchi nomi bilan funktsiya ichida nomlari ustma-ust tushadigan lokal o'zgaruvchilar faqatgina joriy funktsiyaning ichidagina global o'zgaruvchining qiymatini o'zgartiradi. Lekin global o'zgaruvchi funktsiya o'z ishini tugatgach u chaqirilishidan oldingi qiymatini saqlab qoladi, ya'ni funktsiya tanasida e'lon qilingan lokal o'zgaruvchi funktsiyaning ichida global o'zgaruvchini yashiradi xolos. Bunda lokal o'zgaruvchi alohida hosil qilinadi va funktsiya ishlash vaqtida global va lokal o'zgaruvchilarning nomlari bir xil bo'lsa faqatgina lokal o'zgaruvchi ustida amallar bajariladi. Global o'zgaruvchi esa funktsiyaning bajarilishi davomida oldingi qiymatini saqlab turadi.

Global va lokal o'zgaruvchilarning qo'llanishi

```
#include <iostream>
```

```
using namespace std;
```

```
void MeningFunktsiyam() ; // prototip
```

```
int x = 5, y = 7; // global o'zgaruvchilar
```

```
int main()
```

```
{
```

```
cout << "main()dagi x ning qiymati:"<<x<<"\n";
```

```
cout<<"main()dagi y ning qiymati y:"<<y<<"\n";
```

```
MeningFunktsiyam();
```

```
cout << "MeningFunktsiyam()funktsiyasi"<<"ishini tugatdi! \n";
```

```
cout<<"main()dagi x ning qiymati:"<<x<<"\n";
```

```
cout<<"main()dagi y ning qiymati:"<<y<<"\n";
```

```
return 0;
```

```
}
```

```
void MeningFunktsiyam()
```

```
{
```

```
int y = 10;
```

```
cout<<"MeningFunktsiyam()dagi"<<"x:" << x <<"\n";
```

```
cout<<"MeningFunktsiyam()dagi"<<"y:"<<y<<"\n";
```

```
}
```

Hatija:

main()dagi x ning qiymati: 5

main()dagi y ning qiymati: 7

MeningFunktsiyam()dagi x: 5

MeningFunktsiyam()dagi y: 10

MeningFunktsiyam()funktsiyasiishini tugatdi!

main()dagi x ning qiymati: 5

main()dagi y ning qiymati: 7

Lokal o'zgaruvchilar haqida batafsilroq ma'lumot. Funktsiyani ichida aniqlangan o'zgaruvchilar lokal ko'rinish sohasiga ega deyiladi. Yuqorida aytib o'tilganidek, bu o'zgaruvchilarni faqatgina funktsiyaning ichidagina qo'llash mumkinligini anglatadi. S++da o'zgaruvchilarni nafaqat dasturning boshida balki ixtiyoriy joyda aniqlash mumkin. Agarda o'zgaruvchi funktsiya tanasidagi biror bir blok ichida aniqlangan bo'lsa, bu o'zgaruvchi faqatgina shu blok ichidagina ta'sirga ega bo'lib butun funktsiyaning ichida ko'rinish sohasiga ega bo'lmaydi.

Funktsiyada ishlatiladigan operatorlar. Funktsiyada ishlatiladigan operatorlar soni va turiga hech qanday chegara yo'q. Funktsiya ichida istalgancha boshqa funktsiyalarni chaqirish mumkin bo'lsada, lekin funktsiyalarni aniqlash mumkin emas.

C++ tilida funktsiyaning hajmiga hech qanday talab qo'yilmasa ham uni kichikroq tarzda tuzgan ma'quldir. Har bir funktsiya tushunish oson bo'ladigan bitta masalani bajarishi lozim. Agarda funktsiya hajmi kattalashayotganligini sezsangiz, bu funktsiyani bir nechta kichikroq funktsiyalarga ajratish haqida o'ylashingiz kerak.

Funktsiya argumentlari. Funktsiyaning argumentlari turli tipda bo'lishi mumkin. SHuningdek, argument sifatida C++ tilidagi biror bir qiymat qaytaradigan o'zgaruvchilar, matematik va mantiqiy ifodalar va boshqa ixtiyoriy funktsiyalarni berish mumkin.

Misol sifatida bizda biror bir qiymat qaytaruvchi double(), triple(), square() va cube() funktsiyalari berilgan bo'lsin. Biz quyidagi ifodani yozishimiz mumkin:

```
Answer=(double(triple(square(my Value))))
```

Bu ifodada myValue o'zgaruvchisini qabul qilinadi va u cube() funktsiyasiga argument sifatida uzatiladi.

cube() funktsiyasi qaytargan qiymat esa square() funktsiyasiga argument sifatida uzatiladi. square() funktsiyasi qiymat qaytargandan keyin, buning qiymati o'z navbatida triple() funktsiyasiga argument sifatida beriladi. triple() funktsiyasining qiymati esa double() funktsiyasiga argument qilib beriladi va u Answer o'zgaruvchisiga o'zlashtiriladi.

Parametrlar bu lokal o'zgaruvchilardir. Funktsiyaga uzatilgan har bir argument, bu funktsiyaga nisbatan lokal munosabatda bo'ladi. Funktsiyani bajarilishi jarayonida argumentlar ustida bajarilgan o'zgartirishlar funktsiyaga qiymat sifatida berilgan o'zgaruvchilarga ta'sir qilmaydi.

Funktsiyaning qaytaradigan qiymatlari. Funktsiya yo biror bir real qiymatni, yo kompilyatorga hech qanday qiymat qaytarilmasligi haqida xabar beruvchi void tipidagi qiymatni qaytaradi.

Funktsiyani qiymat qaytarishi uchun return kalitli so'zidan foydalaniladi. Bunda oldin return kalitli so'zi, keyin esa qaytariladigan qiymat yoziladi. Qiymat sifatida esa o'zgaruvchilar kabi butun bir ifodalarni ham berish mumkin. Masalan:

```
return 5 ;
```

```
return (x > 5) ;
```

```
return (MyFunction()) ;
```

MyFunction() funktsiyasi biror bir qiymat qaytarishidan kelib chiqsak,

yuqoridagi barcha ifodalar to'g'ri keltirilgan. `return(x>5)` ifodasi esa `x` 5 dan katta bo'lsa `true`, kichik yoki teng bo'lsa `false` mantiqiy qiymatini qaytaradi.

Agarda funktsiyada `return` kalit so'zi uchrasa undan keyingi ifoda bajariladi va uning natijasi funktsiya chaqirilgan joyga uzatiladi. `return` operatori bajarilgandan keyin dastur funktsiya chaqirilgan satrdan keyingi ifodaga o'tadi. `return` kalitli so'zidan keyingi funktsiya tanasidagi operatorlar bajarilmaydi.

Grafik biblioteka. Turbo C va Borland C++ kompilyatorlarida grafik biblioteka bilan bog'lanish uchun `graphic.h` – sarlavxali fayl qo'llaniladi.

Bu bibliotekaga kiruvchi ba'zi grafik funktsiyalar:

`void initgraph(int* graphdriver, int* graphmode, char* pathdriver)`- grafik rejimga o'tkazish

`void closegraph(void)` -grafik rejimdan matnli rejimga o'tkazish.

`void putpixel(int x, int y, int color)` - Ekranda `color` rangli(`x,y`) kordinatali nuqtani tasvirlaydi.

`void line (int x1, int y1, int x2, int y2)` - Ekranda chiziq chizadi chizadi.

`void rectangle (int left, int top, int right, int bottom)` - Ekranda to'rtburchak chizadi.

`void circle (int x, int y; int radius)` - Ekranda aylana chizadi.

`void ellipse (int x, int y, int stangle, int endangle, int xradius, int yradius)` - Ekranda ellips chizadi.

`void outtextxy (int x, int y, char* textstring)` – Textni berilgan pozitsiyada chiqaradi.

`void outtext (char* textstring)` – Textni joriy pozitsiyada chiqaradi.

`int getbcolor(void)` - Fon rangini qaytaradi

`int getcolor(void)` - Tasvir rangini qaytaradi.

`void getimage (int left, int top, int right, int bottom, void* bitmap)` - ekran oynasini xotirada saqlash;

`putimage (int left, int top, void* bitmap, int op)`- xotirada saqlangan tasvirni ekranga joylash;

Grafik sinfga misol. Misol tariqasida nuqta tushunchasini aniqlovchi `point` sinfini yaratib `point.cpp` fayliga yozib qo'yamiz:

```
#ifndef POINTH
#define POINTH 1
class point
{
protected:
int x,y;
public:
point ( int xi=0, int yi=0);
int& givex(void);
int& givey (void);
void show(void);
void move(int xn=0, int yn=0);
```

```
private:
void hide();
}
#endif
```

Kelgusida point sinfini boshqa sinflarga qo'shish mumkin bo'lgani uchun shartli protsessor direktivasi `#ifndef POINTH` ishlatilgan. Protsessorli identifikator `POINTH` `#define POINT 1` direktivasi orqali kiritilgan.

SHuning uchun `#include "point.h"` direktivasi bir necha marta ishlatilganda ham `POINT` sinfi ta'rifi teksti faqat bir marta kompilyatsiya kilinaetgan faylda paydo bo'ladi.

`POINT` sinfi komponenta funktsiyalarini quyidagicha ta'riflaymiz:

```
#ifndef POINTCPP
#define POINTCPP 1
#include <graphics.h>
#include "point.h"
point::point(int xi=0, int yi=0)
{
x=xi; y=yi;
}
int &point::givex(void)
{
return x;
}
int &point::givey(void)
{
return y;
}
void point::show(void)
{
putpixel(x,y, getcolor());
}
void point::hide (void)
{
putpixel(x,y,getbcolor());
}
void point::move(int xn=0, int yn=0)
{
hide( );
x=xn; y=yn;
show( );
}
#endif
```

Kiritilgan point sinfi va komponenta funktsiyalari bilan ishlovchi dasturni keltiramiz:

```
#include <graphics.h>
```

```

#include <conio.h>
#include "point.cpp"
int main()
{
point A(200,50);
point B;
point D(500,200);
int dr=DETECT, mod;
initgraph(&dr, &mod, "c:\\borlandc\\bgi");
A.show();getch();
B.show();getch();
D.show();getch();
A.move();getch();
B.move(50,60);getch();
closegraph();
return 0;
}

```

Qo'shimcha yuklash. Misol tarikasida '+' amalini point sinfi bilan ta'riflanuvchi displeydagi nuqtalarga qo'llaymiz. Soddashtirish uchun point sinfida eng kerakli komponentalarni qoldiramiz.

```

#include <graphics.h>
#include <conio.h>
class point
{
protected:
int x,y;
public:
point (int xi=0, int yi=0)
{
x=xi; y=yi;
}
void show (void)
{
putpixel (x,y,get color ( ) );
};
point operator+ (point p);
};
point point : : operator + (point &p)
{
point d;
d.x=this.x+p.x;
d.y=this.y+p.y;
return d;
}

```

```

int main()
{
int dr=DETECT, mod;
point A(200,50);
point B;
point D(50,120);
INITGRAPH (&dr, &mod, "C\\borlandc \\ bgi");
A.show ( );
getch ( );
B.show ( ); getch ( );
D.show ( ); getch ( );
B=A+D
;
B.show ( ); getch ( );
B=A.operator + (B);
B.show ( ); getch ( );
closegraph ( );
return 0;
}

```

Dastur bajarilishi natijasida displey ekraniga ketma-ket quyidagi nuqtalar qo'yiladi: A(200,50); B(0,0); D(50,120); B(250,70), B(450,220)

Sinov savollari.

1. Siz jadvalning ma'lum bir turini qanday tanlaysiz?
2. Xujjatda rasm va shakl qo'yish uchun nimadan foydalaniladi?
3. Qanday qilib formulaga xujjat va simvollarni qo' yasiz?
5. Elektron jadvallarni qanday raqamlashtirish mumkin va ularning maksimal soni?
7. Qnday qilib mavjud katakka formula berish mumkin?
8. Formula vazifalarida qanday asosiy funksiyalardan foydalaniladi?

12-laboratoriya ishi.Kimyo injineri uchun MathCad muxiti.

Ishdan maqsad: MathCad dasturi bilan tanishish.

Topshiriq:

- 1.Nazariy qism bilan tanishish.
- 2.MathCAD dasturining asosiy imkoniyatlari bilan tanishish.
- 3.MathCAD dasturi ish sohasi bilan tanishish.
- 4.Tajriba ishining hisobotini tayyorlang.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Mathcad professor-o'qituvchilar, stajyorlar, tadqiqotchilar, talabalar, texnik muxandislar, fiziklar, qolaversa barcha kasb egalari uchun qisoblash ishlarini bajaruvchi dasturiy ta'minot hisoblanadi. Bu dastur bilan turli kasb egalari o'z

sohasi bo'yicha masalalarni hal etishi va kerakli grafiklarni, diagrammalarni olishlari mumkin. Mathcad dasturini boshqacha qilib aytganda dasturlash tili deyish mumkin.

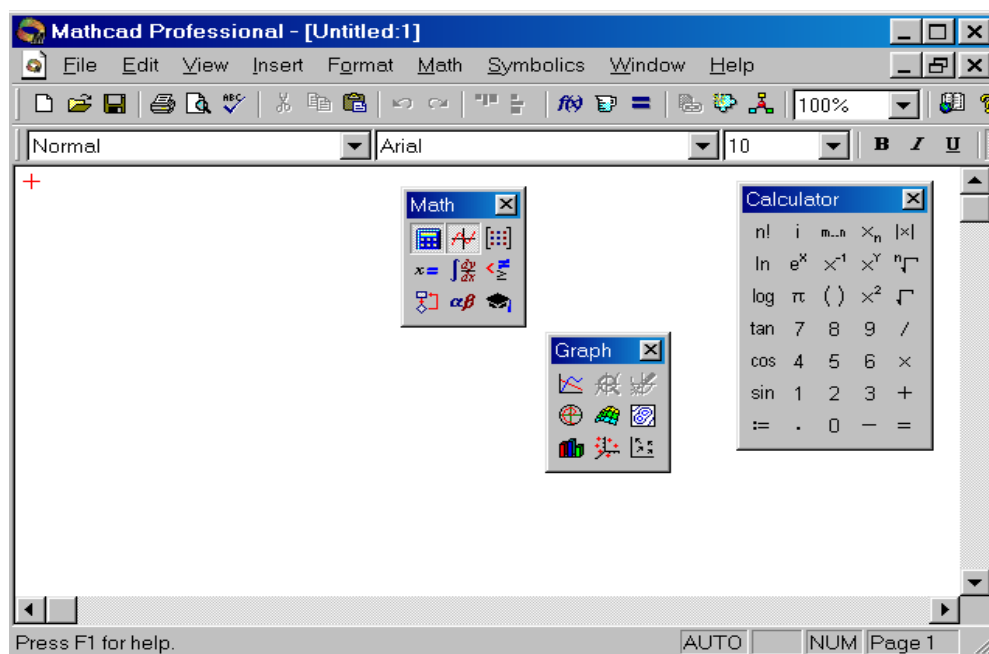
Mathcadda matematik tenglamalarni qog'ozga qanday yozilsa, ekranga ham shunday yoziladi. Bir vaqtning o'zida natijalarni ham olish mumkin. Foydalanuvchi bema'lol tenglamalariga izoh ham yozishi, 2 va 3 o'lchovli grafiklarni ham chizishi mumkin. Mathcad dasturining afzal tomonlaridan biri murakkab hisoblashlarni bajara olishi imkoniga ega. Foydalanuvchi masalasini, maqolasini, qolaversa barcha ilmiy ishlarini tayyorlashda ularni formatlash va chiroyli ko'rinishga keltirishda Mathcad ko'p imkoniyat yaratib beradi.

Mathcad yuzdan ortiq o'zgaruvchili va konstantali chiziqli va chiziqsiz tenglamalar tizimi, matritsa va vektorlar ustida amallar, algebraik hisoblashlar, Laplas, Fure integrallari, massivlar, oddiy differentsial tenglamalar, chegaraviy shartlar, xususiy hosilali differentsial tenglamalar, polinomlarni tushuna oladi, ular ustida hisoblash ishlarini bajaradi.

Mathcad ilmiy ishlarning natijalarni grafiklar bilan vizual qarashga imkon beradi. Foydalanuvchi funksiyalarini osongina 2 va 3 o'lchovli grafiklarda turli ranglar ko'rinishida, tekislikda tasvirlash imkoniga ega bo'ladi. Mathcad Help darchasidan foydalanishda ancha qulayliklar yaratilgan, bu ma'lumotnomadan kerakli ma'lumotlarni osongina qidirib topish mumkin.

Mathcad dasturini ishga tushirish:

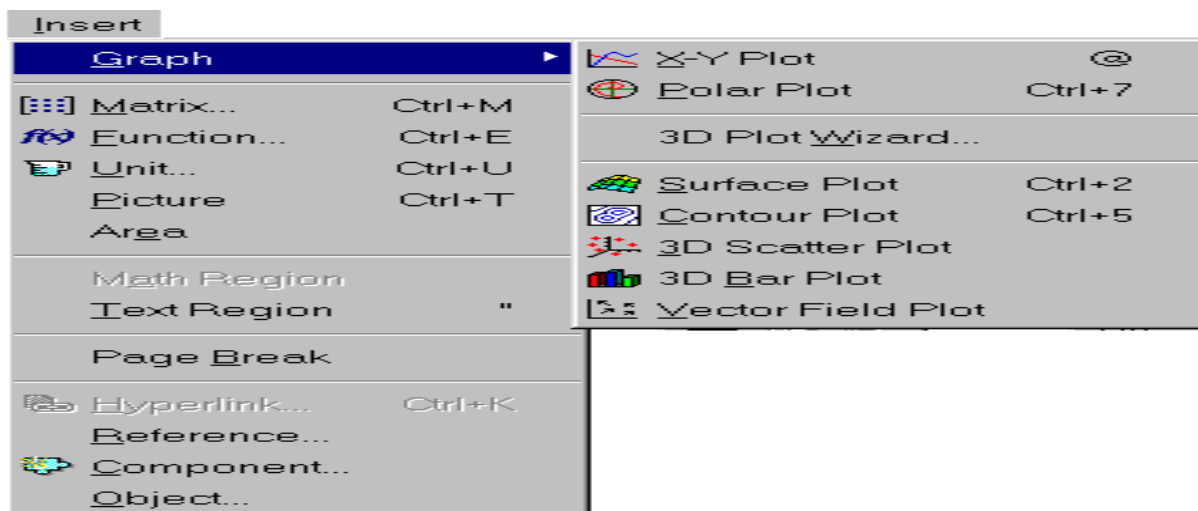
Mathcad dasturini ishga tushirish uchun Pusk > Programmo`>Mathsoft>Mathcad 11 Professional buyrug'i berilsa ekranda 12.1-rasmda ko'rsatilgan darcha hosil bo'ladi:



12.1-rasm. Mathcad 11 Professional dasturining asosiy darchasi va uning turli xil panellari

Asboblar paneli

Ekrandagi darchada dasturning turli xil panellar ko'rsatilgan. Bu panellarning vazifasidan kelib chiqqan holda ularning har biri nomlanadi. Misol uchun Calculation paneli yordamida hisoblash ishlarini bajariladi. Graph paneli yordamida turli xil grafiklar chiziladi. Bu Graph panelini Insert tavsiyanomasidagi Graph bo'limidanham ishga tushirish mumkin va bu tavsiyanoma 12.2-rasmda ko'rsatilgan.



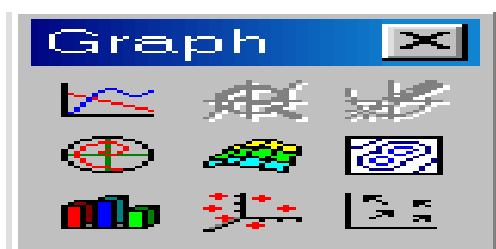
12.2 -rasm. Mathcad dasturdagi Insert tavsiyanomasining ko'rinishi

Bu panellarda knopkalar mavjud bo'lib, bu knopkalar esa grek harflar, hisoblashlar, grafiklar, operatorlar va simvollar panellarini ochishga yordam beradi. Ular ustidaqisqacha qilib quyidagilarni keltirib o'tish mumkin.

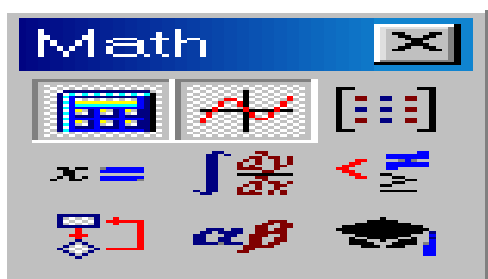
Graph panel

Bu paneldan quyidagi turdagi grafiklar chiziladi:

X-Y Plot	@	- x va y koordinatalar (2 o'lchovli) bo'yicha
Polar Plot	Ctrl+7	- qutb bo'yicha
Surface Plot	Ctrl+2	- yuza bo'yicha
Contour Plot	Ctrl+5	- kontur bo'yicha
3D Scatter Plot		- 3 o'lchovli grafik bo'yicha
Vector Field Plot		- vektor bo'yicha

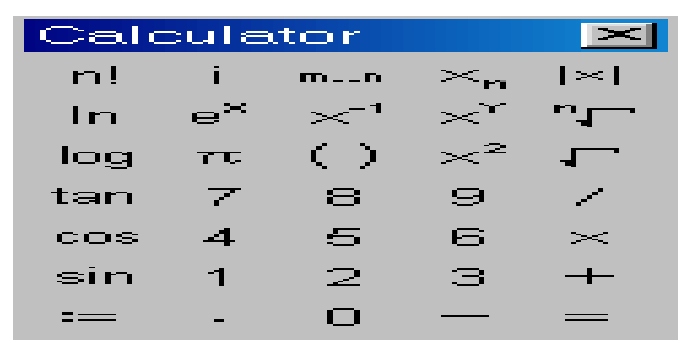


Matrix paneli

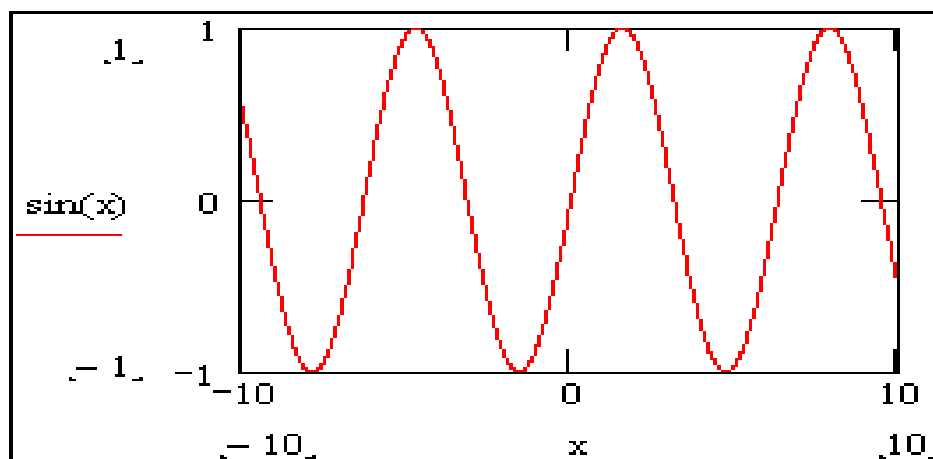


Bu panelidagi knopkalaridan matritsa, x ning har xil ko'rinishlari, modullarni chizish mumkin.
Greek panelil yordamida grek harflarini yozish mumkin.

Calcuiation paneli



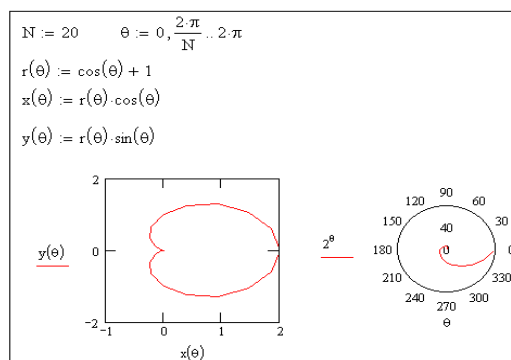
Bu panel bilan esa hisoblash ishlarini bajarish mumkin, shu bilan birga yana math, modifier, boolean, evalution va programming panellari mavjud.



12.3-rasm. $\sin(x)$ funksiyaning grafigi.

Grafikning o'lchamini o'zgartirish uchun sichqonchanning ko'rsatkichini grafikni ustiga olib borib, chap klavishini bosiladi. Grafikning hamma tomonlarida ramka va uning chetlarida qora kichkinagina kvadratlar paydo bo'ladi. Shu kvadratlari ustiga sichqonning ko'rsatkichini olib borish kerak, birdaniga ko'rsatkichning ko'rinishi o'zgaradi. So'ngra esa sichqonchanning chap knopkasi bosiladi va knopkani quyib yubormay grafikning o'lchamini o'zgartirish mumkin.

Yozgan va chizilgan rafiklarni ixtiyoriy joyga olib borish uchun sichqonchani ular ustiga bosilsa ko'rsatkich odamning qo'l shakliga o'zgaradi. Yana chap knopkani bosib kerakli joyga olib borib qo'yiladi. Bir hujjatda bir necha funksiyalarning grafiklarini chizish ham mumkin. Ekranga funksiyalarni oldindan yozib olish kerak. So'ngra esa Insert tavsiyanomasidagi Graph bo'limi tanlanadi va grafiklardan x va u koordinatalar (2 o'lchovli) bo'yicha grafik ishga tushiriladi va ekranda koordinatali grafik hosil bo'ladi. Ramkani ichidagi kursor turgan joylarga x, u funksiyalar yoziladi va Enter klavishi bosiladi. Birinchi grafik hosil bo'ladi. Yana Insert tavsiyanomasidagi Graph bo'limiga o'tiladi va grafiklardan qutb bo'yicha grafik ekranga chiqariladi va funktsiya oldingidek yozilsa quyidagi rasmdagi grafik hosil qilinadi:



13-laboratoriya ishi. Mathcadda va elektron jadvallarda tenglamalar yechish.

Ishdan maqsad: MathCad dasturida ishlash.

Topshiriq:

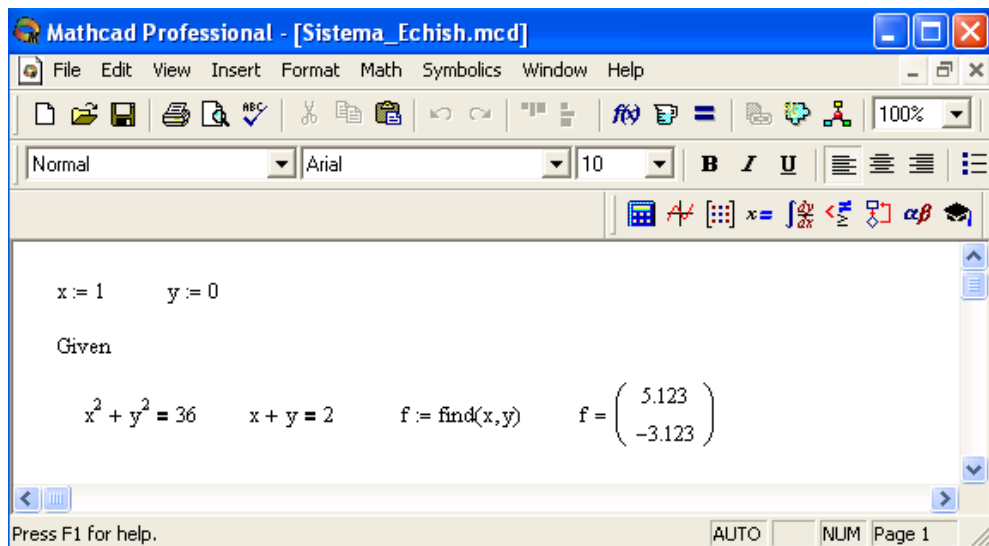
1. Nazariy qism bilan tanishish.
2. MathCAD dasturining asosiy imkoniyatlari bilan tanishish.
3. Mathcadda va elektron jadvallarda tenglamalar yechish.
4. Tajriba ishining hisobotini tayyorlang.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Mathcadda tenglamalar tizimini yechish Given...Find hisoblash bloki yordamida amalga oshiriladi. Tenglamalar tizimini yechish uchun iteratsiya usuli qo'llaniladi va yechishdan oldin boshlang'ich yaqinlashish barcha noma'lumlar uchun beriladi.

Tenglamalar tizimini yechish uchun quyidagi protsedurani bajarish kerak:

1. Tizimga kiruvchi barcha noma'lumlar uchun boshlang'ich yaqinlashishlarni berish.
2. Given kalit so'zi kiritiladi.

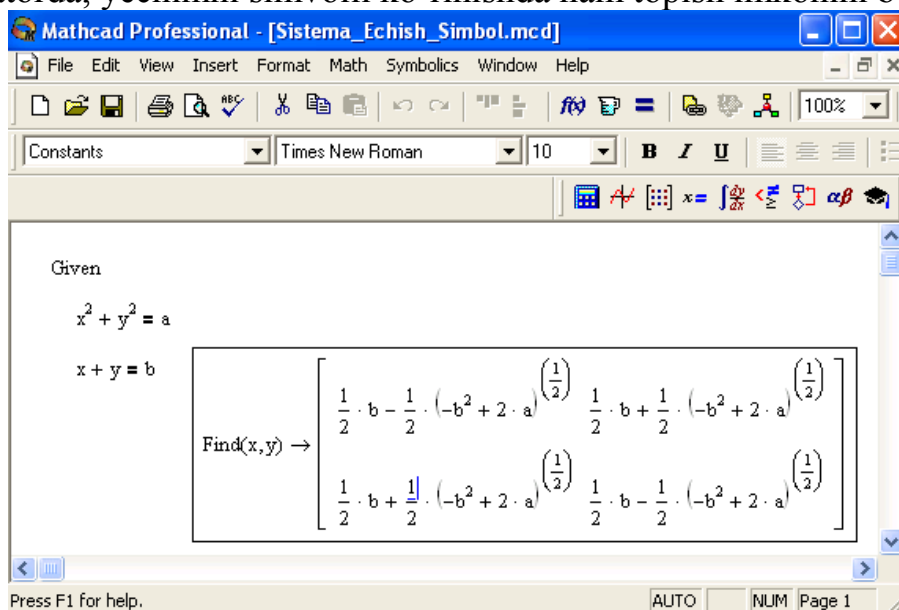


13.1-rasm. Chiziqsiz tenglamalar tizimini echish.

3. Tizimga kiruvchi tenglama va tengsizlik kiritiladi. Tenglik belgisi qalin bo'lishi kerak, buning uchun $\text{Ctrl}+=$ klavishlarini birgalikda bosish kerak bo'ladi yoki Boolean (Bul operatorlari) panelidan foydalanish mumkin.

4. Find funksiyasi tarkibiga kiruvchi o'zgaruvchi yoki ifodani kiritish. Funksiyaga murojaat quyidagicha bajariladi: $\text{Find}(x,y,z)$. Bu erda x,y,z – noma'lumlar. Noma'lumlar soni tenglamalar soniga teng bo'lishi kerak.

Find funksiyasi funksiya Root ga o'xshab tenglamalar tizimini sonli yechish bilan bir qatorda, yechimni simvolli ko'rinishda ham topish imkonini beradi.

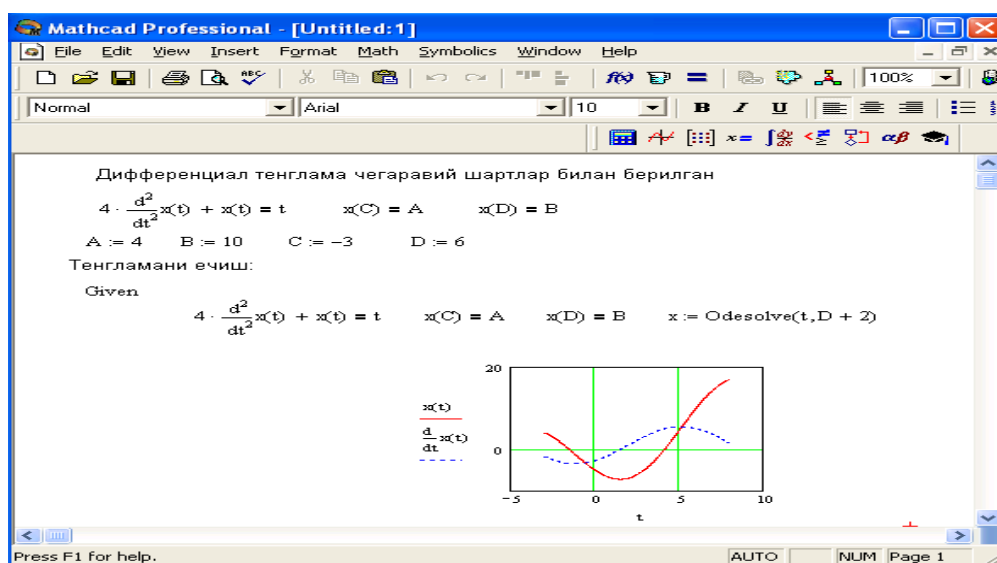


13.2-rasm. Chiziqsiz tenglamalar tizimini simvoli yechimini topish.

Differensial tenglamalarni yechish

Differensial tenglamalarni yechish ancha murakkab masala. Shu sabab Mathcadda barcha differensial tenglamalarni ma'lum chegaralanishlarsiz to'g'ri yechish imkoniyati mavjud emas. Mathcadda differensiallar tenglama va tizimlarini yechishning bir necha usullari mavjud. Bu usullardan biri Odesolve funksiyasi yordamida yechish bo'lib, bu usul boshqa usullarga nisbatan eng

soddasidir. Bu funksiya Mathcad 2000 da birinchi bor yaratildi va u birinchi bor differensial tenglamani yechdi. Mathcad 2001da bu funktsiya yanada kengaytirildi. Odesolve funksiyasida differensial tenglamalar tizimini ham echish mumkin. Mathcad differensial tenglamalarni yechish uchun yana ko'gina qurilgan funksiyalarga ega. Odesolve funksiyasidan tashqari ularning barchasida, berilgan tenglama formasini yozishda ancha murakkablik mavjud. Odesolve funksiyasi tenglamani kiritish blokida oddiy differensial tenglamani o'z shaklida, xuddi qog'ozga yozgandek yozishga imkon yaratadi. Odesolve funksiyasi yordamida differensial tenglamalarni boshlang'ich shart va chegaraviy shartlar bilan ham yechish mumkin.



13.3.-rasm. Differensial tenglamalarni yechish.

Berilgan tenglamani yozishda xuddi differentsiallashtirish operatorini ishlatgan holda ham yoki shtrixlar bilan ham yozish mumkin. Boshlang'ich shartni yozishda esa faqat shtrix bilan yozish kerak va uni kiritish uchun Ctrl+F7 klavishilarni baravar bosish kerak.

Odesolve funksiyasiga murojaat uch qismdan iborat hisoblash bloki yozuvini talab qiladi:

Given kalit so'zi;

- Differensial tenglama va boshlang'ich yoki chegaraviy shart yoki differensial tenglamalar tizimi va unga shartlar;
- Odesolve(x,xk,n) funksiya, bu erda x – o'zgaruvchi nomi, xk – integrallashtirish chegarasi oxiri (integrallashtirishning boshlang'ich chegarasi boshlang'ich shartda beriladi); n – ichki ikkinchi darajali parametr bo'lib, u integrallashtirish qadamlar sonini aniqlaydi (bu parametr berilmasa ham bo'ladi. Unda qadamni Mathcad avtomatik ravishda tanlaydi).

Differensial tenglamalar tizimini echish uchun Odesolve funksiyasi ko'rinishi quyidagicha: Odesolve(<noma'lumlar vektori>, x, xk, n).

Topshiriq. Aniq integrallarni hisoblang:

1. $\int_0^3 (-1x + 1x^2) dx.$ j : 24.
2. $\int_0^4 \left(2x^2 - x - \frac{1}{2\sqrt{3}} \right) dx.$
3. $\int_0^1 (-\sqrt{x^2}) dx.$ j : 0,8.
4. $\int_1^2 \frac{x+}{\sqrt{x}} dx.$ j : $6\frac{2}{3}.$
5. $\int_0^{\frac{\pi}{4}} \left(\cos 2x + \frac{2}{\cos^2 x} \right) dx.$ j : 2,5.
6. $\int_0^3 \frac{dx}{\sqrt{9-x^2}}.$ j : $\frac{\pi}{4}.$
7. $\int_0^{\sqrt{3}} \frac{dx}{9-x^2}.$ j : $\frac{\pi}{18}.$
8. $\int_0^2 (x-3) dx.$ j :
9. $\int_0^{\frac{3\pi}{2}} \cos \frac{x}{3} dx.$ j : 3.
10. $\int_{\frac{\pi}{6}}^{\frac{\pi}{4}} \sin 2x dx$ j : 0,25.
11. $\int_{\frac{\pi}{4}}^{\frac{\pi}{2}} \sin \left(2x - \frac{\pi}{2} \right) dx.$ j : 0,5.
12. $\int_{\frac{\pi}{12}}^{\frac{\pi}{9}} \frac{3dx}{2\cos^2 3x}.$ j : $\frac{1}{2} (\sqrt{3} - 1)$
13. $\int_0^{\frac{\pi}{12}} \frac{dx}{\sin^2 \left(x + \frac{\pi}{6} \right)}$ j : $\sqrt{3} - 1.$
14. $\int_0^{\sqrt{2}} \frac{2dx}{\sqrt{4-x^2}}.$ j : $\frac{\pi}{4}.$
15. $\int_0^1 \frac{dx}{4+x^2}.$ j : $\frac{\pi}{8}.$
16. $\int_{\frac{3}{4}}^{\frac{3\sqrt{3}}{4}} \frac{4}{9+16x^2} dx.$ j : $\frac{\pi}{36}.$
17. $\int_0^2 (x^2 - 3) dx.$ j : $10\frac{1}{8}.$
18. $\int_1^{\sqrt{3}} \frac{128xdx}{(x^2+3)^3}.$ j : $\frac{15}{16}.$
19. $\int_0^{\sqrt{3}} 6\sqrt{x^4+6x^3} dx.$ j : 61.
20. $\int_0^{\sqrt{3}} \frac{xdx}{2\sqrt{1+x^2}}.$ j : 0,5.
21. $\int_2^3 \frac{15xdx}{(x^2-3)^3}.$ j : 0,4.
22. $\int_0^{\pi/3} \frac{\sin x dx}{3-\cos x}.$ j : $\ln \frac{5}{4}.$
23. $\int_0^{\pi/3} e^{\cos x} \sin x dx.$ j : $e - \sqrt{e}.$
24. $\int_0^{\pi/2} \frac{\cos x dx}{2+\ln x}.$ j : $\ln \frac{3}{2}.$
25. $\int_0^1 \frac{e^x dx}{e^x + 1}.$ j : $\ln \frac{e+5}{6}.$
26. $\int_0^1 3e^{x^3} x^2 dx.$ j : $e - 1.$
27. $\int_0^1 \frac{dx}{6-(x+1)^2}.$ j : $\ln \frac{4}{3}.$
28. $\int_3^4 \frac{dx}{x^2 - (x+1)}$ j : $\ln \frac{4}{3}.$
29. $\int_0^1 \frac{dx}{1-\sqrt{x}}.$ j : $4 - \ln 3.$
30. $\int_0^1 e^x \sin x dx.$ j : $\frac{1-e^\pi}{2}.$

14-laboratoriya ishi. Mathcadda simvolli hisoblash turlari orqali hisoblash.

Ishdan maqsad. MathCAD hisoblash jarayonida ishlatiladigan simvolli(analitik) asosiy hisoblash turlari bilan tanishish.

Topshiriq.

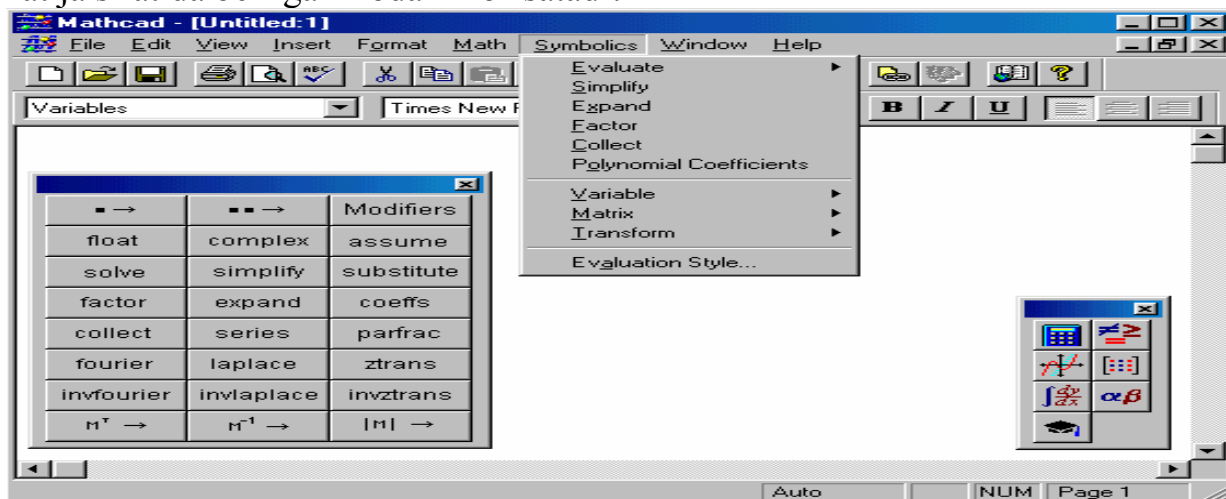
1. Amaliyotda simvolli hisoblash turlari orqali MathCADda hisoblashni o'rganish.
2. Laboratoriya masalalarini bajarish uchun metodik ko'rsatmalarni o'rganish.
3. Tajriba ishining hisobotini tayyorlang.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Ancha vaqtgacha matematik kompyuter dasturlari (MathCAD ning oldingi versiyasi Eureka) raqamli hisoblash tizimi sifatida rivojlanib keldi. Lekin XX asrning 90- yillar boshiga kelib simvolik matematika tizimi tez rivojlanib ketdi. (MathCAD, Maple, MatLab va x.k.). Natijada intellektual ko'rinishdagi analitik (simvolli) hisoblash amallari: funksiyaning limitini va xosilasini topish, aniq va noaniq integrallarni xisolash, funksiyaning qatorlarga yoyish, o'rin almashtirish, kombinatsiyalar yaratish va x. Simvolik hisoblash natijalari analitik ko'rinishda ya'ni formula ko'rinishida beriladi.

MathCADda simvolik hisoblash usulini qo'llash uchun "Symbolics" menyusidan yoki "Simvolno'e vo'chisleniya" palitrasidan foydalanish mumkin. Bu palitrada asosiy operator "Simvolicheskiy znak ravenstva" hisoblanadi (" $\rightarrow$ " tugmasi). Agar "=" belgisi o'rniga " $\rightarrow$ " belgisi ishlatilsa u holda MathCAD raqamli hisoblash o'riniga analitik hisoblash bajaradi. Masalan qatorlar yig'indisini, yoki ko'paytmasini topish, funksiyaning limitini va hosilasini topish, aniq va noaniq integrallarni hisoblash.

Eslatma : Agar tizim simvolik hisoblash jarayonini bajara olmasa, u holda natija sifatida berilgan ifodani ko'rsatadi!



14.1-rasm Simvolik hisoblash usuli menyusi- "Symbolics"

qator yiq'idisi $\sum_{n=1}^{\infty} \frac{1}{n^2} \rightarrow \frac{1}{6} \cdot \pi^2$	hosikf $\frac{d^2}{dx^2} \cos(x) \rightarrow -\cos(x)$
aniq integrak	
noaniq integral $\int_a^b e^x \cdot \sin(x) dx \rightarrow \frac{-1}{2} \cdot \exp(b) \cdot \cos(b) + \frac{1}{2} \cdot \exp(b) \cdot \sin(b) + \frac{1}{2} \cdot \exp(a) \cdot \cos(a) - \frac{1}{2} \cdot \exp(a) \cdot \sin(a)$	limit $\lim_{n \rightarrow 0} \frac{n^2 + 2}{n^2 - 3 \cdot n + 1} \rightarrow 2$

14.2-rasm. Simvolik hisoblash usuliga misollar

Simvolik hisoblash “ palitrasidagi bir nechta operatorlarni ko’rib chiqaylik

- simplify - ifodani soddalashtirish, masalan

$$\frac{2}{x+1} + \frac{3}{(x+1)^2} \text{ simplify } \rightarrow \frac{(2 \cdot x + 5)}{(x+1)^2}$$

expand - biror o’zgaruvchining darajasi bo’yicha yoyish, ifodani ochish,

$$(2x^2 + 5)(x+1)(x^3 - 6) \text{ expand, } x \rightarrow 2x^6 - 7x^3 + 2x^5 - 12x^2 + 5x^4 - 30x - 30$$

factor - ifodani ko’pxadlarga yoyish (expand operatsiyasiga teskari),

$$2x^6 - 7x^3 + 2x^5 - 12x^2 + 5x^4 - 30x - 30 \text{ factor, } x \rightarrow (2x^2 + 5)(x+1)(x^3 - 6)$$

coeffs - polinom koeffitsientlarini topish. Bu operatsiya xuddi expand komandasi kabi bo’lib, faqat natijaviy polinom koeffitsientlarini vektor ko’rinishda beradi.

substitute - ifodada o’zgaruvchini almashtirish .

series - funktsiyani ko’rsatilgan o’zgaruvchi bo’yicha Teylor qatoriga yoyish

$$y(x) := e^x$$

$$y(x) \text{ series}, x, 4 \rightarrow 1 + x + \frac{1}{2} \cdot x^2 + \frac{1}{6} \cdot x^3$$

Bu misolimizda ikkinchi parametr 4 ga teng, qaysiki qatordagi qoldirilgan hadlar sonini anglatadi.

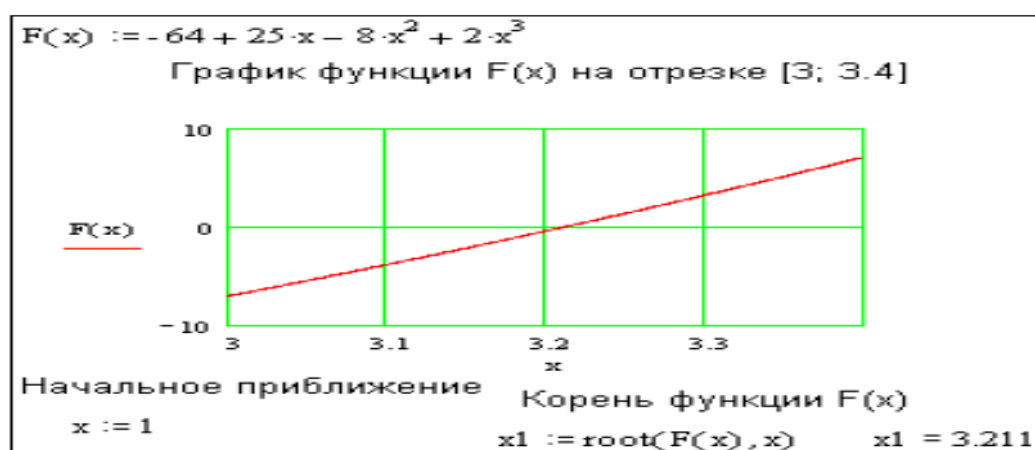
parfrac - ifodani sodda kasirlarga yoyish

$$\frac{(x+4)^2}{x-1} + \frac{x}{x+2x^2-1} \text{ convert, parfrac, } x \rightarrow x+9 + \frac{25}{(x-1)} + \frac{1}{3 \cdot (1+x)} + \frac{1}{3 \cdot (2 \cdot x-1)}$$

solve - ko'rsatilgan o'zgaruvchiga nisbatan tenglamani yoki tengsizlikni echadi Masalan bizga $2x^2 + x - 10 = 0$ tenglamasini yechish kerak. Buning uchun MathCAD ga quyidagi formulani kiritamiz:

$$2 \cdot x^2 + x - 10 \text{ solve}, x \rightarrow \begin{bmatrix} 2 \\ -5 \\ 2 \end{bmatrix}$$

Ba'zi tenglamalar analitik yechimga ega bo'lmasligi mumkin. Bu hollarda sonli metodlarni qo'llashga to'g'ri keladi. MathCAD da biror $F(x)$ funksiyasining taqribiy ildizini topish uchun yordamchi funksiyadan $\text{root}(F(x), x)$ foydalanadi Komandani chaqirishdan avval boshlag'ich qiymatni kiritish lozim. $F(x) = -64 + 25x - 8x^2 + 2x^3$ funksiyasining ildizini toppish ko'rsatilgan. Bunda avval $F(x)$ funktsiyasi aniqlanadi so'ng boshlag'ich intilish nuqtasi $x = 1$ beriladi va ildiz topiladi



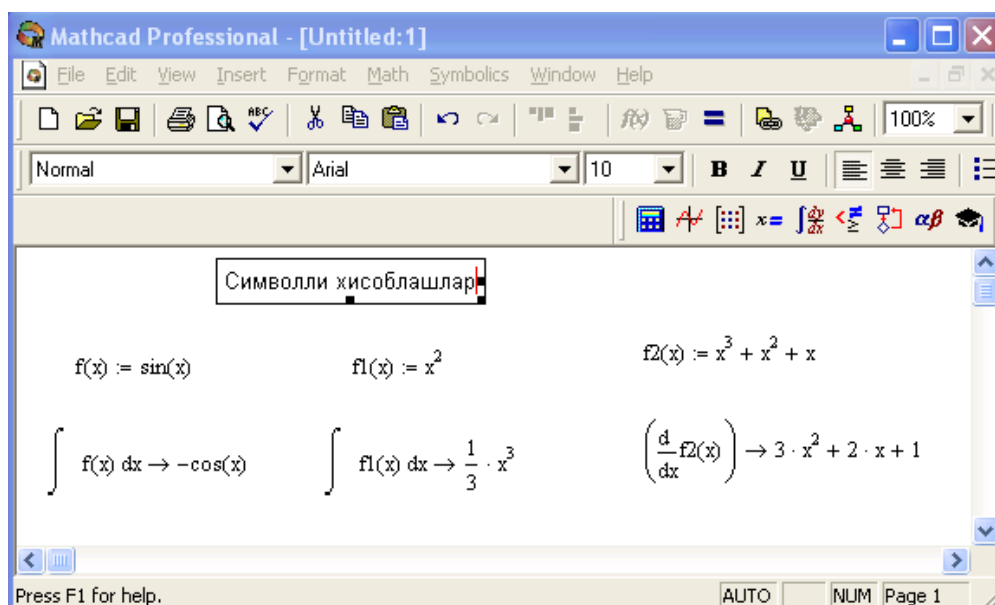
14.3-rasm.

Qiymatlarni global yuborish. Simvolli hisoblashlar.

Ayrim o'zgarmlarga global qiymatni berish uchun quyidagi protsedurani bajarish kerak bo'ladi:

- 1.O'zgarma nomi kiritiladi.
- 2.Matematika panelidan Evaluation Toolbar (Baholash paneli) tugmasi bosiladi.
- 3.Ochilgan Evaluation (Baholash) oynasidan Global Definition (Global aniqlash) tugmasi bosiladi yoki Shift+~ tugmalari baravar bosiladi. Bunday aniqlanish barcha hujjatlar uchun ta'sir qiladi, ya'ni barcha hujjatlarda bu qiymatni ishlatish mumkin.

Sonli hisoblashlardan tashqari Mathcad belgisi (simvolli) hisoblashlarni ham amalga oshiradi. Bu degani hisoblashlar natijasini analitik ko'rinishda tasvirlash mumkin. Masalan, aniqlanmas integral, differentsiallashtirish va boshqa shu kabi masalalarni echishda uning echimini analitik ko'rinishda tasvirlaydi. Bunday oddiy simvolli hisoblashlar 14.4-rasmda keltirilgan.



14.4-rasm. Simvolli hisoblashlarni bajarish.

Simvolli hisoblashlarni bajarishda ikkita asosiy vosita mavjud:

Symbolics (Simvolli hisoblash) menyusi;

Matematika panelidan Symbolic paneli.

Bu vositalar ancha murakkab simvolli hisoblashlarda qo'llaniladi. Hozir esa oddiy simvolli hisoblashni bajarishning eng soddagina usuli, ya'ni tez-tez ishlatilgan usullardan biri - simvolli tenglik belgisi ($\rightarrow$) usulini ko'rib chiqamiz. Quyida bu usuldan foydalanishning ketma-ketlik tartibi berilgan:

1.Matematika panelidan Calculus Toolbar (Hisoblash paneli) tugmasi bosiladi.

2.Ochilgan panel oynasidan Calculus (Hisoblash) ni tanlab, aniqlanmas integralni sichqonchada chiqillatiladi (misol tariqasida aniqlanmas integral qaralayapdi).

3.Kiritish joylari to'ldiriladi, ya'ni funktsiya nomi va o'zgaruvchi nomi kiritiladi.

4. Simvolli belgi tengligi ($\rightarrow$) belgisi kiritiladi.

Vosita	Shablon	Ta'rifi
float	• Float, $\bullet \rightarrow$	Siljuvchi nuqtani hisoblash
complex	• complex, $\bullet \rightarrow$	Kompleks son formasiga o'tkazish
expand	• expand, $\bullet \rightarrow$	Bir necha o'zgaruvchili yig'indi, ko'paytma va darajani ochish
solve	• solve, $\bullet \rightarrow$	Tenglama va tenglamalar tizimini echish
simplify	• simplify, $\bullet \rightarrow$	Ifodalarni ixchamlash
substitute	• substitute, $\bullet \rightarrow$	Ifodalarni hisoblash
collect	• collect, $\bullet \rightarrow$	Oddiy yig'indida tasvirlangan palinom ko'rinishdagi ifodani ixchamlash
series	• series, $\bullet \rightarrow$	Darajali qatorda ifodani yoyish
assume	• assume, $\bullet \rightarrow$	Aniq qiymat bilan yuborilgan o'zgaruvchini hisoblash
parfrac	• parfrac, $\bullet \rightarrow$	Oddiy kasrga ifodalarni yoyish
coeffs	• coeffs, $\bullet \rightarrow$	Polinom koeffitsienti vektorini aniqlash
factor	• factor, $\bullet \rightarrow$	Ifodalarni ko'paytuvchilarga yoyish
fourier	• fourier, $\bullet \rightarrow$	Fure to'g'ri almashtirishi
laplace	• laplace, $\bullet \rightarrow$	Laplas to'g'ri almashtirishi
ztrans	• ztrans, $\bullet \rightarrow$	To'g'ri z-almashtirish
invfourier	• invfourier, $\bullet \rightarrow$	Fure teskari almashtirishi
invlaplace	• invlaplace, $\bullet \rightarrow$	Laplas teskari almashtirishi
invztrans	• invztrans, $\bullet \rightarrow$	Teskari z-almashtirish
$M^T \rightarrow$	• $^T \rightarrow$	Matritsani transponirlash
$M^{-1} \rightarrow$	• $^{-1} \rightarrow$	Matritsaga murojaat
$ M \rightarrow$	• $ \bullet \rightarrow$	Matritsa determinantini hisoblash
Modifiers		Modifier panelini chiqarish

Limitlarni hisoblash. Mathcadda limitlarni hisoblashning uchta operatori bor.

1. Matematika panelidan Calculus Toolbar (Hisoblash paneli) tugmasi basilsa, Colculus (Hisoblash) paneli ochiladi. U yerning pastki qismida limitlarni hisoblash operatorlarini kiritish uchun uchta tugmacha mavjud. Ularning birini bosish kerak.

2. lim so'zining o'ng tomonidagi kiritish joyiga ifoda kiritiladi.

3. lim so'zining ostki qismiga o'zgaruvchi nomi va uning intiladigan qiymati kiritiladi.

4. Barcha ifodalar burchakli kursorda yoki qora ranga ajratiladi.

5. Symbolics $\rightarrow$ Evaluate $\rightarrow$ Symbolically (Simvolli hisoblash $\rightarrow$ Baholash $\rightarrow$ Simvolli) buyruqlari beriladi. Mathcad agar limit mavjud bo'lsa, limitning intilish qiymatini qaytaradi.

15-laboratoriya ishi. Tasvir va grafiklar.

Ishdan maqsad: MathCad dasturida grafiklar bilan tanishish.

Topshiriq:

1. Nazariy qism bilan tanishish.
2. MathCAD dasturida ikki o'lchamli grafik qurish bilan tanishish.
3. MathCAD dasturida uch o'lchamli grafik qurish bilan tanishish.
4. Tajriba ishining hisobotini tayyorlang.

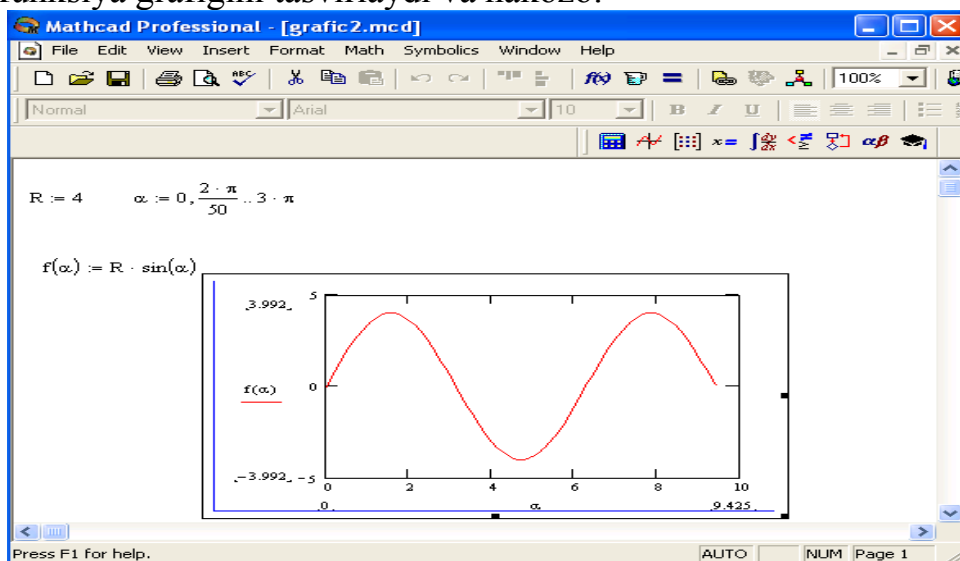
Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Ikki o'lchamli funksiya grafigini qurish uchun quyidagi protseduralarni bajarish kerak.

1. Qaysi joyga grafik qurish kerak bo'lsa, shu joyga krestli kursor qo'yiladi.
2. Matematik panelining Graph (Grafik) panelidan x-y Plot (Ikki o'lchovli grafik) tugmasi bosiladi.
3. Hosil bo'lgan ikki o'lchamli grafik shabloniga abstsiss o'qi argumenti nomi, ordinata o'qi qiga funksiya nomi kiritiladi.
4. Argumentning berilgan o'zgarish diapazonida grafikni qurish uchun grafik shabloni tashqarisi sichqonchada bosiladi. Agar argumentning diapazon qiymati berilmasa, u holda avtomatik holda argument diapazon qiymati 10 dan 10 gacha bo'ladi va shu diapazonda grafik quriladi.

Grafik formatini qayta o'zgartirish uchun grafik maydonini ikki marta tez-tez sichqonchani ko'rsatib bosish va ochilgan muloqot oynasidan kerakli o'zgarishlarni qilish kerak.

Agar bir necha funksiyalar grafigini qurish kerak bo'lsa va ular argumentlari har xil bo'lsa, u holda grafikda funksiyalar va argumentlar nomlari ketma-ket vergul qo'yilib kiritiladi. Bunda birinchi grafik birinchi argument bo'yicha birinchi funksiya grafigini va ikkinchisi esa mos ravishda ikkinchi argument bo'yicha ikkinchi funksiya grafigini tasvirlaydi va hakoza.



15.1-rasm. Funksiya grafigini qurish.

Quyida grafik formati muloqot oynasi qo'yilmalarini beramiz.

1.X-Y Axes – koordinata o`qini formatlash. Koordinata o`qiga setka, sonli qiymatlarni grafikga belgilarni qo`yish ba quyidagilarni o`rnatish mumkin:

LogScale – logarifmik masshtabda o`qga sonli qiymatlarni tasvirlash;

Grid Lines – chiziqqa setkalar qo`yish;

Numbered – koordinata o`qi bo`yicha sonlarni qo`yish;

Auto Scale – son qiymatlar chegarasini o`qda avtomatik tanlash;

Show Markers – grafikka belgi kiritish;

Autogrid – chiziq setkasi sonini avtomatik tanlash.

2.Trace – funksiya grafiklarini formatlash. Har bir funksiya grafigini alohida o`zgartish mumkin:

chiziq ko`rinishi (Solid – uzliksiz, Dot – punktir, Dash – shtrixli, Dadot – shtrixli punktir);

chiziq rangi (Color);

grafik tipi (Type) (Lines – chiziq, Points – nuqtali, Bar yoki SolidBar – ustunli, Step – pog`onali grafik va boshqa);

chiziq qalinligi (Weight);

simvol (Symbol) - grafikda hisoblangan qiymatlar uchun (aylana, krestik, to`g`ri burchak, romb).

3.Label – grafik maydoni sarlovhasi. Title (Sarlovha) maydoniga sarlovha matni kiritiladi.

4.Defaults – bu qo`yilma yordamida grafik ko`rinishga qaytish mumkin.

Uch o`lchamli grafik qurish

Uch o`lchamli grafik qurish uchun quyidagi protseduralarni bajarish kerak.

1.Ikki o`zgaruvchili funksiya nomini keyin (:=) yuborish operatori va funksiya ifodasini kiritish.

2.Grafik qurish kerak bo`lgan joyga kursor qo`yiladi.

3.Matematik panelining Graph (Grafik) panelidan Surface Plot (uch o`lchamli grafik) tugmasi bosiladi. Shu joyda uch o`lchamli grafik shablona paydo bo`ladi.

4.Shablon maydonidan tashqarisida sichqoncha bosiladi va grafik quriladi, masalan, 6-rasm chap tomon.

Ikki o`zgaruvchili funksiya bo`yicha grafik sirtini qurishni tez qilish maqsadida boshqa usul ham mavjud va u ayrim hollarda funksiya sirtini tuzishda funksiya massiv sonli qiymatlarini ishlatadi, masalan, 6-rasm chap tomon. Bunday grafikni qurish uchun quyidagi protseduralarni bajarish kerak.

1.Diskret o`zgaruvchilar yordamida ikki funksiyaning o`zgaruvchisi uchun ham qiymatlarini kiritish.

2.Massiv kiritish. Uning elementlari funksiya qiymatlari bo`lib, ular berilgan funksiya argumentlari qiymatlaridan tashkil etiladi.

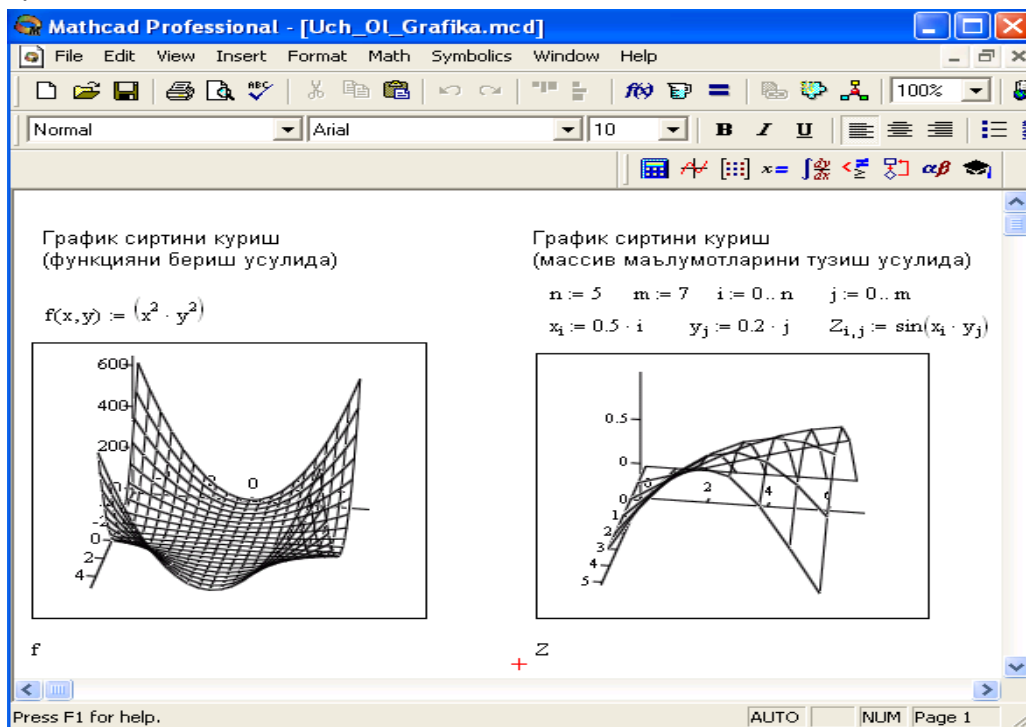
3.Kursor qaysi joyga grafik qurish kerak bo`lsa shu joyga qo`yiladi.

4.Grafik shabloniga funksiya nomi kiritiladi.

5.Shablon maydonidan tashqarisida sichqoncha bosiladi va grafik quriladi, masalan, 6-rasm o`ng tomon.

Grafik formatini qayta o`zgartirish va unga ranglar berish uchun grafik maydonini ikki marta tez-tez sichqonchani ko`rsatib bosish va ochilgan muloqot oynasidan

kerakli o'zgarishlarni qilish kerak. Bu o'zgartirishlar muloqot oynasi 7-rasmda berilgan.



15.2-rasm. Ikki o'zgaruvchili funktsiya grafigini qurish.

Bunda:

- Surface Plot – grafik sirti;
- Contour Plot – grafik chizig'i darajasi;
- Data Points – grafikda faqat hisob nuqtalarini tasvirlash;
- Vector Field Plot – vektor maydoni grafigi;
- Bar Plot – uch o'lchovli grafik gistogrammasi;
- Patch plot – hisob qiymatlari maydoni.

Bulardan tashqari yana bir qancha boshqarish elementlari mavjud. Ular grafikni formatlashda keng imkoniyatni beradi. Masalan, grafik masshtabini o'zgartirish, grafikni aylantirish, grafikga animatsiya berish va boshqa.

Grafikni boshqarishning boshqa usullari quyidagilar:

Grafikni aylantirish uni ko'rsatib sichqoncha o'ng tugmasini bosish bilan amalga oshiriladi.

Grafikni masshtablashtirish Ctrl tugmasini bosib sichqoncha orqali bajariladi.

Grafikga animatsiya berish Shift tugmasini bosish bilan sichqoncha orqali amalga oshiriladi.

16-laboratoriya ishi. Dasturlash funksiyalari.

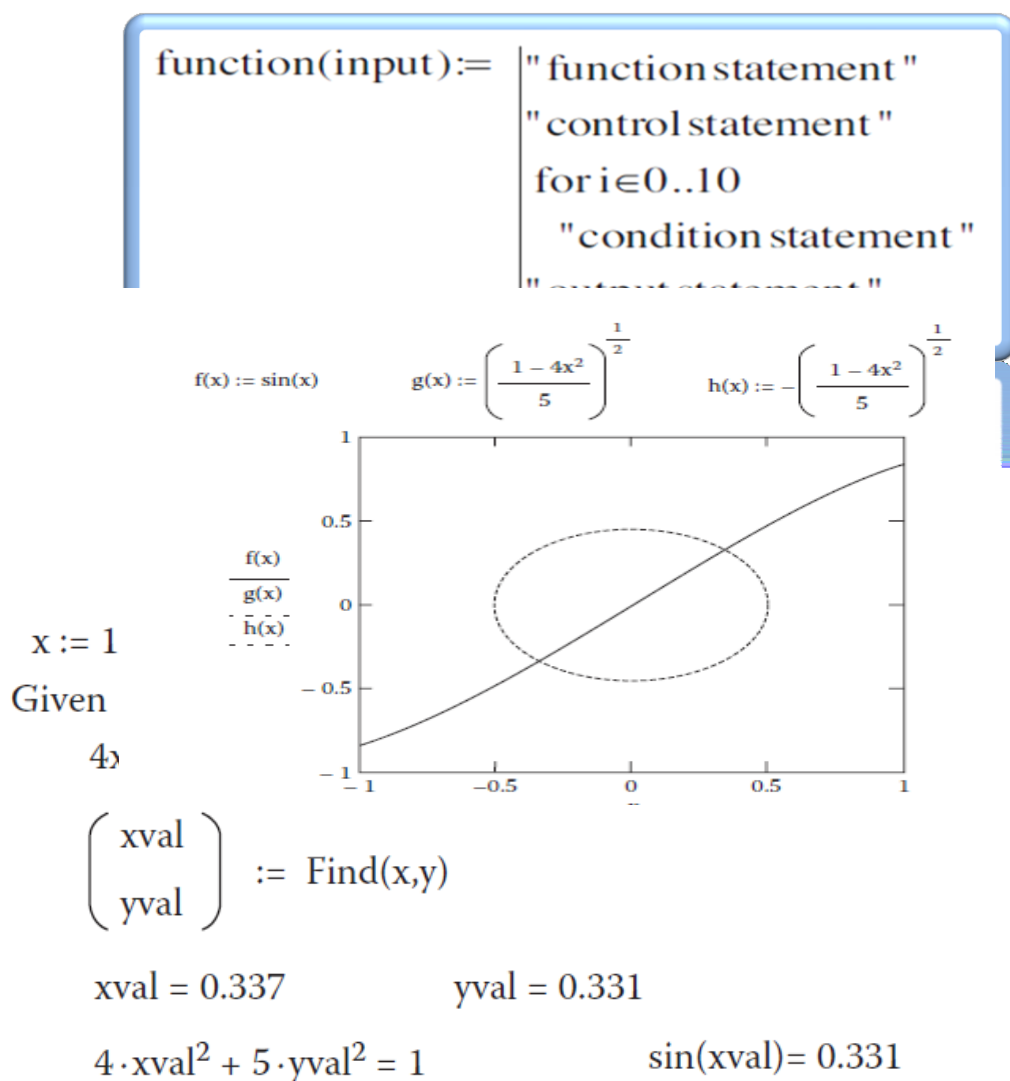
Ishdan maqsad. MathCAD hisoblash jarayonida ishlatiladigan hisoblash turlari bilan tanishish.

Topshiriq.

1. Nazariy qism bilan tanishish.
2. Dasturlash funksiyalari bilan ishlash.
3. For va while funksiyalari bilan ishlash.
4. Topshiriq hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

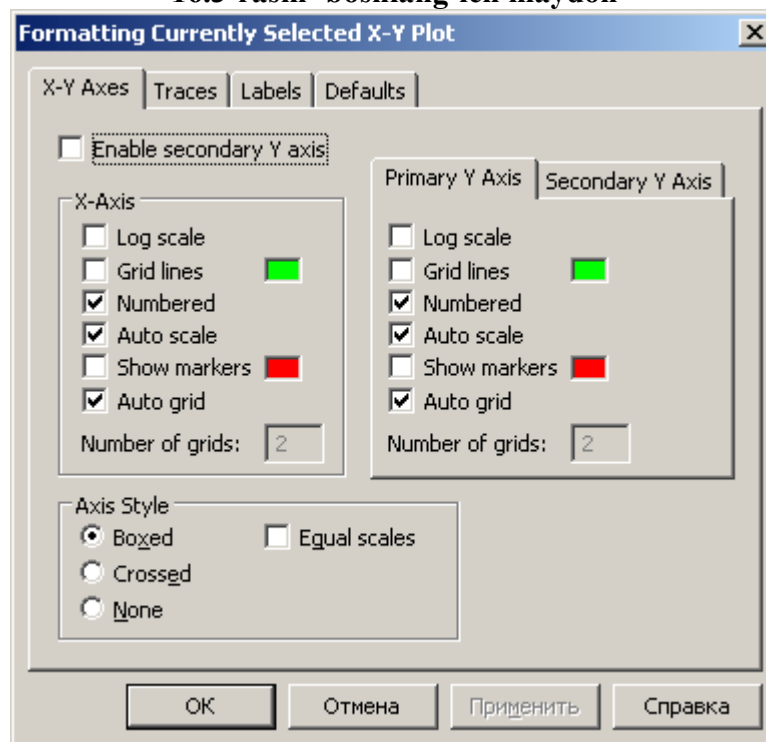
MathCad da quyidagi dasturlash funksiyalarini yaratish mumkin. Avtorlar dasturlash funksiyalarning imkoniyatlarini yanada chuqurroq o'rganishni kitobxonga tavsiya etib, lekin funksiyalar bo'yicha boshqarish strukturasi shu jumladan for – sikl funksiyasi, while- sikl funksiyasi va if- e'lon funksiyalari uchun ko'rsatadi. Yakuniy boshqarishning umumiy formati quyidagicha ko'rinadi.



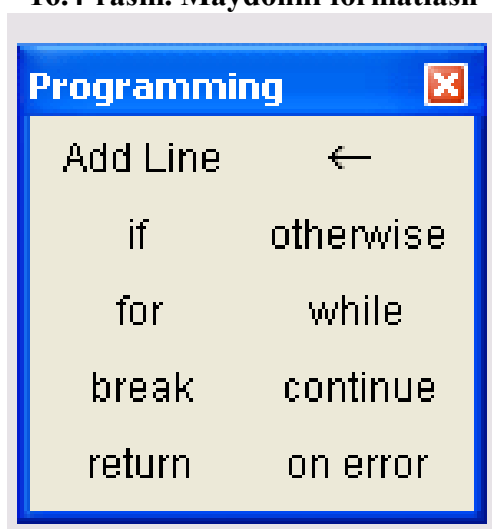
MathCadda Find operatorini ishlatilishi



16.3-rasm- boshlang'ich maydon



16.4-rasm. Maydonni formatlash



16.5-rasm. Dasturlash instrumentlar paneli.

Xar bir boshqarish struktura turi o'zining konstruksiyasida ozgina farq qiladi, lekin barcha operatorlari dasturlash instrumentlar panelidan topish mumkin. For sikl funksiyasi ko'rsatilgan diapazon uchun kod satrini bajaradi., shuning uchun bir qator o'zgaruvchilar va operatorlar talab qilinmaydi. Sodda misol quyida ko'rsatilgan

```

sum(x):=
| s ← 0
| for i ∈ 0..x
|   s ← s + i
| s

```

$\text{sum}(5) = 15$

While sikl operatori xuddi for sikl operatori kabi oʻrnatilib, faqat While tanlaniladi. Oʻrnatilgan shart While kriteriyasi bajarilguncha ishlaydi. If sikl operatori xam instrumentlar panelidan If ni tanlash orqali oʻrnatiladi. Agar oʻrnatilgan shartning mantiqiy ifoda qiymati True (xa) boʻlgandagina If operator ishlaydi. Boshqarish operatorlarini yangi satrlar qoʻshish orqali murakkablashtirish mumkin. Shuni esda saqlash kerak, foydalanuvchi klaviatura orqali while, if, yoki for oʻzlarini termasdan dasturlash instrumentlar panelidan tanlash zarur. Shuningdek MathCad indekslarini nuldin boshlaydi va operator indeksi (matritsa instrumentlar panelida joylashgan) aniq elementini chaqirish uchun foydalaniladi.

17-laboratoriya ishi. Chiziqli tenglamalar tizimlarini yechish.

Ishdan maqsad. MathCADda chiziqli tenglamalar tizimlarini yechish turlari bilan tanishish.

Topshiriq.

1. Nazariy qism bilan tanishish.
2. Masalaning qoʻyilishi.
3. Tenglamalarni yechish va tahlil qilish.
4. Topshiriq hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qoʻllanmalar, koʻrgazmalar.

Qisqacha nazariy maʼlumot: Mathcad har qanday tenglamani, hamda koʻpgina differensial va integral tenglamalarni yechish imkoniyatini beradi. Misol uchun kvadrat tenglamaning oldin simvolli yechimini topishni keyin esa sonli yechimini topishni qarab chiqamiz.

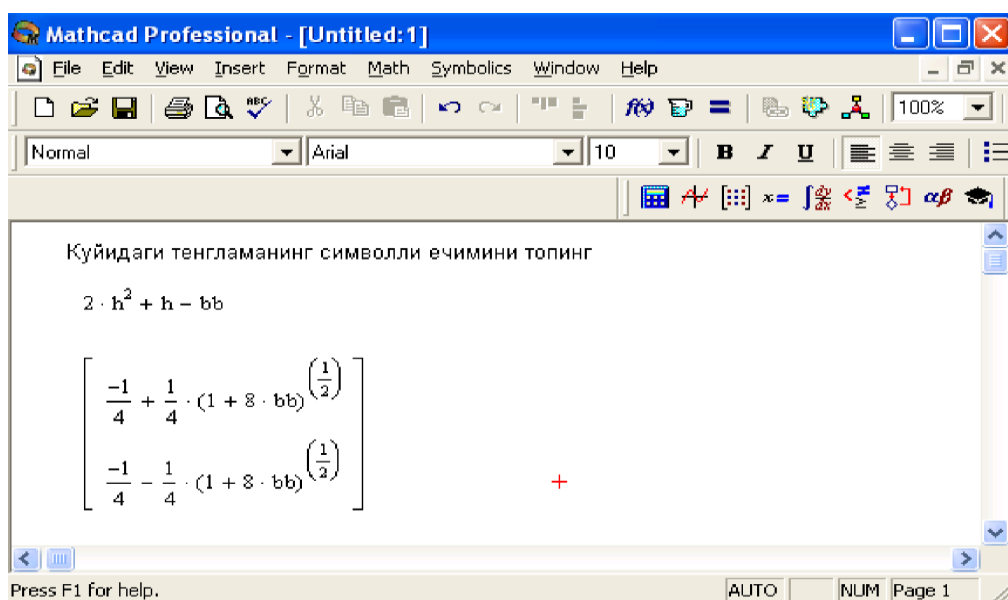
Simvolli yechish. Tenglamaning simvolli yechimini topish uchun quyidagi protsedurani bajarish kerak:

1. Yechiladigan tenglamani kiritish va tenglama yechimi boʻlgan oʻzgaruvchini kursorning koʻk burchagida ajratish.

2. Bosh menyudan Symbolics—»Variable—»Solve (Simvolli ifoda—Oʻzgaruvchi—(Yechish) buyrugʻini tanlash. Tenglamani yechish 1-rasmda keltirilgan.

Sonli yechish. Algebraik tenglamalarni yechish uchun Mathcadda bir necha funktsiyalar mavjud. Ulardan Root funktsiyasini koʻrib chiqamiz. Bu funktsiyaga murojaat quyidagicha:

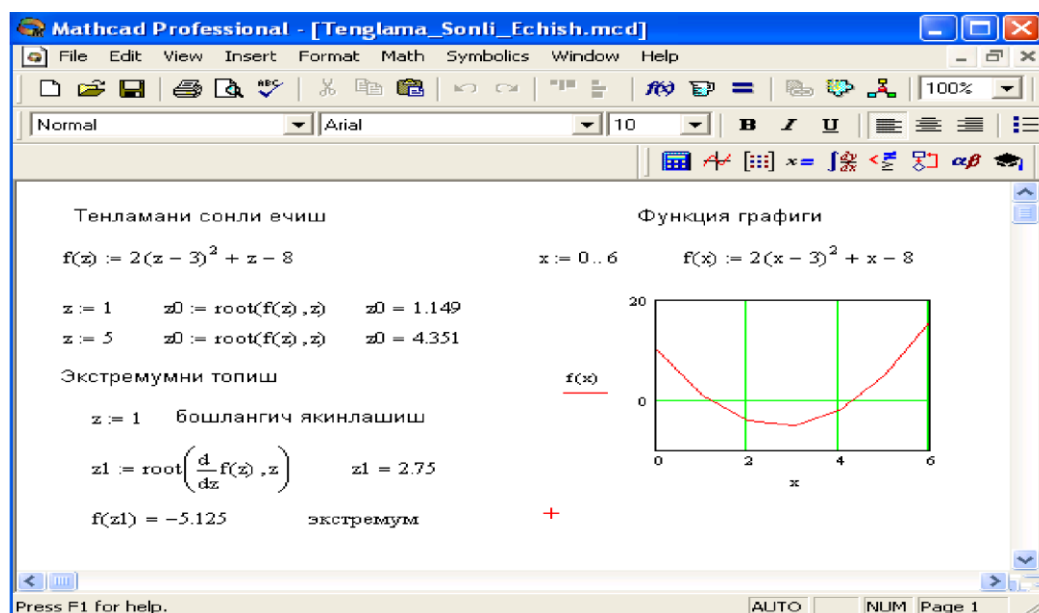
Root(f(x),x).



17.1-rasm. Tenglamani simvolli yechish.

Root funksiyasi iteratsiya usuli bilan yechadi va sabab boshlang'ich qiymat oldindan talab etilmaydi. 17.2-rasmda tenglamani sonli yechish va uning ekstremumini topish keltirilgan.

Tenglamani yechish uchun odlin uning grafigi quriladi va keyin uning sonli yechimi izlanadi. Funksiyaga murojaat qilishdan oldin yechimga yaqin qiymat beriladi va keyin Root funksiya kiritilib, x_0 = beriladi.



17.2-rasm. Tenglamani sonli yechish va uning grafigini qurish.

Root funksiyasi yordamida funksiya hosilasini nulgga tenglashtirib uning ekstremumini ham topish mumkin. Funksiya ekstremumini topish uchun quyidagi protsedurani bajarish kerak: 1. Ekstremum nuqtasiga boshlang'ich yaqinlashishni berish kerak.

2. Root funksiyasini yozib uning ichiga birinchi tartibli differensialni va o'zgaruvchini kiritish.

3.Ozgaruvchini yozib teng belgisini kiritish.

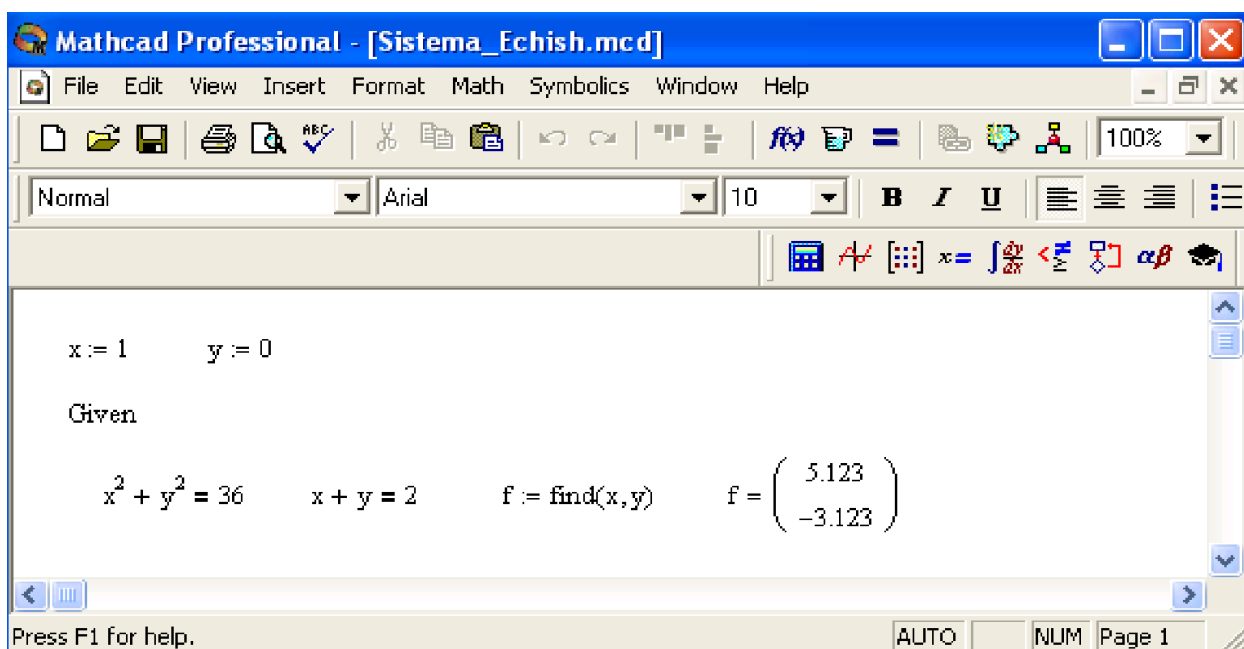
4.Funksiyani yozib teng belgisini kiritish. Root funksiyasi yordamida tenglamaning simvolli echimini ham olish mumkin. Buning uchun boshlang'ich yaqinlashish talab etilmaydi. Root funksiya ichiga oluvchi ifodani kiritish kifoyadir (masalan, $\text{Root}(2h^2+h-bb,h)$). Keyin Ctrl+. klavishasini birgalikda bosish kerak. Agrar simvolli yechim mavjud bo'lsa, u paydo bo'ladi.

Tenglamalar tizimini yechish

Mathcadda tenglamalar tizimini yechish Given...Find hisoblash bloki yordamida amalga oshiriladi. Tenglamalar tizimini yechish uchun iteratsiya usuli qo'llaniladi va yechishdan oldin boshlang'ich yaqinlashish barcha noma'lumlar uchun beriladi(17.2 -rasm). Tenglamalar tizimini yechish uchun quyidagi protsedurani bajarish kerak:

1.Tizimga kiruvchi barcha noma'lumlar uchun boshlang'ich yaqinlashishlarni berish.

2. Given kalit so'zi kiritiladi.

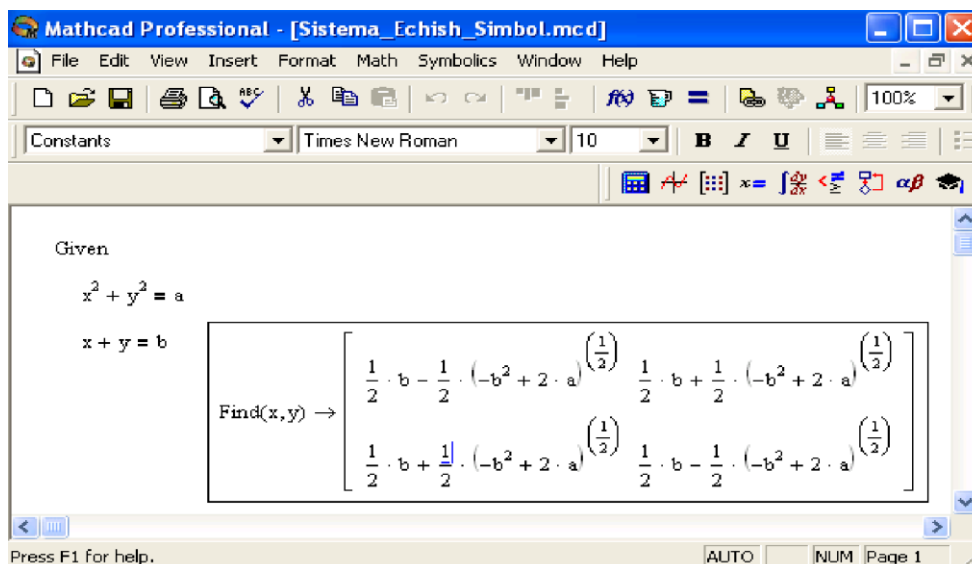


17.3-rasm. Chiziqsiz tenglamalar tizimini yechish.

3.Tizimga kiruvchi tenglama va tengsizlik kiritiladi. Tenglik belgisi qalin bo'lishi kerak, buning uchunCtrl+= klavishilarini birgalikda bosish kerak bo'ladi yoki Boolean (Bul operatorlari) panelidan foydalanish mumkin.

4.Find funksiyasi tarkibiga kiruvchi o'zgaruvchi yoki ifodani kiritish. Funksiyaga murojaat quyidagicha bajariladi: Find(x,y,z). Bu erda x,y,z - noma'lumlar. Noma'lumlar soni tenglamalar soniga teng bo'lishi kerak.

Find funksiyasi Root ga o'xshab tenglamalar tizimini sonli yechish bilan bir qatorda, yechimni simvolli ko'rinishda ham topish imkonini beradi (17.4 - rasm).



17.4-rasm. Chiziqsiz tenglamalar tizimini simvoli yechimini topish.

Chiziqli dasturlash masalalarini yechish

Chiziqli dasturlash masalasining umumlashgan matematik modeli formasining yozilishi quyidagi ko'rinishga ega.

$$Za..x. < b., \quad (i = 1, m)$$

$$j=1$$

$$x. > 0$$

$$j$$

$$(J = 1, n)$$

$$Z = 2^{\wedge} c_j \rightarrow \max(\min) j=1$$

Matematik modelning birinchi formulasi iqtisodiy ma'noda izlanayotgan miqdorlarga qo'yiladigan cheklanishlarni ifodalaydi, ular resurslar miqdori, ma'lum talablarni qondirish zarurati, texnologiya sharoiti va boshqa iqtisodiy hamda texnikaviy faktorlardan kelib chiqadi. Ikkinchi shart - o'zgaruvchilarning, yani izlanayotgan miqdorlarning manfiy bo'lmaslik sharti bo'lib hisoblanadi. Uchinchisi maqsad funksiyasi deyilib, izlanayotgan miqdorning biror bog'lanishini ifodalaydi. Chiziqli dasturlash masalasiga keluvchi quyidagi masalani qaraymiz.

Fabrika ikki xil A va V tikuv maxsuloti ishlab chiqaradi. Bu mahsulotlarni ishlab chiqarishda uch xil N_1, N_2, N_3 turdagi materiallarni ishlatadi. N_1 -materialdan 15 m., N_2 -materialdan 16 m., N_3 -materialdan 18 m. mavjud.

M_1 -mahsulotni ishlab chiqarish uchun N_1 -dan 2 m., N_2 -dan 1 m., N_3 -dan 3 m. ishlatadi.

M_2 - mahsulotni ishlab chiqarish uchun N_1 -dan 3 m., N_2 -dan 4 m., N_3 -dan 0 m. ishlatadi.

M_1 - mahsulotning bir birligidan keladigan foyda 10 so'mni, M_2 - mahsulotdan keladigan foyda 5 so'mni tashkil qiladi.

Ishlab chiqarishning shunday planini tuzish kerakki fabrika maksimal foyda olsun. Masalaning matematik modelini tuzamiz:

$$2x_1 + 3x_2 < 15$$

$$3x_1 < 18$$

$$x_1 > 0, \quad x_2 > 0 \quad Z = 10x_1 + 5x_2 \rightarrow \max$$

Mathcadda chiziqli dasturlash masalasi yechishda maximize va minimize funksiyalaridan foydalanish mumkin. Bu funksiyalar umumiy holda quyidagi ko'rinishda yoziladi:

Maximize($F, \langle \text{o'zgaruvchilar ro'yxati} \rangle$) Minimize($F, \langle \text{o'zgaruvchilar ro'yxati} \rangle$) Mathcadda chiziqli dasturlash masalasini yechish quyidagicha bajariladi

1.Mathcadni ishga tushurgandan so'ng, maqsad funksiyasi yoziladi, masalan $f(x,y)=\langle \text{funksiya ko'rinishi} \rangle$ va o'zgaruvchilarning boshlang'ich qiymati kiritiladi.

2.Given kalit so'zi yoziladi.

3.Tengsizliklar tizimi va cheklanishlar kiritiladi.

4.Brur o'zgaruvchiga maximize yoki imimize funksiyasi yuboriladi.

5.Shu o'zgaruvchi yozilib tenglik kiritiladi. Natija vektor ko'rinishida hosil bo'ladi.

6.Maqsad funksiyasi qiymatini hisoblash uchun, masalan $f(p_0, p_1)$ yozilib tenglik belgisi kiritiladi.

18-laboratoriya ishi. MathCad dasturidan foydalanib masalalar tenglamalarni yechish.

Ishdan maqsad. MathCAD dasturidan foydalanib masalalar tenglamalarni yechish.

Topshiriq.

1.Nazariy qism bilan tanishish.

2. Matritsalar ustida amallar bajarish.

3.Tenglamalarni yechish va tahlil qilish.

4.Topshiriq hisobotini tayyorlash.

Jihoz va materiallar: kompyuter, adabiyotlar, qo'llanmalar, ko'rgazmalar.

Matematik masalalarni yechishda MathCadning xizmati matritsalar ustida amallar bajarishda yaqqol ko'rinadi. Matritsalar katta bo'lganda bu amallarni bajarish ancha murakkab bo'lib, kompyuterda MathCadda dastur tuzishni talab etadi. MathCad tizimida bunday ishlarni tez va yaqqol ko'rinishda amalga oshirsa bo'ladi.

Matritsani tuzish. Matritsa yoki vektorni quyidagi protsedura yordamida aniqlash mumkin:

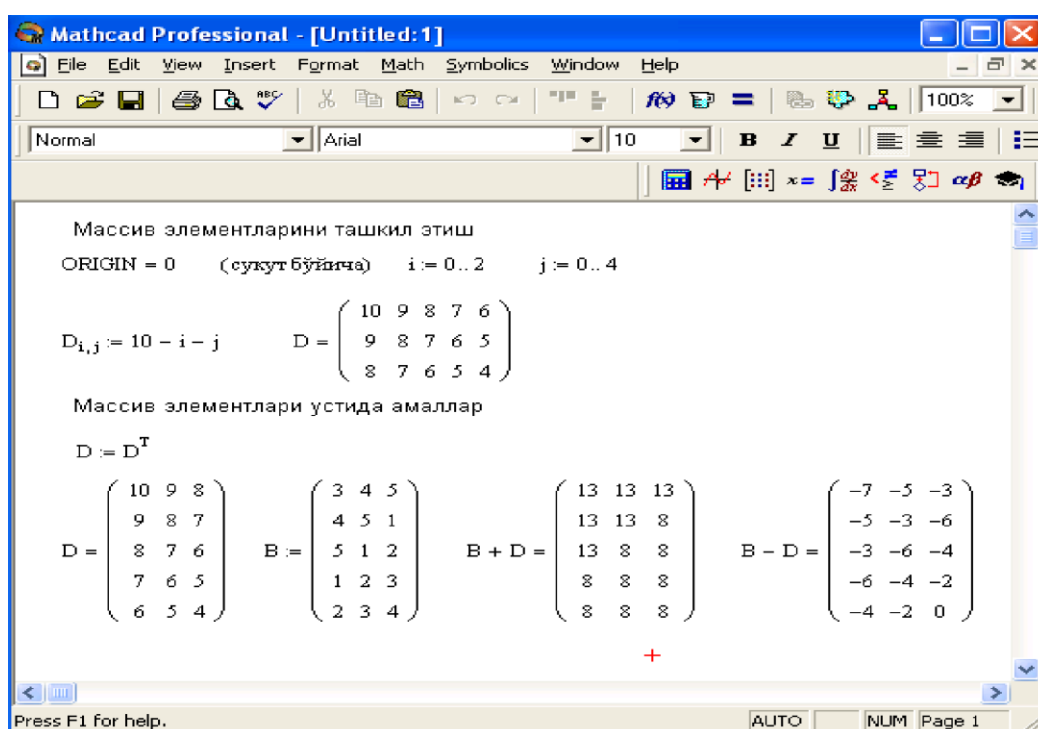
1.Matritsa nomini va ($:=$) yuborish operatorini kiritish.

2.Matematika panelidan Vector and Matrix Toolbar (Matritsa va vektor paneli) tugmachasi bosiladi. Keyin Matrix or Vector (Matritsa va vektor) tugmasi bosiladi, natijada Matrix (Matritsa) paneli ochiladi. Ochilgan muloqot oynasidan ustun va satr sonlari kiritilib Ok tugmasi bosiladi. Bu holda ekranda matritsa shabloni paydo bo'ladi.

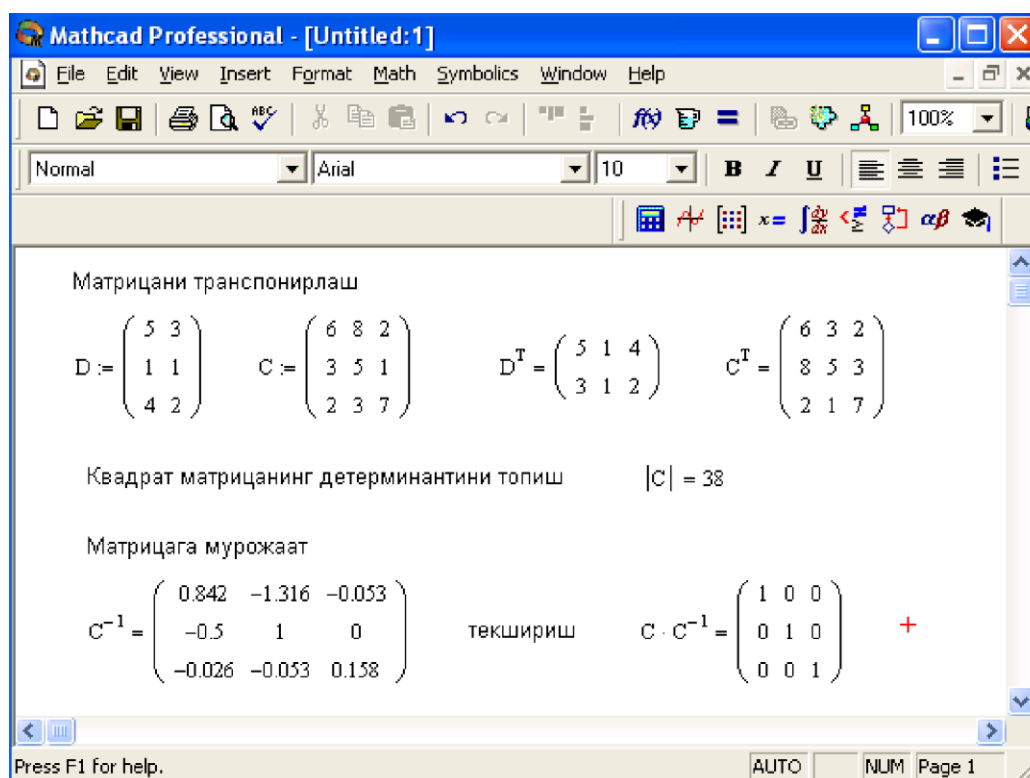
3.Har bir joy sonlar bilan to'ldiriladi, ya'ni matritsa elementlari kiritiladi. Shablon yordamida 100 dan ortiq elementga ega bo'lgan matritsani kiritish mumkin. Vektor - bu bir ustunli matritsa deb qabul qilinadi. Har qanday matritsa elementi matritsa nomi bilan uning ikki indeks orqali aniqlanadi. Birinchi indeks qator nomerini, ikkinchi indeks - ustun nomerini bildiradi. Indeksni kiritish uchun matematika vositalar panelidan Matrix panelini ochib, u erdan Vector and

Matrix Toolbar, keyin Subscript (Pastki indeks) bosiladi. Klaviaturadan buni [(ochuvchi kvadrat qavs) yordamida bajarsa ham bo'ladi. Massiv elementi nomeri 0, 1 yoki istalgan sondan boshlanishi mumkin (musbat yoki manfiy). Massiv elementi nomeri boshqarish uchun maxsus ORIGIN nomli o'zgaruvchi ishlatiladi. Avtomatik 0 uchun ORIGIN=0 deb yoziladi. Bunda massiv elementlari nomeri nuldin boshlanadi. Agar nuldin boshqa sondan boshlansa unda ORIGIN dan keyin ikki nuqta qoyiladi, masalan ORIGIN:=1. ORIGIN=0 bo'lgani uchun avtomatik ravishda birinchi element 10 ga teng.

Matritsalar ustida asosiy amallar. MathCad matritsalar bilan quyidagi arifmetik operatsiyalarni bajaradi: matritsani matritsaga qo'shish, ayirish va ko'paytirish, bundan tashqari transponirlash operatsiyasini, murojaat qilish, matritsa determinantini hisoblash, maxsus son va maxsus vektorni topish va boshqa. Bu operatsiyalarning bajarilishi 18.1, 18.2 -rasmlarda keltirilgan.

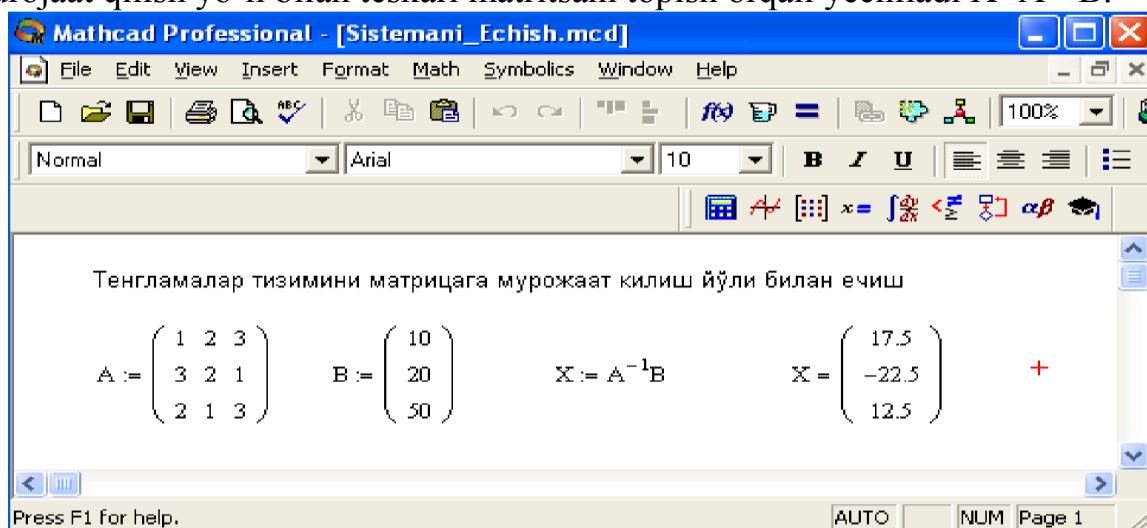


18.1-rasmda D matritsaning pastki indekslardan foydalanib elementlarini topish ko'rsatilgan.



18.2-rasm. Matritsa ustida amallar bajarish.

Matritsali tenglamalarni yechish. Matritsali tenglamalar bu chiziqli algebraik tenglamalar tizimi bo'lib $A \cdot X = B$ ko'rinishda yoziladi va u matritsaga murojaat qilish yo'li bilan teskari matritsani topish orqali yechiladi $X = A^{-1} \cdot B$.



18.3-rasm. Tenglamalar tizimini matritsa usulida yechish.

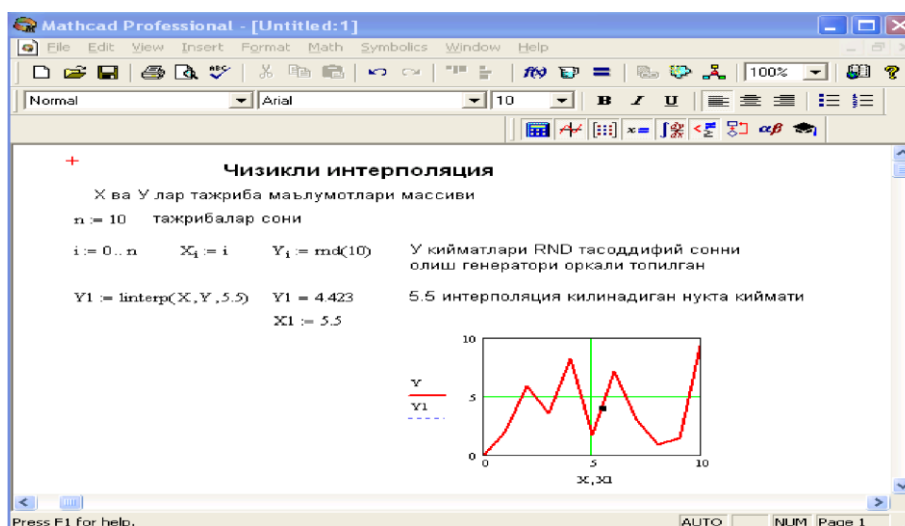
Matritsalar ustida simvolli operatsiyalar Symbolics (Simvolli hisoblash) menyusining buyruqlari va simvolli tenglik belgisi ($\rightarrow$) yordamida bajariladi.

Tajriba natijalarini tahlil qilishga doir masalalarni yechish

Turli tajribalarni o'tkazishda odatda tajriba ma'lumotlarini funktsiya ko'rinishida tasvirlash va ularni keyingi hisoblashlarda ishlatish uchun massivlar kerak bo'ladi. Agar funktsiyani tasvirlovchi egri chiziq barcha tajriba nuqtalaridan

o'tish kerak bo'lsa, u holda olingan oraliq nuqtalar va hisoblangan funktsiyaga interpolatsiya deyiladi. Agar funktsiyani tasvirlovchi egri chiziq barcha tajriba nuqtalaridan o'tish kerak bo'lmasa, u holda olingan oraliq nuqtalar va hisoblangan funktsiyaga regressiya deyiladi.

Interpolyatsiya. MathCad bir necha interpolatsiyalash funktsiyalariga ega bo'lib, ular har xil usullarni ishlatadi. Chiziqli interpolatsiyalash jarayonida **linterp** funktsiyasidan foydalaniladi.



18.4-rasm. Interpoyatsiyalash.

Bu funktsiyaga murojaat quyidagicha:

linterp(x, y, t)

Bu erda

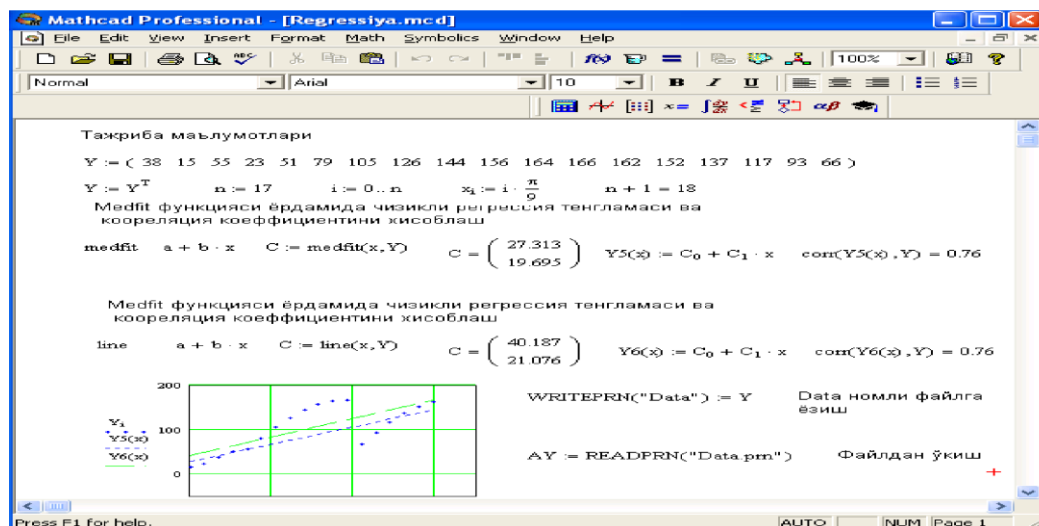
- x - argument qiymati vektori;
- y - funktsiya qiymatlari vektori;
- t - interpolatsiya funktsiyasi hisoblanadigan mos argument qiymati.

Regressiya. Regressiya ma'nosi tajriba ma'lumotlarini approksimatsiya qiladigan funktsiya ko'rinishini aniqlashdir. Regressiya u yoki bu analitik bog'lanishning koeffitsientlarini tanlashga keladi.

MathCadda ikki xildagi bir necha qurilgan regressiya funktsiyalari mavjud. Ular quyidagilar:

- line(X,Y) -xatolar yig'indisi kvadratini minimallashtiruvchi to'g'ri chiziqli regressiya $f(t)=a+b \cdot t$;
- medfit(X,Y) -median to'g'ri chiziqli regressiya $f(t)=a+b \cdot t$;
- lnfit(X,Y) -logarifmik funktsiyali regressiya $f(t)=a \ln(t)+b$.

Bu regressiya funktsiyalari boshlang'ich yaqinlashishni talab etmaydi. Ularga doir misollar 18.5-rasmda keltirilgan.



18.5-rasm.Chiziqli regressiya tenlamasini tuzish.

Yana beshta qurilgan funksiyalar mavjud bo'lib ular boshlang'ich yaqinlashishni talab etadi:

- $\text{expfit}(X, Y, g)$ -eksponentali regressiya $f(x) = ae^{bt} + c$;
- $\text{sinfite}(X, Y, g)$ - sinisoid regressiya $f(x) = a \sin(t + b) + c$;
- $\text{pwrfit}(X, Y, g)$ - darajaga bog'liq regressiya $f(x) = at^b + c$;
- $\text{lgsfit}(X, Y, g)$ - logistik funksiyali regressiya $a(e) = a / (1 + be^{kt})$;
- $\text{logfit}(X, Y, g)$ - logorifmik funksiyali regressiya $f(t) = a \ln(t + b) + c$. Bu funksiyalarda
- x - argument qiymatlari vektori;
- y - funktsiya qiymatlari vektori
- g - a,b,c koeffisientlar boshlang'ich yaqinlashish qiymatlari vektori;
- t - interpolatsiya qilinayotgan funksiya hisoblanayotgan argument qiymati.

Yuqoridagi rasmlarda massiv (tajriba) ma'lumotlari bilan approksimatsiyalangan funksiya orasidagi bog'liqlikni baholash uchun koorelyatsiya koeffisienti corr hisoblangan.

Mustaqil ta'lim va mustaqil ishlar

Mustaqil ishni bajarishdan asosiy maqsad — professor-o'qituvchilarning bevosita rahbarligi va nazorati ostida talabalarni semestr davomida fanlarni uzluksiz o'rganishini tashkil etish, olingan bilim, ko'nikmalarini chuqur o'rganib mustahkamlash, eng soddaki kimyo injiniringda masalalarini yechish hamda tahlil qilish shuningdek o'z mutaxassisliklari haqida yanada tasavvurlarini boyitish, yangi bilimlarni mustaqil ravishda izlab topish va qabul qilishini shakllanirishdan iborat.

Talaba mustaqil ishini tashkil etishda "Kimyo injiniringda axborot texnologiyalar" fanining xususiyatlarini, shuningdek, har bir talabaning akademik o'zlashtirish darajasi va qobiliyatini hisobga olgan holda quyidagi shakllardan foydalaniladi: ayrim nazariy mavzularni turli xil o'quv adabiyotlari yordamida mustaqil o'zlashtirish, berilgan mavzu bo'yicha referat tayyorlash, laboratoriya va amaliy mashg'ulotlarga tayyorgarlik ko'rish, kurs ishini bajarish, bitiruv malakaviy ishini tayyorlash, anjumanlarga ma'ruza tezislarni tayyorlash, fan olimpiadalarida ishtirok etish, ilmiy jamiyatlar, to'garaklarda, kafedralarning ilmiy ishlarida ishtirok etish va hokazolar.

Tavsiya etilayotgan mustaqil ishlarning mavzulari

1.	Axborot texnologiyalari tushunchasi va mohiyati. Uni vujudga kelishi va rivojlanishini asosiy bosqichlari
2.	Kompyuterda masalalarni yechish bosqichlari. Algoritm tushunchasi va uning xossalari
3.	Masalaning algoritmini yaratish.
4.	Algoritmlar turlari. Chiziqli hisoblash jarayonlarini algoritmlash.
5.	Tarmoqlanuvchi algoritmlar tuzish
6.	Takrorlanuvchi algoritmlar tuzish.
7.	Masalalarni algoritmlash va dasturlashtirish asoslari. Algoritmlarga qo'yiladigan talablar.
8.	Algoritmlarni ta'riflash usullari. Dasturiy ta'minotning tarkibi va amaliy dasturlar paketi.
9.	Mamlakatda kompyuterlashtirish, ishlab chiqarishda axborot texnologiyalarni keng qo'llash va axborot kommunikatsiya texnologiyalarini rivojlantirish ushbu fanni o'qitishdagi roli va ahamiyati.
10	Dasturlash. Kompyuter anatomiyasi, mashina kodi va dasturlash tillari, dasturlash muhiti bilan tanishish, ilk dasturni tahlil qilish, xatolar, muammoni yechish, dizayn algoritmi.
11	MS Excel elektron jadvalining ishchi oynasi, menyu bo'limlari.
12	MS Excel elektron jadvali va listlarni boshqarish.
13	Funksiyalar masteri bilan ishlash. Diagrammalar, grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.

14	Funktsiya masteri fx dan foydalanib formulalar kiritish va undan foydalanish. Shuningdek trigonometric funksiyalar (SIN, COS, TAN, ACOS, va boshqalar), Statistik funksiyalari (MEAN (o'rta), AVERAGE (o'rtacha)).
15	Trigonometrik funksiyalar t-test (t-sinovi) arifmetik funksiyalar (SUM, ROUND, LOG, LOG10) mantiqiy, (IF (balki) FALSE (hato) va boshqalar) va boshqalar bilan ishlash. VBA (Visual Basic For Applications)
16	MS Excelda Solver imkoniyatlarini o'rganish, kimyo texnologiyasi masalalarini yechishda MS Excelda Solver imkoniyatlarini o'rganish.
17	C++ algoritmik tilida masala dasturini tuzish.
18	O'zgaruvchilar, o'zgarmaslar, arifmetik amallar, kiritish va natija, muammo yechimi va boshqalar.
19	if operatori, sonlarni va qatorlarni solishtirish, birdan ortiq tanlovlar, muammo yechimi, operatorlar va o'zgaruvchilar
20	Funksiyalarni joriy qilish, parametrni o'tkazish, qaytish qiymatlari, qaytish qiymatlarisiz funksiyalar, qayta ishlatiladigan funksiyalar, aniqlash maydoni va global o'zgaruvchilar, izohlar parametrik funksiyalar
1.	MathCad dasturi ishchi oynasi menyu bo'limlari. MathCad dasturida funksiyalar masteri bilan ishlash.
2.	Diagrammalar, ikki o'lchovli va uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.
3.	Kimyo texnologiyasidagi sodda masalalarni yechimini topishda MathCad dasturdan foydalanish. MathCad dasturini kimyo texnologiyasida qo'llab obyekt haqidagi dastlabki vizual tasavvurni shakllantirish.
4.	Oddiy va murakkab matematik masalalarni MathCad dasturi yordamida yechishni o'rganish. O'zgaruvchilar belgilash, ulardan ishchi oynaning barcha joylarida foydalanish imkoniyati.
5.	Foydalanuvchi interfeysi, o'ziga xos trigonometrik, statistik va algebraik funksiyalar, uch o'lchamli grafik tasvirlarni olishni o'rganish va boshqalar.
6.	X-Y maydonini qutb grafigi, sirt maydoni, konturli grafik, 3D nuqtali grafik, 3D ustunli maydon va vektorlar maydoni.
7.	Dasturlash funksiyalari, boshqarish strukturasi, function, sum, find hamda boshqa operatorlar imkoniyatlaridan foydalanish va dasturlash.
8.	MathCad dasturi imkoniyatlaridan biri boshqa dasturlar bilan o'zaro bog'liqlikda ishlashi. MS Excel elektron jadvali, MS Word matn muxarriri va boshqa dasturlar.
9.	Maxsus funksiyalar, turlari va ularning imkoniyatlari. MathCad dasturining maxsus funksiyalari imkoniyatlaridan foydalanib masalalar va sodda differensial tenglamalarni yechish.
10.	Dasturda o'zgaruvchi, o'zgarmaslar va ularga qiymat berish. Dasturlash funksiyalari, boshqarish strukturasi shu jumladan for va while sikl operatori funksiyasi, if-operatori hamda shu kabi boshqa funksiya va operatorlardan foydalanish va dasturlash.

11.	Matlab dasturi ishchi oynasi, menyu bo'limlari. Matlab dasturida funksiyalar masteri bilan ishlash. Diagrammalar, ikki hamda uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.
12.	Kimyo injiniring masalalarini yechimini topishda MatLab dasturi imkoniyatlaridan yuqori darajada foydalanish.
13.	Vektorli yoki matrisalar shaklida saqlanadigan ma'lumot hamda element nomlanishi va boshqalar.
14.	Matlab dasturida elementar algebraik funksiyalar, standart funksiyalar va trigonometrik funksiyalardan foydalanish. MATLAB da matematik ifodalar ustida shakl almashtirishlar.
15.	Matlabda hisoblash ishlarini bajarish Matlab dasturida oddiy arifmetik masalalarni yechish
16.	Matlab dasturida funksiya hamda operator tushunchasi, turlari va ulardan foydalanish. MATLAB dasturida ikki va uch o'lchovli grafiklar qurish.
17.	Chizilgan grafiklar va grafik oynalarni loyihalash va boshqarishda grafik oyna menyu va uskunalar paneli elementlari hamda buyruqlar orqali amalga oshirish.
18.	MATLAB dasturining m-fayl imkoniyatlaridan foydalanib oziq-ovqat injiniring masalalarini yechish.
19.	Simulink® (simulyatsiya va Link) tushunchasi va uning vazifasi. MATLABda Simulink vazifasi imkoniyati va boshqalar.
20.	MATLAB menyu satrida «Simulink kutubxona»si bo'limi orqali yoki shunchaki yozib Simulinkni ishga tushirishish.
21.	Aralashtirish jarayonini avtomatlashtirishda MATLAB dasturidan Simulink paketini qo'llash.
22.	Aspen amaliy dasturi haqida tushuncha. Aspen ishchi oynasi, menyu bo'limlari.
23.	Funksiyalar masteri bilan ishlash va boshqa shu kabi imkoniyatlaridan foydalanish.

Talabalar bilimini baholash mezonlari

Kimyo injiniringida axborot texnologiyalari kursini semestrda o'qish va o'zlashtirish davomida olinadigan eng yuqori baho miqdori **“5”** baho.

Baholash semestr davomida talabaning o'quv ishlarni bajarish va o'zlashtirish darajasini aniqlash uchun joriy, oraliq nazoratga va yakuniy nazoratga ajratilgan.

Joriy va oraliq ballarni to'plash talabaning o'quv ishlarini o'z vaqtida va sifatli bajarish va topshirishi, shuningdek o'quv mashg'ulotlarida, ilmiy-metodik ishlarda va boshqa o'quv tadbirlarda faol qatnashishiga bog'liq.

Semestr mobaynida **joriy** nazorat kunlik baholab boriladi.

Semestr mobaynida 2 ta oraliq nazorat belgilangan. (haftasiga 2 soat o'qitiladigan fanlarda 1 ta oraliq).

Birinchi oraliq nazorat o'quv semestrning 9-xaftasida o'tkaziladi.

Ikkinchi oraliq nazorat esa o'quv semestrning 18-xaftasida o'tkaziladi.

Oraliq nazoratlar kurs dasturidagi bo'limlar tugallangandan keyin yozma ish yoki test tarzida o'tkaziladi.

Yozma ish tarkibida kurs dasturining asosiy mavzularini o'z ichiga olgan bo'lib, 3 yoki 4 ta savoldan iborat bo'ladi. Har bir savolning murakkabligi va hajmiga qarab 2,3,4,5 bahogacha berish nazarda tutilgan.

Test shaklida o'tkazilsa variant 30 ta savoldan iborat bo'lsa quyidagicha baholanadi:

27-30 "5" (a'lo)

21-26 "4" (yaxshi)

18-20 "3" (qoniqarli)

17 va undan kam bo'lsa "2" (qoniqarsiz)

ON turini topshirmagan va "2" (qoniqarsiz) baho olgan talaba YN ga kiritilmaydi va akademik qarzdor sanaladi. O'qish davrida dars qoldirgan talabalar mazkur darslar buyicha referat sifatida hisobot topshirgan holdagina oraliq va yakuniy nazoratlarga kiritiladi.

Yakuniy nazoratda joriy va oraliq nazoratda to'plangan baholar miqdori hisobga olinmaydi. Talaba sababli holatda ON va YaN turlariga ishtirok etmasa, ushbu talabaga tegishli nazorat turini qayta topshirishga fakultet dekani farmoyishi bilan ruxsat beriladi;

Semestr davomida kurs bo'yicha to'plagan ballar yig'indisi quyidagi o'zlashtirish ko'rsatkichi bo'yicha baholanadi:

Baho	Talabaning bilim darajasi
a'lo	talaba mustaqil qaror qabul qilsa, ijodiy fikrlasa, bilimini amalda qo'llay olsa va savol mohiyatini bilib, xulosa chiqarganligi uchun
yaxshi	talaba qisman mustaqil qaror qabul qilsa va ijodiy fikrlasa hamda bilimini amalda qo'llay olsa, berilgan savolning mohiyati haqida tushunchaga ega bo'lsa
qoniqarli	talaba bilimini amalda qisman qo'llay olsa, savol mohiyatini tasavvur qilsa
qoniqarsiz	Aniq tasavvurga ega emaslik, bilmaslik

*Yakuniy nazorat turi fakultet dekani, o'quv-uslubiy boshqarma bilan kelishgan holda va o'quv ishlari bo'yicha prorektor tomonidan imzolangandan so'ng tegishli fan bo'yicha o'quv mashg'ulotini olib borgan o'qituvchi ishtirokisiz o'tkaziladi.

Yakuniy nazorat shakli fan xususiyati va o'quv auditoriya soatidan kelib chiqib, kafedra belgilaydi.

Yakuniy nazorat

Test shaklida o'tkazilsa variant 30 ta savoldan iborat bo'lsa quyidagicha baholanadi:

27-30 "5"(a'lo)

21-26 “4”(yaxshi)

18-20 “3”(qoniqarli)

17 va undan kam bo'lsa “2”(qoniqarsiz)

“Yozma” shaklda o'tkazilsa savollar tarkibi 3 tadan iborat bo'ladi va har bir savol “5” baholi tizimda baholanib o'rtacha baho hisobga olinadi.

Yakuniy nazorat test shaklida o'tkazilsa O'qituvchi jurnalga bahoni shu kunning o'zida qayd etiladi. ON va YaN lar yozma ish shaklida bo'lsa, o'qituvchi 3 kun muddatda baholarni jurnalga qayd etadi.

“5”(a'lo), 4(yaxshi), 3(qoniqarli) baho olgan talabaga qayta topshirishga ruxsat berilmaydi;

Talaba sababsiz ravishda nazorat turiga qatnashmasa “0” belgisi qo'yiladi;

YaN turi bo'yicha baho jurnalga qayd etilganda, shu kuni talabaning baholash daftariga ham qayd etilishi shart;

YaN turi bo'yicha 2 (qoniqarsiz) baho yoki jurnalga “0” belgisi yozib qo'yilganda ushbu baho va belgi talabaning baholash daftariga yozilmaydi;

**O'ZBEKISTON RESPUBLIKASI
OLIIY VA O'RTA MAXSUS TA'LIM VAZIRLIGI**

**ISLOM KARIMOV NOMIDAGI
TOSHKENT DAVLAT TEXNIKA UNIVERSITETI
TERMIZ FILIALI**

**KIMYO INJINERIGIDA
AXBOROT TEXNOLOGIYALARI
o'quv fanidan**

Glossariy

Glossariy

Tavsif		
Ingliz tilida	Rus tilida	O'zbek tilida
Automated information system (AIS) is an assembly of computer hardware, software, firmware, or any combination of these, configured to accomplish specific information-handling operations, such as communication, computation, dissemination, processing, and storage of information.	Автоматизированная информационная система (АИС) - организационно-техническая система, использующая автоматизированные информационные технологии в целях информационно-аналитического обеспечения научно-инженерных работ и процессов управления.	Avtomatlashtirilgan axborot tizimi (AAT)-Tashkiliy texnik tizim bo'lib, avtomatlashtirilgan axborot texnologiyalarini boshqarish jarayonidava ilmiy – muhandislikislarida axborot taxlil ta'minoti maqsadida ishlatiladi.
Automated Information System (AIS) An AIS is a combination of computer hardware and computer software, data, and/or telecommunications that performs functions such as collecting, processing, storing, transmitting and displaying information.	Автоматизированная информационная технология (АИТ)-информационная технология, в которой для передачи, сбора, хранения и обработки данных используются методы и средства вычислительной техники и систем связи.	Avtomatlashtirilgan axborot texnologiyalari (AATex) – Axborot texnologiyasi bo'lib, ma'lumotlarni yig'ish, saqlash, uzatish, qayta ishlashda ishlatiladigan usullar va xisoblash texnikasi vositalari va aloqa tizimidan iborat bo'ladi.
Automated Training System - system, which include the complex of teaching and learning materials (demonstrations, theoretical, practical, control) and computer programs that control the learning process).	Автоматизированная обучающая система - система, включающая комплекс учебно-методических материалов (демонстрационных, теоретических, практических, контролирующих) и компьютерных программ, управляющих процессом обучения	Avtomatlashtirilgan o'qitish tizimi-bu tizim o'quv jarayonini boshqaradigan o'quv-uslubiy kompleks materiallaridan (namoyish, nazariy, amaliy, nazorat) va kompyuter dasturlaridan tashkil topgan.

Automated Data Bank (ADB) - a set of database management system and a specific database (database) data located (are) under its control.	Автоматизированный банк данных (АБД) - совокупность системы управления базами данных и конкретной базы (баз) данных, находящейся (находящихся) под ее управлением.	Avtomatlashtirilgan ma'lumotlar banki(AMB) – Ma'lumotlar bazalarini boshqarish tizimlarining yig'indisi va ularning boshqaruvi ostidagi konkret ma'lumotlar bazasidir.
Security Administrator is a person who manually administers user access rights to systems. A workflow system may call on a Security Administrator to fulfill an approved request on systems where automated administration agents are not available or have not yet been configured.	Администратор безопасности - лицо или группа лиц, ответственных за обеспечение безопасности системы, за реализацию и непрерывность соблюдения установленных административных мер защиты и осуществляющих постоянную организационную поддержку функционирования применяемых физических и технических средств защиты.	Havfsizlik adminstratori – tizim havfsizligini ta'minlashga javobgar, adminstratorlar tomonidan o'rnatilgan himoya tizimini uzluksiz ishlashini kuzatuvchi va ximoyani ta'minlovchi fizikaviy, texnik vositalarni doimiyishlashini tashkillashtiruvchi shaxs yoki shaxslar guruhidir.
WEB adress- Every computer connected to the internet has its unique web address, without which it cannot be reached by other computers. Also called universal resource locator or Uniform Resource Locator (URL).	Адрес страницы - данные, точно определяющие логический адрес сайта или Web-страницы в Internet.	Sahifa manzili – saytning aniq mantiqiy manzili yoki Internetdagi Web-sahifa.
Algorithm - is a set of instructions designed to perform a specific task.	Алгоритм - совокупность действий со строго определенными правилами выполнения.	Algoritm – bajarilish ketma-ketligi qat'iy qoidalarda aniqlangan amallar majmui.
Algoritmization-creating algorithm in order to solve tasks.	Алгоритмизация - составление алгоритмов для решения поставленных задач.	Algaritmlash – Berilgan masalani echish uchun algaritm tuzilishi.

A database is a collection of information that is organized so that it can easily be accessed, managed, and updated. In one view, databases can be classified according to types of content: bibliographic, full-text, numeric, and images. Byte-a unit of computer information that is equal to eight bits.	База данных - единая система данных, организованная по определенным правилам, которые предусматривают общие принципы описания, хранения и обработки данных.	Ma'lumotlar bazasi – umumiy prinsiplar asosida tavsiflanadigan, saqlanadigan va qayta ishlanadigan, aniq qoidalar asosida tashkil qilingan umumiy ma'lumotlarning yagona tizimi.
Information security, sometimes shortened to InfoSec, is the practice of defending information from unauthorized access, use, disclosure, disruption, modification, perusal, inspection, recording or destruction. It is a general term that can be used regardless of the form the data may take (e.g. electronic, physical).	Безопасность информации - состояние информации, информационных ресурсов и информационных систем, при котором с требуемой вероятностью обеспечивается защита информации от утечки, хищения, утраты и т. д.	Axborotni havfsizligi – axborot, axborot zaxiralari va axborot tizimlarida talab qilingan extimollikda axborot chiqib ketishidan, o'g'irlanishidan, yo'qotilishidan himoya ta'minlanadi.
Browser - a program with a graphical user interface for displaying HTML files, used to navigate the World Wide Web.	Браузер (Browser) - средство просмотра. Более полно: программное обеспечение, предоставляющее графический интерфейс для интерактивного поиска, обнаружения, просмотра и обработки данных в сети.	Browser - ko'rishvositasi. To'liq: tarmoqda ma'lumotlarni interaktiv qidirish, kashf etish, ko'rish va qayta ishlash uchun grafik interfeys taqdim etadigan dastur ta'minoti.
A web client is an application that communicates with	Веб-клиент - программа, позволяющая пользователю запрашивать документы с	Veb-klient – foydalanuvchiga veb-server hujjatlarni talab qilish

a web server, using Hypertext Transfer Protocol (HTTP)	веб-сервера.	imkonini beruvchi dastur.
A Web server is a program that uses HTTP (Hypertext Transfer Protocol) to serve the files that form Web pages to users, in response to their requests, which are forwarded by their computers' HTTP clients. Dedicated computers and appliances may be referred to as Web servers as well.	Веб-сервер - программа, запущенная на компьютере, предназначенная для предоставления документов другим компьютерам WWW, которые посылают соответствующие запросы.	Veb-server – shunday so'rovlarni jo'natuvchi boshqa kompyuterlar WWW xizmati uchun hujjatlarni taqdim etishga mo'ljallangan dastur.
Web page - a hypertext document connected to the World Wide Web.	Веб-страница - одиночный документ, содержащий гиперссылки, размещенный в WWW и определяемый с помощью адреса URL. Его можно открыть и просмотреть содержание с помощью программы просмотра - браузера.	Веб-sahifa – WWWga joylashtirilgan, URL adres yordamida aniqlanadigan, gipermurojatli alohida hujjat. Uni browser yordamida ko'rish va ochish mumkin. Bu multimedia hujjatlariga matn, grafika, tovush, video, animatsiya, gipermurojat va boshqalar kiradi.
Vector graphics is the use of polygons to represent images in computer graphics. Vectorgraphics are based on vectors, which lead through locations called control points or nodes.	Векторное изображение - это изображение, строящееся при помощи математического описания простых объектов - линий, окружностей, из которых создаются более сложные.	Vektorli tasvir – bu tasvir oddiy ob'ektlar: chiziqlar, doiralarni matematik ta'rifi yordamida barpo etadi va ular yordamida yanada murakkab tasvirlarni yaratadi.
White-board - an area on a display screen common to several users, on which they can write and draw.	Виртуальная аудиторная доска (белая доска) - электронная доска с возможностями непосредственного редактирования текста либо внесения соответствующих	Virtual auditoriya doskasi (oq doska) – matnni bevosita taxrirlash imkoniyatiga ega bo'lgan yoki berilgan matnga kerakli belgilarni kiritishi mumkin bo'lgan va axborotni

	пометок поверх исходного текста с передачей этой информации на расстояние.	masofaga uzatuvchi elektron doska.
A Virtual Library is a collection of resources available on one or more computer systems, where a single interface or entry point to the collections is provided. The key point being that the user need not know where particular resources are located -- the location is "virtual".	Виртуальная библиотека - учебно-методическая и дополнительная литература, размещенная в глобальной сети Интернет.	Virtual kutubxona – global Internet tarmog’iga joylashtirilgan uquv-uslubiy va qo’shimcha adabiyot.
A virtual reality - the computer-generated simulation of a three-dimensional image or environment that can be interacted with in a seemingly real or physical way by a person using special electronic equipment, such as a helmet with a screen inside or gloves fitted with sensors.	Виртуальная реальность - новая технология бесконтактного информационного взаимодействия, реализующая с помощью комплексных мультимедиа-операционных сред иллюзию непосредственного вхождения и присутствия в реальном времени в стереоскопически представленном «экранном мире». Более абстрактно - это мнимый мир, создаваемый в воображении пользователя.	Virtual haqiqat – muloqotsiz axborot hamkorlikning yangi texnologiyasi bo’lib, kompleks multimedia amaliyot vositalari yordamida real vaqt oralig’ida "dunyo ekrani"ga kirish ilyuzasini ta’minlaydi, Bu foydalanuvchi tasavvuridagi batafsil mavhum xayoliy dunyodir.
The virtual school differs from the traditional school through the physical medium that links administrators, teachers, and students.	Виртуальное учебное заведение - сообщество географически разделенных преподавателей и студентов, которые в процессе обучения общаются и взаимодействуют между собой с использованием электронных средств коммуникаций при	Virtual o’quv muassasasi – bir-biri haqida juda oz ma’lumotga ega bo’lgan, yoki umuman tanimagan, geografik jixatdan ajratilgan, o’qituvchi va talabalar jamiyati bo’lib ular elektron kommunikatsion vositalar yordamida o’quv jarayonida muloqot va xamkorlik

	минимальном или полностью отсутствующем личном, непосредственном контакте.	qiladilar.
Leased line is a private bidirectional or symmetric telecommunications line between two or more locations provided in exchange for a monthly rent. Sometimes known as a private circuit or data line in the UK. Hypermedia, a term derived from hypertext, extends the notion of the hypertext link to include links among any set of multimedia objects, including sound, motion video, and virtual reality. It can also connote a higher level of user/network interactivity than the interactivity already implicit in hypertext.	Выделенная линия - линия связи (канал передачи данных), установленная постоянно или на длительное время. Такой канал может называться также арендуемым, так как оборудование обычно принадлежит телекоммуникационным компаниям и сдается ими в аренду для исключительного пользования.	Ajratilgan liniya –doimiy yoki uzoq muddatga o'rnatilgan aloqa liniyasi (ma'lumotlar uzatish kanali). Telekommunikatsiya kompaniyalari tomonidan bunday kanallar ijaraga beriladi.
Hyperlink is a word, phrase, or image that you can click on to jump to a new document or a new section within the current document. Hyperlinks are found in nearly all Web pages, allowing users to click their way from page to page. Text hyperlinks are often blue and underlined,	Гиперссылка (Hyperlink) - элемент документа для связи между различными компонентами информации внутри самого документа, в других документах, в том числе и размещенных на различных компьютерах.	Gipermurojat – bir yoki turli kompyuterlarda joylashgan, hujjat ichidagi va tashqarisidagi turli axborot komponentalari orasidagi aloqa.

but don't have to be. When you move the cursor over a hyperlink, whether it is text or an image, the arrow should change to a small hand pointing at the link. When you click it, a new page or place in the current page will open.		
Hypertext - a software system that links topics on the screen to related information and graphics, which are typically accessed by a point-and-click method.	Гипертекст (Hypertext) - понятие, описывающее тип интерактивной среды с возможностями выполнения переходов по ссылкам. Ссылки (адреса форматаURL), внедренные в слова, фразы или рисунки, позволяют пользователю выбрать (установить указатель и нажать левую кнопку мыши) текст или рисунок и немедленно вывести связанные с ним сведения и материалы мультимедиа.	Gipertekst- интерактив muhit turini tasvirlash tushunchasi bo'lib, murojatlarga o'tish imkoniyatini bajaradi. So'zlar, iboralar yoki rasmlarga o'rnatilgan murojatlarni (URL manzil formati), foydalanuvchi (murojat ustida sichqonchaning chap tugmasini bosing) tanlashi matn yoki rasm va darhol tegishli ma'lumotlarni va multimedia materiallarlarini olib chiqish imkonini beradi.
Distance learning - a method of studying in which lectures are broadcast or classes are conducted by correspondence or over the Internet, without the student's needing to attend a school or college. Also called distance education.	Дистанционное обучение - обучение на расстоянии с использованием учебников, персональных компьютеров и сетей ЭВМ.	Masofaviy o'qitish – masofadan turib shaxsiy kompyuter va kompuyster tarmog'idan, darsliklardan foydalanib o'qitish.
Distance education - teaching system, which implements the method of distance learning educational	Дистанционное образование - педагогическая система, в которой реализуются способы дистанционного обучения с подтверждением	Masofaviy ta'lim- malaka oshirilganligini tasdiqlovchi pedagogik tizim bo'lib, unda masofaviy o'qitish usullari ishlatiladi.

qualification confirmation.	образовательного ценза.	
Document - a piece of written, printed, or electronic matter that provides information or evidence or that serves as an official record.	Документ - информация, зафиксированная на материальном носителе, имеющая реквизиты, позволяющие ее идентифицировать.	Xujjat – identifikatsion beruvchi imkonini ega bo'lgan rekvizitlarga axborot tashuvchiga yozilgan axborot.
Domain-a group of computers and devices on a network that are administered as a unit with common rules and procedures. Within the Internet, domains are defined by the IP address. All devices sharing a common part of the IP address are said to be in the same domain.	Домен (domain) - организационная единица в Интернете, служащая для идентификации узла или группы родственных узлов. Крупные домены могут подразделяться на поддомены, отражающие различные области интересов или ответственности.	Domen – Internetning tashkiliy birligi bo'lib, identifikatsion tugunlarga yoki qarindosh guruhlar tuguniga xizmat qiladi. Katta domenlar turli soxa qiziqishi yoki masulligini ifodalovchi domen o'rtiga bo'linadi.
Data protection - Data protection is the process of safeguarding important information from corruption and/or loss.	Защита информации - действия и средства по предотвращению утечки, хищения, искажения или подделки информации.	Axborot himoyasi – axborotni soxtalashtirish, buzish, o'g'rilash, noqonuniy tarqatishni bartaraf etish harakatlari va vositalari.
Internet - The global communication network that allows almost all computers worldwide to connect and exchange information. Some of the early impetus for such a network came from the U.S. government network Arpanet, starting in the 1960s.	Интернет (Internet) - открытая мировая информационная система, состоящая из взаимосвязанных компьютерных сетей, обеспечивающая доступ к удаленной информации и обмен информацией между компьютерами.	Internet – Kompyuterlar orasida axborot almashishini ta'minlaydigan, o'zaro bog'langan kompyuterlar tarmog'i bo'lib, ochiq jahon axborot tizimi tashkil qiladi.
An Internet service provider (ISP) is a company that provides	Интернет-провайдер (Internet Service Provider, ISP) - организация,	Internet provayder (Internet Service Provider, ISP) –

customers with Internet access. Data may be transmitted using several technologies, including dial-up, DSL, cable modem, wireless or dedicated high-speed interconnects.	предоставляющая пользователям доступ к Интернету.	foydalanuvchilarga Internetga kirishni ta'minlovchi tashkilot.
Information security, sometimes shortened to InfoSec, is the practice of defending information from unauthorized access, use, disclosure, disruption, modification, perusal, inspection, recording or destruction. It is a general term that can be used regardless of the form the data may take (e.g. electronic, physical).	Информационная безопасность - состояние защищенности информационной среды, обеспечивающее ее формирование, использование и развитие в интересах граждан, организаций, государства.	Axborot xavsizligi – axborot muxitini himoyalash, fuqorolar, korxonalar, davlatlar axborotini ximoyalangan xolatini taminlash, rivojlantirish va undan foydalanish.
Information technology training - IT Training is specific to the Information Technology (IT) industry, or to the skills necessary for performing information technology jobs. IT training includes courses related to the application, design, development, implementation, support or management of computer-based information systems.	Информационная технология - система научных и инженерных знаний, а также методов и средств, которая используется для создания, сбора, передачи, хранения и обработки информации в предметной области.	Axborot texnologiyalari – ilmiyva Muxandislik bilimlarini xamda usullari va vositalrini foydalanish va ularni ygish, uzatish ,saqlash va qayta ishlash axborot tizimi majmuasidir.
Case study is an account of an activity,	Кейс-технология - технология организации	Keys-texnologiyasi – o'quv jarayonida ishlatiladigan

event or problem that contains a real or hypothetical situation and includes the complexities you would encounter in the workplace. Case studies are used to help you see how the complexities of real life influence decisions.	учебного процесса, при которой учебно-методические материалы комплектуются в специальный набор (кейс) и передаются (пересылаются) студенту для самостоятельного изучения (с периодическими консультациями у назначенных ему преподавателей).	ixtiyoriy uquv uslubiy materiallar texnologiyasi va talabaga mustaqil ishlarni berish va qabul qilish to'plami.
Client-server architecture is common in both local and wide area networks. For example, if an office has a server that stores the company's database on it, the other computers in the office that can access the database are "clients" of the server.	Клиент (client) - программное обеспечение для доступа и получения данных при взаимодействии с программным обеспечением сервера, размещенного на другом компьютере.	Mijoz (client) – Kerak bo'lgan ma'lumot yoki resursga kirish uchun mijoz dastur ishga tushadi boshqa kompyuterga ulanadi.
CD - A compact disc [sometimes spelled disk] (CD) is a small, portable, round medium made of molded polymer (close in size to the floppy disk) for electronically recording, storing, and playing back audio, video, text, and other information in digital form.	Компакт-диск - оптический диск, используемый для постоянного хранения информации больших объемов	Kompakt-disk – optik tolali disk, ma'lumotlarni saqlash uchun.
Computer graphics are simply images displayed on a computer screen. Graphics are often contrasted with text, which is comprised of	Компьютерная графика - это создание, демонстрация и обработка графических изображений с помощью компьютера.	Kompyuter grafikasi – grafik ko'rinishdagi ma'lumotlarni kompyuter yordamida namoyish etish va qayta ishlash .

characters, such as numbers and letters, rather than images.		
Multimedia - is the field concerned with the computer-controlled integration of text, graphics, drawings, still and moving images (Video), animation, audio, and any other media where every type of information can be represented, stored, transmitted and processed digitally.	Мультимедиа (Multimedia) - компьютерные системы с интегрированной поддержкой звукозаписей и видеозаписей.	Multimedia (Multimedia) – tovushli va videoyozuvlarni kompyuter tizimi orqali bevosita boshqarish.

I.O'quv fanining dolzarbligi va oliy kasbiy ta'limdagi o'rni

Ushbu fan dasturida «Kimyo injineringda axborot texnologiyalar» faniga tegishli bo'lgan barcha mavzular bo'yicha talabalarga Davlat ta'lim standartlari asosida yetkazilishi shart bo'lgan minimum bilimlar va ko'nikmalarni to'la qamrab olingan. Unda «Kimyo injineringda axborot texnologiyalar» fanining o'qitilish ta'lim yo'nalishfariga mos holda rejalashtirilishi ko'zda tutilgan. Shuningdek ushbu fan II-kurslarda o'qitishi maqsadga muvofiq.

Kimyo sohasida axborot texnologiyalarining o'ziga xos xususiyatlari va imkoniyatlarini, hamda axborot texnologiyalarini inson faoliyatining turli sohalarida qo'llanilish asoslarini o'rgatishdan iboratdir. Bunda asosiy e'tibor axborot texnologiyalarini kimyo sohasida qo'llanilishiga qaratilgandir. Shuningdek talabalarni axborot texnologiyalarini texnik va dasturiy vositalari bilan tanishtirish va bu vositalarni ishlab chiqarish, ilmiy tadqiqot ishlari, hamda o'quv jarayonlariga tadbiq etish usullari va ularning o'ziga xos xususiyatlarini o'rgatishdan iboratdir.

II.O'quv fanining maqsadi va vazifasi

Fanni o'qitishdan maqsad - Kimyo injineringda axborot texnologiyalarining o'ziga xos xususiyatlari va imkoniyatlarini o'rganish, axborot texnologiyalarini kimyo texnologiyalarni turli sohalarida qo'llanilish asoslarini o'rganishdan iboratdir. Asosiy e'tibor axborot texnologiyalarini kimyo sanoat korxonalarida texnologik jarayonlarga qo'llanilishiga qaratilgandir. Jumladan axborotni saqlash, qayta ishlash va ilmiy dunyo qarashni shakllantirish, barcha ishlab chiqarish sohalarida hamda turli axborot tizimlarida kechayotgan axborot jarayonlarini roli va xarakteri to'g'risida umumiy tasavvur hosil qilishdir. Shuningdek zamonaviy dasturiy paketlar orqali eng oddiy texnologik jarayonlar muommolarini yechish haqida qisman ma'lumotga ega bo'lish.

Ayniqsa talabalarda fanining mohiyati, asosiy tamoyillari, vazifalari va yangi dasturiy ta'minot va tizimlar imkoniyatlari, dasturlash asoslari, algoritmik til tuzilmasi, funksiyalari va asosiy parametrlarini o'rganish, shuningdek biror algoritmik tilning barcha imkoniyatlari, obektga mo'ljallangan dasturlash tillari hamda dunyo tajribasida qo'llanilayotgan eng zamonaviy amaliy dasturiy paketlarini o'rgatish va ishlab chiqarish sohalariga mos dastlabki vizual qarashlarni hosil qilish.

«Kimyo injineringda axborot texnologiyalar» fanini o'zlashtirish jarayonida talaba:

- dasturlash va hisoblash texnikasi imkoniyatlaridan foydalanish ko'nikmalariga ega bo'lishi lozim;
- turli darajadagi dasturlash tillarini o'zlashtirish;
- dasturiy ta'minot va dasturlash texnologiyasini bilishi va ulardan foydalana olishi;
- ishlab chiqarish jarayonlarida axborot texnologiyalarini qo'llay bilish;
- amaliy dasturlar paketlarni bilishi va ulardan foydalana olishi;
- turli darajadagi dasturlash tillarini bilish;
- amaliy dasturiy paketlardan foydalanishni mukammal bilish;
- amaliy dasturlar paketlarni kimyo injineringga qo'llashda ularni tanlay bilish;
- amaliy dasturiy paketlarni kimyo injinering muammolari uchun qo'llash haqida ko'nikmalarga ega bo'lishi kerak.

III. Asosiy nazariy qism (ma'ruza mashg'ulotlari)

1-Modul. Kimyo injineriingda axborot texnologiyalar tushunchasi va mohiyati.

1-mavzu: Kimyo injineriingda axborot texnologiyalar faniga kirish.

Kimyo injineriingda axborot texnologiyalar tushunchasi va mohiyati. Uni vujudga kelishi va rivojlanishini asosiy bosqichlari. "Kimyo injineriingda axborot texnologiyalar" fani va uning bugungi kundagi ahamiyati. Mamlakatda kompyuterlashtirish, ishlab chiqarishda axborot texnologiyalarni keng qo'llash va axborot kommunikatsiya texnologiyalarini rivojlantirish ushbu fanni o'qitishdagi roli va ahamiyati. Talabalarni fanga bo'lgan qiziqishini oshirish va boshqa fanlar bilan uzviy bog'liqligini tushintira olish.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim, blits so'rov, ajurali arra, nilufar guli, menyu, klaster, munozara, o'z - o'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

2-modul. MS Excel dasturini kimyo injeneriingda qo'llash.

2-mavzu. MS Excel elektron jadvali xaqida umumiy tushuncha.

MS Excel elektron jadvalining ishchi oynasi, menyu bo'limlari. MS Excel elektron jadvali va listlarni boshqarish. Funktsiyalar masteri bilan ishlash. Diagrammalar, grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Pog'ona, qadamba - qadam, Venn diagrammasi, T-sxemasi, o'z - o'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

3-mavzu. MS Excel asoslari.

Funksiya masteri $f(x)$ dan foydalanib formulalar kiritish va undan foydalanish. Shuningdek trigonometrik funksiyalar (SIN, COS, TAN, ACOS, va boshqalar), statistik funksiyalar (MEAN (o'rta), AVERAGE (o'rtacha), t-test (t-sinov()), arifmetik funksiyalar (SUM, ROUND, LOG, LOG10), mantiqiy, (IF (balki) FALSE (hato) va boshqalar) va boshqalar bilan ishlash. VBA (Visual Basic For Applications) imkoniyatidan foydalanish va boshqalar.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Blits so'rov, zig-zag usuli, munozara, BBB, Insert, o'z - o'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

4-mavzu. MS Excel elektron jadvali yordamida misollar yechish.

MS Excelda Solver imkoniyatlarini o'rganish. Kimyo texnologiyasi masalalarini yechishda MS Excelda Solver imkoniyatlarini o'rganish. Kimyo texnologiyasi muommolarini MS Excel electron jadvalining Solver bo'limi yordamida optimallashtirish.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. klaster, munozara, Venn diagrammasi.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

3-modul. C++ algoritmik tili.

5-mavzu. Algoritm lash asoslari. Algoritm turlari va xossalari.

Kompyuterda masalalarni yechish bosqichlari. Algoritm tushunchasi va uning xossalari. Algoritm lar turlari. Chiziqli hisoblash jarayonlarini algoritm lash. Masalalarni algoritm lash va dasturlashtirish asoslari. Algoritm larga qo'yiladigan talablar.

Algoritmnlarni ta'riflash usullari. Dasturiy ta'minotning tarkibi va amaliy dasturlar paketi.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. BBB, munozara, Venn diagrammasi, T-sxema, o'z - o'zini boshqarish.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

6-mavzu. C++ algoritmik tili. Kirish.

Dasturlash. Kompyuter anatomiyasi, mashina kodi va dasturlash tillari, dasturlash muhiti bilan tanishish, ilk dasturni tahlil qilish, xatolar, muammoni yechish: dizayn algoritmi.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Ajurali arra, bumerang, 3x3 usuli, munozara, o'z - o'zini boshqarish.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

7-mavzu. Asosiy ma'lumot turlari.

O'zgaruvchilar, o'zgarmlar, arifmetik amallar, kiritish va natija, muammo yechimi va boshqalar.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Blits, 4x4 usuli, munozara, o'z - o'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

8,9-mavzu. Operatorlar va standart funktsiyalar.

If operatori, sonlarni va qatorlarni solishtirish, birdan ortiq tanlovlar, muammo yechimi: diagrammalar, biriktirilgan o'tishlar, muammo yechimi: kiruvchi ma'lumotni tasdiqlash, operatorlar va o'zgaruvchilar.

Funksiyalarni joriy qilish, parametrlarni o'tkazish, qaytish qiymatlari, qaytish qiymatlarisiz funktsiyalar, muammo yechimi: qayta ishlatiladigan funktsiyalar, muammo yechimi: bosqichma bosqich detallashtirish, o'zgaruvchni aniqlash maydoni va global o'zgaruvchilar, izohlar parametri, funktsiyalar

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Klaster, Blits - so'rov, Fikrlash xaritasi, Veer.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

3-modul. Mathcad va Matlab dasturlarini oziq-ovqat injeneringda qo'llash.

10-mavzu. MathCad dasturi haqida umumiy tushuncha.

MathCad dasturi ishchi oynasi, menyu bo'limlari. MathCad dasturida funktsiyalar masteri bilan ishlash. Diagrammalar, ikki o'lchovli va uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Ma'ruza, namoyish etish, Baliq skeleti, guruhlar bilan ishlash.. Adabiyotlar:

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

11-mavzu. MatCad dasturidan foydalanib masalalar va sodda differensial tenglamalarni yechish.

Kimyo texnologiyasidagi sodda masalalarni yechimini topishda MathCad dasturdan foydalanish. MathCad dasturini kimyo texnologiyasida qo'llab obyekt haqidagi dastlabki vizual tasavvurni shakllantirish. Oddiy va murakkab matematik masalalarni MathCad dasturi yordamida yechishni o'rganish. O'zgaruvchilar belgilash, ulardan ishchi oynaning barcha joylarida foydalanish imkoniyati. Foydalanuvchi interfeysi, o'ziga xos

trigonometrik, statistik va algebraik funksiyalar, uch o'lchamli grafik tasvirlarni olishni o'rganish va boshqalar.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Bino, blits, ajurali arra, nilufar guli, menyu, algoritm, munozara, o'z - o'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

12-mavzu. Tasvir va grafik yaratish.

X-Y maydonini qutb grafigi, sirt maydoni, konturli grafik, 3D nuqtali grafik, 3D ustunli maydon va vektorlar maydoni. Dasturlash funksiyalari, boshqarish strukturasi, function, sum, find hamda boshqa operatorlar imkoniyatlaridan foydalanish va dasturlash.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Pog'ona, qadamba - qadam, Venn diagrammasi, T- sxemasi, o'z - o'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

13-mavzu. MathCadda dasturlash.

Dasturda o'zgaruvchi, o'zgarmaslar va ularga qiymat berish. Dasturlash funksiyalari, boshqarish strukturasi shu jumladan for va while sikl operatori funksiyasi, if- operatori hamda shu kabi boshqa funktsiya va operatorlardan foydalanish va dasturlash.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Pog'ona, qadamba - qadam, Venn diagrammasi, T- sxemasi, o'z - o'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

14-mavzu. MathCadning boshqa dasturiy ta'minotlar bilan o'zaro bog'liqligi.

MathCad dasturi imkoniyatlaridan biri boshqa dasturlar bilan o'zaro bog'liqlikda ishlashi. Misol uchun MS Excel elektron jadvali, MS Word matn muxarriri va boshqa dasturlar. Bu imkoniyatni yaratib beruvchi buyruqlar, ularning vazifalari va boshqalar.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Blits so'rov, zig - zag usuli, munozara, BBB, Insert, o'z-o'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

15-mavzu. Matlab dasturi haqida umumiy tushuncha.

Matlab dasturi ishchi oynasi, menyu bo'limlari. Matlab dasturida funksiyalar masteri bilan ishlash. Diagrammalar, ikki xamda uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish. Kimyo injinerining masalalarini echimini topishda MatLab dasturi imkoniyatlaridan yuqori darajada foydalanish. Vektorli yoki matrisalar shaklda saqlanadigan ma'lumot hamda element nomlanishi va boshqalar.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Ajurali arra, bumerang, 3x3 usuli, munozara, o'z - o'zini boshqarish.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

16-mavzu. Matlab dasturida funktsiya va grafik.

Matlab dasturida funktsiya hamda operator tushunchisi, turlari va ulardan foydalanish. MATLAB dasturida ikki va uch o'lchovli grafiklar qurish. Chizilgan grafiklar va grafik oynalarni loyihalash va boshqarishda grafik oyna menyu va uskunalar paneli elementlari hamda buyruqlar orqali amalga oshirish.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Klaster, Blits - so'rov, Fikrlash xaritasi.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

17-mavzu. Aspen amaliy dasturi haqida tushuncha.

Aspen amaliy dasturi haqida tushuncha. Aspen ishchi oynasi, menyu bo'limlari. Funktsiyalar masteri bilan ishlash va boshqa shu kabi imkoniyatlaridan foydalanish.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Blits so'rov, zig-zag usuli, munozara, BBB, Insert, o'z - o 'zini nazorat.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

18-mavzu. MathCad, Matlab va Aspen amaliy dasturiy paketlarini kimyo injinerigida qo'llanilishi.

Matlab dasturida funksiya hamda operator tushunchisi, turlari va ulardan foydalanish. MATLAB dasturida ikki va uch o'lchovli grafiklar qurish. Chizilgan grafiklar va grafik oynalarni loyihalash va boshqarishda grafik oyna menyu va uskunalar paneli elementlari hamda buyruqlar orqali amalga oshirish. Aspen ishchi oynasi, menyu bo'limlari. Funktsiyalar masteri bilan ishlash va boshqa shu kabi imkoniyatlaridan foydalanish.

Qo'llaniladigan ta'lim texnologiyalari: dialogik yondoshuv, muammoli ta'lim. Klaster, Blits - so 'rov, Fikrlash xaritasi.

Adabiyotlar: A1,A2,A3,Q4,Q5,Q6,Q7

Ma'ruza mashg'ulotlari

1 - jadval

№	Ma'ruza mavzular(barcha)	Ajratilgan soat
3-semestr		
1.	Kimyo injinerigida axborot texnologiyalar faniga kirish.	2
2.	MS Excel elektron jadvali xaqida umumiy tushuncha	2
3.	MS Excel asoslari	2
4.	MS Excel elektron jadvali yordamida misollar yechish.	2
5.	Algoritmash asoslari. Algoritm turlari va xossalari.	2
6.	C++ algoritmik tili. Kirish.	2
7.	Asosiy ma'lumot turlari.	2
8.	Operatorlar	2
9.	Standart funksiyalar	2
Jami: – 18 soat		
4 - semestr		
10.	MathCad dasturi haqida umumiy tushuncha	2
11.	MathCad dasturi haqida umumiy tushuncha	2
12.	Mathcad dasturidan foydalanib masalalar va sodda differensial tenglamalarni yechish	2
13.	Mathcad dasturidan foydalanib masalalar va sodda differensial tenglamalarni yechish	2
14.	Tasvir va grafik yaratish.	2
15.	Tasvir va grafik yaratish	2
16.	MathCadda dasturlash.	2

17.	MathCadda dasturlash.	2
18.	Mathcadning boshqa dasturiy ta'minotlar bilan o'zaro bog'liqligi.	2
19.	Mathcadning boshqa dasturiy ta'minotlar bilan o'zaro bog'liqligi.	2
20.	Matlab dasturi haqida umumiy tushuncha.	2
21.	Matlab dasturi haqida umumiy tushuncha.	2
22.	Matlab dasturida funksiya va grafik.	2
23.	Matlab dasturida funksiya va grafik.	2
24.	Aspen amaliy dasturi haqida tushuncha	2
25.	Aspen amaliy dasturi haqida tushuncha	2
26.	Mathcad , Matlab va aspen amaliy dasturiy paketlarini kimyo injinerligida qo'llanilishi.	2
27.	Mathcad , Matlab va aspen amaliy dasturiy paketlarini kimyo injinerligida qo'llanilishi.	2
		Jami:36 soat

Umumiy jami: – 54 soat

Izoh:Mavzular bo'yicha ma'ruza rejalarini tuzishda asosiy qismdagi mazmunini to'liq qamrab olinishi shart.

Masalan: Mavzu: MathCad dasturi haqida umumiy tushuncha

Reja:

- 1.MathCad dasturi ishchi oynasi, menyu bo'limlari.
- 2.MathCad dasturida funksiyalar masteri bilan ishlash.
- 3.Diagrammalar, ikki o'lchovli va uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.

IV. Amaliy mashg'ulotlar bo'yicha ko'rsatma va tavsiyalar.

Amaliy mashg'ulotlardan maqsad amaliyotda uchraydigan sodda masalalarni yechish ko'nikmalarini hosil qilish. Shuningdek, amaliy mashg'ulotlarda talabalar muayyan masala bo'yicha mavjud bo'lgan yoki mustaqil tarzda kichik ishchi guruhlar yordamida hosil qilingan amaliy ishlarni muvofiq qiladilar va ma'ruzalar mavzularini talabalar mustaqil o'rganib, masalalarni mustaqil yechishga tayyorlanish, mavzuni tahlil qilib, fikrlash qobiliyatlarini oshirishadi.

«Kimyo injinerida axborot texnologiyalar» fanining paydo bo'lishi."Kimyo injinerida axborot texnologiyalar" faniga kirish. Algoritm asoslari. Algoritm turlari va xossalari. Ularning xarakteristikallari va xususiyatlari. Zamonaviy ob'ektga mo'ljallangan algoritmik tillar. Chiziqli hisoblash jarayonlarini algoritmash. C++ algoritmik tili va unda dastur yaratish. Dasturlash muhiti bilan tanishish, ilk dasturni tahlil qilish, xatolar, muammoni yechish.Asosiy ma'lumot turi. Dasturlash tillari imkoniyatlaridan foydalanib Kimyo injineridagi masalalarni dasturini dasturlash va ularni yechimini topishda qo'llash.

MS Excel dastur imkoniyatlaridan foydalanib sodda Kimyo injiner masalalarini yechishga qo'llash. MathCad va MatLab dasturlari imkoniyatlaridan foydalanib kimyo injiner masalalarini yechish.

Kimyo injineridagi mavjud sodda masalalarni optimallashtirishda MS Excel elektron jadvalining Solver bo'limidan foydalanish.

MatLab va MathCad amaliy dasturiy paketlarini vazifalari va ularning imkoniyatlarini kimyo injineri muammolarini yechishga qo'llash. Yuqoridagi dasturlarda ishlash va ularning imkoniyatlaridan foydalanib Kimyo va boshqa sohalar

sanoatidagi mavjud masalalarni yechish.

№	Amaliy mashg'ulotlari mavzulari	Dars soatlari hajmi
3-semestr		
1.	“Kimyo injinering uchun axborot texnologiyalari” faniga kirish.	4
2.	Algoritmash asoslari.	4
3.	Algoritm turlari va xossalari.Ularning xarakteristikallari va xususiyatlari.	4
4.	Zamonaviy ob’ektga mo’ljallangan algoritmik tillar.	4
5	Chiziqli hisoblash jarayonlarini algoritmash.	4
6	C++ algoritmik tili va unda dastur yaratish.	4
7	Dasturlash muhiti bilan tanishish, ilk dasturni taxlil qilish, xatolar, muammoni yechish.	4
8	Asosiy ma’lumot turlari.	4
9.	Dasturlash tillari imkoniyatlaridan foydalanib kimyo injineringidagi masalalarni dasturini va ularni yechimini topishda qo’llash.	4
4-semestr		36
10	MS Excel elektron jadval haqida umumiy tushuncha.	4
11	MS Excel dastur imkoniyatlaridan foydalanib sodda kimyoviy injinering masalarini yechishga qo’llash.	4
12	Kimyo uchun Mathcad muhiti.	4
13	Mathcadda simvolli hisoblash.	4
14	Tasvir va grafiklar, dasturlash funksiyalari.	4
15	Mathcad va Matlab dasturlari imkoniyatlaridan foydalanib kimyo injinering masalarini yechish.	4
16	Kimyo injineringidagi mavjud sodda masalalarni optimallashtirishda MS Excel elektron jadvalining Solver bo’limidan foydalanish.	4
17	Matlab va MathCad amaliy dasturiy paketlarini vazifalari va ularning imkoniyatlarini kimyo injinering muammolarini yechishga qo’llash.	4
18	Yuqoridagi dasturlarda ishlash va ularning imkoniyatlaridan foydalanib kimyo va boshqa sohalar sanoatidagi mavjud masalalarni yechish.	4
4-semestr		36
		Jami: 36 soat

Umumiy jami: 72 soat

V. Laboratoriya mashg'ulotlar bo'yicha ko'rsatma va tavsiyalar.

Tajriba ishlarida talabalar berilgan topshiriqlar yuzasidan kompyuter yordamida grafik ma’lumotlarni hosil qilishadi va amalda uning natijalarini ko’rib, ularni tahlil qiladilar va xulosalar chiqaradilar.

“Kimyo injineringda axborot texnologiyalar” fanining paydo bo’lishi. « Kimyo injineringda axborot texnologiyalar» faniga kirish. Algoritmash asoslari. Algoritm turlari va xossalari. Chiziqli hisoblash jarayonlarini algoritmash. C++ algoritmik tili va unda

dastur yaratish. Dasturlash muhiti bilan tanishish, ilk dasturni tahlil qilish, xatolar, muammoni yechish. Asosiy ma'lumot turlari. Dasturlash tillari imkoniyatlaridan foydalanib oziq-ovqat injiniringidagi masalalarini yechimini laboratoriya ishlari yordamida bajarish.

MS Excel dastur imkoniyatlaridan foydalanib sodda kimyo injinerida masalalarini yechishga qo'llash. MathCad va MatLab dasturlari imkoniyatlaridan foydalanib kimyo injineri muammolarga qo'llab laboratoriya ishlarini bajarish. Laboratoriya ishlari davomida kimyo injineringdagi mavjud sodda masalalarni optimallashtirishda MS Excel elektron jadvalining Solver bo'limidan foydalanish.

MatLab va MathCad amaliy dasturiy paketlarini vazifalari va ularning imkoniyatlarini kimyo injineri muammolarini yechishga qo'llash. Yuqoridagi dasturlarda ishlash va ularning imkoniyatlaridan foydalanib kimyo va boshqa sohalaridagi mavjud masalalarni laboratoriya ishlari orqali amalga oshirish.

Laboratoriya mashg'ulotlarining ro'yxati

№	Laboratoriya mashg'ulotlari mavzulari	Dars soatlari hajmi
3-semestr		
1.	Asosiy ma'lumot turlari.	2
2.	Algoritmash asoslari.	2
3.	Tarmoqlanuvchi va takrorlanuvchi hisoblash jarayonini tashkil qilish.	2
4.	MS Excel elektron jadval haqida umumiy tushuncha	2
5.	MS Excel dasturi yordamida kimyo injineringida muammoli vaziyatni yechish.	2
6.	MS Excel elektron jadvali imkoniyatlaridan foydalanib masalalar yechish.	2
7.	C++ dasturlash tili va uning asosiy operatorlari.	2
8.	C++ algoritmik tilning tarmoqlanish operatorlari.	2
9.	C++ algoritmik tilning takrorlanish operatorlar	2
10.	C++ algoritmik tilning bir o'lchamli va ko'p o'lchamli massivlari.	2
11.	C++ algoritmik tilida funksiyalarning grafigini qurish.	2
12.	Kimyo injinering uchun Mathcad (m.c.) muxiti	2
13.	Mathcadda va elektron jadvallarda tenglamalar yechish.	2
14.	Mathcadda simvolli hisoblash turlari orqali hisoblash.	2
15.	Tasvir va grafiklar.	2
16.	Dasturlash funksiyalari.	2
17.	Chiziqli tenglamalar tizimlarini yechish.	2
18.	Mathcad dasturidan foydalanib masalalar tenglamalarni yechish.	2
		Jami:36 soat
4-semestr		
19.	Mathcad dasturidan foydalanib masalalar tenglamalarni yechish.	2
20.	Mathcad dasturidan foydalanib sodda differensial tenglamalarni yechish.	2

21.	Mathcad dasturidan foydalanib sodda differensial tenglamalarni yechish.	2
22.	Matlab dasturi haqida umumiy tushuncha.	2
23.	Matlab dasturidagi ishchi oynada amallar bajarish.	2
24.	Matlab dasturidagi ishchi oynada amallar bajarish.	2
25.	Sonlar ketma-ketligini matlab vositalari yordamida yaratish.	2
26.	Matritsalar ustida amallar.	2
27.	Maxsus matritsalar.	2
28.	Matlab dasturida funksiyalar bilan ishlash.	2
29.	Matlab dasturida ikki va uch o'lchamli grafik qurish.	2
30.	Kimyo injinerining masalalarini yechish uchun Matlab dasturi.	2
31.	Simulink.	2
32.	Simulink qism dasturidagi tayyor bloklar bilan ishlash.	2
33.	Simulink qism dasturidagi tayyor bloklar bilan ishlash.	2
34.	Simulink paketini qo'llash.	2
35.	Aspen amaliy dasturiy paketining interfeysida ishlash	2
36.	Aspen amaliy dasturiy paketining interfeysida ishlash	2

Jami:36 soat

Umumiy jami: – 72 soat

Kurs ishini tashkil etish bo'yicha uslubiy ko'rsatmalar

Chiziqli hisoblash jarayonlarini algoritmizatsiya qilish.

C++ algoritmik tili va unda dastur yaratish.

Dasturlash muhiti bilan tanishish, ilk dasturni taxlil qilish, xatolar, muammoni yechish.

Asosiy ma'lumot turlari.

Dasturlash tillari imkoniyatlaridan foydalanib kimyo injineridagi masalalarini yechishda C++ algoritmik tili yordamida bajarish.

Quyidagi differensial tenglamalarning Eyler usuli orqali, boshlang'ich qiymatlari asosida hisoblash jarayonini algoritmik til- C++da dasturlang.

№	f(x,y) berilgan funksiya	y⁰ boshlang'ich qiymat
1	$X^3 \sin y + 1$	0.0
2	$X^2 \sin y - 1$	0.1
3	$e^x + 3y$	2.0
4	$y^2 + x^3$	0.3
5	$Y^3 + x^2$	0.4
6	$1/(1+y^2) + x^2$	0.0
7	$1/(1+y^2) + xy$	0.1
8	$\cos y + xy$	0.2
9	$X^2 \cos y + 0.1$	0.3
10	$X^3 \cos y + 0.1$	0.4
11	$\cos(xy) - 0.5$	0.5
12	$e^{-y} + e^x - 2$	0.0
13	$e^{-y} - e^x - 0.1$	0.5
14	$e^{-xy} + 1$	0.4
15	$Y^2 + x^4$	0.3
16	$XY^2 + x^{(1/2)}$	0.2
17	$\cos y + x \cdot \tan y$	0,2

18	$X^2 \cos y + \sin x$	0.3
19	$X^3 \cos y + \cos x$	0.4
20	$\cos(y) - \ln(x)$	0.5

VI. Mustaqil ta'lim va mustaqil ishlar

Mustaqil ishni bajarishdan asosiy maqsad — professor-o'qituvchilarning bevosita rahbarligi va nazorati ostida talabalarni semestr davomida fanlarni uzluksiz o'rganishini tashkil etish, olingan bilim, ko'nikmalarini chuqur o'rganib mustahkamlash, eng sodda kimyo injiniringda masalalarini yechish hamda tahlil qilish shuningdek o'z mutaxassisliklari haqida yanada tasavvurlarini boyitish, yangi bilimlarni mustaqil ravishda izlab topish va qabul qilishini shakllanirishdan iborat.

Talaba mustaqil ishini tashkil etishda "Kimyo injiniringda axborot texnologiyalar" fanining xususiyatlarini, shuningdek, har bir talabaning akademik o'zlashtirish darajasi va qobiliyatini hisobga olgan holda quyidagi shakllardan foydalaniladi: ayrim nazariy mavzularni turli xil o'quv adabiyotlari yordamida mustaqil o'zlashtirish, berilgan mavzu bo'yicha referat tayyorlash, laboratoriya va amaliy mashg'ulotlarga tayyorgarlik ko'rish, kurs ishini bajarish, bitiruv malakaviy ishini tayyorlash, anjumanlarga ma'ruza tezislarni tayyorlash, fan olimpiadalarida ishtirok etish, ilmiy jamiyatlar, to'garaklarda, kafedralarning ilmiy ishlarida ishtirok etish va hokazolar.

Tavsiya etilayotgan mustaqil ishlarning mavzulari

1. Masalaning algoritmini yaratish.
2. Qo'yilgan masalaning dasturni C++ algoritmik tilida quyida keltirilgan mavzular asosida yaratish.
3. Qo'yilgan masalaning MS Excel dastur imkoniyatlaridan foydalanib yechish.
4. MathCad haqida umumiy tushunchalari va ulardan foydalanish
5. MathCad dasturini kimyo injiniringda qo'llash
6. Qo'yilgan masalaning MathCad dastur imkoniyatlaridan foydalanib yechish.
7. MatLab haqida umumiy tushunchalari va ulardan foydalanish
8. MatLab dasturini kimyo injiniringda qo'llash
9. Qo'yilgan masalaning Matlab dastur imkoniyatlaridan foydalanib yechish.

«Kimyo injiniringda axborot texnologiyalar» fanidan (mustaqil ta'lim) mavzulari va ularning soatlar bo'yicha taqsimoti

№	Mustaqil ta'lim mavzulari	Berilgan topshiriqlar	Bajarish muddati	Hajmi (soatda)
3-semestr				
1.	Axborot texnologiyalari tushunchasi va mohiyati. Uni vujudga kelishi va rivojlanishini asosiy bosqichlari	Referat tayyorlash	1-hafta	4
2.	Kompyuterda masalalarni yechish bosqichlari. Algoritm tushunchasi va uning xossalari	Individual topshiriqlarni bajarish	2-hafta	4
3.	Masalaning algoritmini yaratish.	Individual topshiriqlarni bajarish	3-hafta	4

4.	Algoritmlar turlari. Chiziqli hisoblash jarayonlarini algoritmlash.	O'quv loyihani bajarish	4-hafta	4
5.	Tarmoqlanuvchi algoritmlar tuzish	Referat tayyorlash	4-hafta	4
6.	Takrorlanuvchi algoritmlar tuzish.	Individual topshiriqlarni bajarish	5-hafta	4
7.	Masalalarni algoritmlash va dasturlashtirish asoslari. Algoritmlarga qo'yiladigan talablar.	Referat tayyorlash	5-hafta	4
8.	Algoritmlarni ta'riflash usullari. Dasturiy ta'minotning tarkibi va amaliy dasturlar paketi.	Individual topshiriqlarni bajarish	6-hafta	4
9.	Mamlakatda kompyuterlashtirish, ishlab chiqarishda axborot texnologiyalarni keng qo'llash va axborot kommunikatsiya texnologiyalarini rivojlantirish ushbu fanni o'qitishdagi roli va ahamiyati.	Individual topshiriqlarni bajarish	7-hafta	4
10	Dasturlash. Kompyuter anatomiyasi, mashina kodi va dasturlash tillari, dasturlash muhiti bilan tanishish, ilk dasturni tahlil qilish, xatolar, muammoni yechish, dizayn algoritmi.	Individual topshiriqlarni bajarish	8-hafta	4
11	MS Excel elektron jadvalining ishchi oynasi, menyu bo'limlari.	Individual topshiriqlarni bajarish	9-hafta	4
12	MS Excel elektron jadvali va listlarni boshqarish.	Referat tayyorlash	10-hafta	4
13	Funksiyalar masteri bilan ishlash. Diagrammalar, grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.	Referat tayyorlash	11-hafta	4
14	Funktsiya masteri fx dan foydalanib formulalar kiritish va undan foydalanish. Shuningdek trigonometric funksiyalar (SIN, COS, TAN, ACOS, va boshqalar), Statistik funksiyalari (MEAN (o'rta), AVERAGE (o'rtacha)).	Referat tayyorlash	12-hafta	4
15	Trigonometrik funksiyalar t-test (t-sinovi) arifmetik funksiyalar (SUM, ROUND, LOG, LOG10) mantiqiy, (IF (balki) FALSE (hato) va boshqalar) va boshqalar bilan ishlash. VBA (Visual Basic For Applications)	Referat tayyorlash	13-hafta	4
16	MS Excelda Solver imkoniyatlarini o'rganish, kimyo texnologiyasi masalalarini yechishda MS Excelda Solver imkoniyatlarini o'rganish.	Individual topshiriqlarni bajarish	14-hafta	4

17	C++ algoritmik tilida masala dasturini tuzish	Taqdimot tayyorlash	15-hafta	4
18	O'zgaruvchilar, o'zgaruvchilar, arifmetik amallar, kiritish va natija, muammo yechimi va boshqalar.	Individual topshiriqlarni bajarish	16-hafta	4
19	if operatori, sonlarni va qatorlarni solishtirish, birdan ortiq tanlovlar, muammo yechimi, operatorlar va o'zgaruvchilar	Individual topshiriqlarni bajarish	16-hafta	4
20	Funksiyalarni joriy qilish, parametrni o'tkazish, qaytish qiymatlari, qaytish qiymatlarisiz funksiyalar, qayta ishlatiladigan flinksiyalar, aniqlash maydoni va global o'zgaruvchilar, izohlar parametrik funksiyalar	Referat tayyorlash	17-hafta	4
Jami: 80 soat				
4-semestr				
1.	MathCad dasturi ishchi oynasi menyu bo'limlari. MathCad dasturida funksiyalar masteri bilan ishlash.	Individual topshiriqlarni bajarish	21-hafta	4
2.	Diagrammalar, ikki o'lchovli va uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.	Referat tayyorlash	21-hafta	4
3.	Kimyo texnologiyasidagi sodda masalalarni yechimini topishda MathCad dasturdan foydalanish. MathCad dasturini kimyo texnologiyasida qo'llab obyekt haqidagi dastlabki vezual tasavvurni shakllantirish.	Individual topshiriqlarni bajarish	22-hafta	4
4.	Oddiy va murakkab matematik masalalarni MathCad dasturi yordamida yechishni o'rganish. O'zgaruvchilar belgilash, ulardan ishchi oynaning barcha joylarida foydalanish imkoniyati.	Individual topshiriqlarni bajarish	22-hafta	4
5.	Foydalanuvchi interfeysi, o'ziga xos trigonometrik, statistik va algebraik funksiyalar, uch o'lchamli grafik tasvirlarni olishni o'rganish va boshqalar.	Individual topshiriqlarni bajarish	23-hafta	4
6.	X-Y maydonini qutb grafigi, sirt maydoni, konturli grafik, 3D nuqtali grafik, 3D ustunli maydon va vektorlar maydoni.	Referat tayyorlash	23-hafta	4
7.	Dasturlash funksiyalari, boshqarish strukturasi, function, sum, find hamda boshqa operatorlar imkoniyatlaridan foydalanish va dasturlash.	Individual topshiriqlarni bajarish	24-hafta	4

8.	MathCad dasturi imkoniyatlaridan biri boshqa dasturlar bilan o'zaro bog'liqlikda ishlashi. MS Excel elektron jadvali, MS Word matn muxarriri va boshqa dasturlar.	Individual topshiriqlarni bajarish	24-hafta	4
9.	Maxsus funksiyalar, turlari va ularning imkoniyatlari. MathCad dasturining maxsus funksiyalari imkoniyatlaridan foydalanib masalalar va sodda differensial tenglamalarni yechish.	Referat tayyorlash	25-hafta	4
10.	Dasturda o'zgaruvchi, o'zgarmaslar va ularga qiymat berish. Dasturlash funksiyalari, boshqarish strukturasi shu jumladan for va while sikl operatori funksiyasi, if-operatori hamda shu kabi boshqa funksiya va operatorlardan foydalanish va dasturlash.	Individual topshiriqlarni bajarish	25-hafta	4
11.	Matlab dasturi ishchi oynasi, menyu bo'limlari. Matlab dasturida funksiyalar masteri bilan ishlash. Diagrammalar, ikki hamda uch o'lchovli grafiklar tuzish va boshqa shu kabi imkoniyatlaridan foydalanish.	Referat tayyorlash	26-hafta	4
12.	Kimyo injinering masalalarini yechimini topishda MatLab dasturi imkoniyatlaridan yuqori darajada foydalanish.	Individual topshiriqlarni bajarish	26-hafta	4
13.	Vektorli yoki matrisalar shaklida saqlanadigan ma'lumot hamda element nomlanishi va boshqalar.	Individual topshiriqlarni bajarish	27-hafta	4
14.	Matlab dasturida elementar algebraik funksiyalar, standart funksiyalar va trigonometrik funksiyalardan foydalanish. MATLAB da matematik ifodalar ustida shakl almashtirishlar.	Referat tayyorlash	27-hafta	4
15.	Matlabda hisoblash ishlarini bajarish Matlab dasturida oddiy arifmetik masalalarni yechish	Individual topshiriqlarni bajarish	28-hafta	4
16.	Matlab dasturida funksiya hamda operator tushunchasi, turlari va ulardan foydalanish. MATLAB dasturida ikki va uch o'lchovli grafiklar qurish.	Individual topshiriqlarni bajarish	28-hafta	4
17.	Chizilgan grafiklar va grafik oynalarni loyihalash va boshqarishda grafik oyna menyu va uskunalar paneli elementlari hamda buyruqlar orqali amalga oshirish.	Individual topshiriqlarni bajarish	29-hafta	4
18.	MATLAB dasturining m-fayl imkoniyatlaridan foydalanib oziq-ovqat injiniring masalalarini yechish.	Referat tayyorlash	30-hafta	4

19.	Simulink® (simulyatsiya va Link) tushunchasi va uning vazifasi. MATLABda Simulink vazifasi imkoniyati va boshqalar.	Individual topshiriqlarni bajarish	31-hafta	4
20.	MATLAB menyu satrida «Simulink kutubxona»si bo'limi orqali yoki shunchaki yozib Simulinkni ishga tushirishish.	Referat tayyorlash	31-hafta	4
21.	Aralashtirish jarayonini avtomatlashtirishda MATLAB dasturidan Simulink paketini qo'llash.	Individual topshiriqlarni bajarish	32-hafta	4
22.	Aspen amaliy dasturi haqida tushuncha. Aspen ishchi oynasi, menyu bo'limlari.	Individual topshiriqlarni bajarish	33-hafta	4
23.	Funksiyalar masteri bilan ishlash va boshqa shu kabi imkoniyatlaridan foydalanish.	Individual topshiriqlarni bajarish	34-hafta	4
Jami: 92 soat				

Umumiy jami: 172 soat

VII. Talabalar bilimni baholash mezonlari

Kimyo injinerida axborot texnologiyalari kursini semestrda o'qish va o'zlashtirish davomida olinadigan eng yuqori baho miqdori “5” baho.

Baholash semestr davomida talabani o'quv ishlarni bajarish va o'zlashtirish darajasini aniqlash uchun joriy, oraliq nazoratga va yakuniy nazoratga ajratilgan.

Joriy va oraliq ballarni to'plash talabani o'quv ishlarini o'z vaqtida va sifatli bajarish va topshirishi, shuningdek o'quv mashg'ulotlarida, ilmiy-metodik ishlarda va boshqa o'quv tadbirlarda faol qatnashishiga bog'liq.

Semestr mobaynida **joriy** nazorat kunlik baholab boriladi.

Semestr mobaynida 2 ta oraliq nazorat belgilangan. (haftasiga 2 soat o'qitiladigan fanlarda 1 ta oraliq).

Birinchi oraliq nazorat o'quv semestrning 9-haftasida o'tkaziladi.

Ikkinchi oraliq nazorat esa o'quv semestrning 18-haftasida o'tkaziladi.

Oraliq nazoratlar kurs dasturidagi bo'limlar tugallangandan keyin yozma ish yoki test tarzida o'tkaziladi.

Yozma ish tarkibida kurs dasturining asosiy mavzularini o'z ichiga olgan bo'lib, 3 yoki 4 ta savoldan iborat bo'ladi. Har bir savolning murakkabligi va hajmiga qarab 2,3,4,5 bahogacha berish nazarda tutilgan.

Test shaklida o'tkazilsa variant 30 ta savoldan iborat bo'lsa quyidagicha baholanadi:

27-30 “5” (a'lo)

21-26 “4” (yaxshi)

18-20 “3” (qoniqarli)

17 va undan kam bo'lsa “2” (qoniqarsiz)

ON turini topshirmagan va “2” (qoniqarsiz) baho olgan talaba YN ga kiritilmaydi va akademik qarzidor sanaladi. O'qish davrida dars qoldirgan talabalar mazkur darslar buyicha referat sifatida hisobot topshirgan holdagina oraliq va yakuniy nazoratlarga kiritiladi.

Yakuniy nazoratda joriy va oraliq nazoratda to'plangan baholar miqdori hisobga olinmaydi. Talaba sababli holatda ON va YaN turlariga ishtirok etmasa, ushbu talabaga

tegishli nazorat turini qayta topshirishga fakultet dekani farmoyishi bilan ruxsat beriladi;

Semestr davomida kurs bo'yicha to'plagan ballar yig'indisi quyidagi o'zlashtirish ko'rsatkichi bo'yicha baholanadi:

Baho	Talabaning bilim darajasi
a'lo	talaba mustaqil qaror qabul qilsa, ijodiy fikrlasa, bilimini amalda qo'llay olsa va savol mohiyatini bilib, xulosa chiqarganligi uchun
yaxshi	talaba qisman mustaqil qaror qabul qilsa va ijodiy fikrlasa hamda bilimini amalda qo'llay olsa, berilgan savolning mohiyati haqida tushunchaga ega bo'lsa
qoniqarli	talaba bilimini amalda qisman qo'llay olsa, savol mohiyatini tasavvur qilsa
qoniqarsiz	Aniq tasavvurga ega emaslik, bilmaslik

*Yakuniy nazorat turi fakultet dekani, o'quv-uslubiy boshqarma bilan kelishgan holda va o'quv ishlari bo'yicha prorektor tomonidan imzolangandan so'ng tegishli fan bo'yicha o'quv mashg'ulotini olib borgan o'qituvchi ishtirokisiz o'tkaziladi.

Yakuniy nazorat shakli fan xususiyati va o'quv auditoriya soatidan kelib chiqib, kafedra belgilaydi.

Yakuniy nazorat

Test shaklida o'tkazilsa variant 30 ta savoldan iborat bo'lsa quyidagicha baholanadi:

27-30 "5"(a'lo)

21-26 "4"(yaxshi)

18-20 "3"(qoniqarli)

17 va undan kam bo'lsa "2"(qoniqarsiz)

"Yozma" shaklda o'tkazilsa savollar tarkibi 3 tadan iborat bo'ladi va har bir savol "5" baholi tizimda baholanib o'rtacha baho hisobga olinadi.

Yakuniy nazorat test shaklida o'tkazilsa O'qituvchi jurnalga bahoni shu kunning o'zida qayd etiladi. ON va YaN lar yozma ish shaklida bo'lsa, o'qituvchi 3 kun muddatda baholarni jurnalga qayd etadi.

"5"(a'lo), 4(yaxshi), 3(qoniqarli) baho olgan talabaga qayta topshirishga ruxsat berilmaydi;

Talaba sababsiz ravishda nazorat turiga qatnashmasa "0" belgisi qo'yiladi;

YaN turi bo'yicha baho jurnalga qayd etilganda, shu kuni talabaning baholash daftariga ham qayd etilishi shart;

YaN turi bo'yicha 2 (qoniqarsiz) baho yoki jurnalga "0" belgisi yozib qo'yilganda ushbu baho va belgi talabaning baholash daftariga yozilmaydi;

Jurnal o'z vaqtida to'g'ri va to'liq yuritilishi uchun, baholarning asossiz o'zgartirishlar kiritilmasligi uchun dekan va fan bo'yicha o'qituvchi mas'uldir.

VIII. Asosiy va qo'shimcha o'quv adabiyotlar hamda axborot manbalari

Asosiy adabiyotlar

1. R. Hamdamov, U. Begimqulov, N. Tayloqov. Ta'limda axborot texnologiyalari. "O'zbekiston Milliy Ensiklopediyasi" Davlat ilmiy nashriyoti. Toshkent - 2010
2. S.S. G'ulomov, B.A. Begalov. Informatika va axborot texnologiyalari. Fan, T: 2010. 686 b.
3. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
4. M.M. Aripov, T. Imomov va boshqalar Informatika. Axborot texnologiyalari. O'quv qo'llanma, 1-2-qism. -Toshkent. 2005, TDTU 334 b. va 394 b.
5. A.M. Polatov. Algoritmlar va C++ tilida dasturlash asoslari. O'quv qo'llanma. –Toshkent "Universitet" 2017. 58 b
6. Axborot kommunikatsiya texnologiyalari. O'quv qo'llanma. –Toshkent. 2012
7. Informatika va axborot texnologiyalari. O'quv qo'llanma. – Toshkent 2012

Qo'shimcha adabiyotlar

1. Mirziyoyev SH.M. Qonun ustvorligi va inson manfaatlarini ta'minlash-yurt taraqqiyoti va xalq farovonligining garovi. 48b. T.O'zbekiston. 2017 yil
2. Mirziyoyev SH.M. Erkin va farovon demokratik O'zbekiston davlatini birgalikda barpo etamiz. 56b, T.O'zbekiston. 2016 yil
3. Mirziyoyev SH.M. Buyuk kelajagimizni mard va olijanob xalqimiz bilan birga quramiz. 488b, T.O'zbekiston. 2017 yil
4. Aripov M.A., Maraximov A.R., Informatika, informatik texnologiyalar. Darslik., T: TDYUI., 2004 yil
5. S. S. G'ulomov, B.A. Begaliyev "Informatika va axborot texnologiyalari" Toshkent, FAN., 2010 yil.

Internet saytlari

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>
3. www.intuit.ru
4. <http://virtual-university-eurasia.org>
5. <http://www.opennet.ru>
6. www.linux.org.ru
7. <https://nashol.com>
8. https://www.youtube.com/video/urok_Project_

**O'ZBEKISTON RESPUBLIKASI
OLIIY VA O'RTA MAXSUS TA'LIM VAZIRLIGI**

**ISLOM KARIMOV NOMIDAGI
TOSHKENT DAVLAT TEXNIKA UNIVERSITETI
TERMIZ FILIALI**

**KIMYO INJINERININGIDA
AXBOROT TEXNOLOGIYALARI
o'quv fanidan
TESTLAR**

TALABANING FAN BO'YICHA O'ZLASHTIRISH KO'RSATKICHINI BELGILASH UCHUN NAZORAT SAVOLLARI

1. Yuqori darajadagi dasturlash tillari nima?
2. Kompilyatsiya jarayonini necha bosqichdan tashkil topadi?
3. Qanday belgilar bilan tugagan barcha belgilar ketma-ketligi izoh hisoblanadi?
4. C++ tilining kalit so'zlariga qaysilar kiradi?
5. Protsessor registrlarini belgilash uchun qaysi so'zlar ishlatiladi?
6. Kompilyatsiya jarayoni nima?
7. Kiritilgan qiymatni o'zgaruvchi turiga mos kelishini qanday tekshirish mumkin?
8. Identifikator nima?
9. Sintaksis – qanday tilning qoidalarini?
10. C++ instruksiyalari qanday belgi bilan tugallanishi zarur?
11. Tilning ma'nosini beruvchi qoidalar to'plami qanday nomlanadi?
12. *int* turidagi son, odatda, nima uchun ishlatiladi?
13. Agar dasturda sintaktik xatolar bo'lsa, kompilyator bu haqida xabar beradimi?
14. Arifmetik operatorlar bajarilish ketma-ketligi qoidasi qanday qoida?
15. Figurali qavslar nima uchun ishlatiladi?
16. Vergul (","), odatda, nima uchun ishlatiladi?
17. "*double*" kalit so'zi nima uchun ishlatiladi?
18. Haqiqiy son turi nima?
19. Berilganlarning strukturalashgan turlari qanday?
20. Butun son turlari nima?
21. Suzuvchi nuqtali turlar nima?
22. Sanab o'tiluvchi tur nima?
23. Haqiqiy sonlar qanday kalit so'z bilan e'lon qilinadi?
24. O'nlik fiksirlangan nuqtali format deganda nima tushuniladi?
25. Eksponensial shaklda haqiqiy o'zgarmas nechta qismdan iborat bo'ladi va ularga misol keltiring.
26. Kompilyator nimaga qarab unga mos turni belgilaydi?
27. Opaliqni teng ikkiga bo'lish usuli asosida $f(x)=0$ tenglamani yeching?
28. $[a,b]$ oraliqda aniqlangan va uzliksiz $f(x)$ funksiya xosil qiluvchi uzani to'g'ri to'rtburchak usulida hisoblash dasturi yarating.
29. Qaysi operator yordamida oshkor ravishda bir turni boshqa turga keltirish mumkin?
30. C++ tilida tuzilgan dasturning asosiy maqsadi nima?
31. O'zgaruvchi nima?
32. O'zgaruvchilarga ifoda qanday belgi orqali yuklanadi?
33. C++ tilida ***num = num + 2;*** ko'rinishidagi ifoda nimani bildiradi?
34. Kod qismidagi o'zgaruvchilarning kompilyator uchun qanday ketmaketlikda qiymat olishlari jadvalini yozing.
35. C++ tilida bir turni boshqa turga keltirishning qanday yo'llari mavjud?

36. Inkrement va dekrement amallari nima?
37. “Prefiks” yoki “postfiks” amal tushunchasi qanday ifodalarda o‘rinli?
38. C++ tilida ixtiyoriy (tayanch va hosilaviy) turdagi o‘zgaruvchilarning o‘lchami qanday amal yordamida aniqlanadi?
39. Qo‘yilgan masalani yechishda biror holat ro‘y bergan yoki yo‘qligini ifodalash uchun 0 va 1 qiymat qabul qiluvchi nimalardan foydalaniladi?
40. C++ tilida bayt razryadlari ustida mantiqiy amallar majmuasi jadvalini ko‘rsating.
41. [a,b] oraliqda aniqlangan va uzliksiz $f(x)$ funksiya xosil qiluvchi uzani trapetsia usulida hisoblash dasturi yarating.
42. Baytdagi bitlar qiymatini chapga yoki o‘ngga surish uchun mos ravishda qaysi amallari qo‘llaniladi?
43. Taqqoslash amali qanday amal va u qanday ko‘rinishga ega?
44. Taqqoslash amallarining natijasi - taqqoslash o‘rinli bo‘lsa yoki o‘rinli bo‘lmasa qanday qiymat bo‘ladi?
45. Ifodalar qiymatini hisoblashda nima hisobga olinadi?
46. O‘qish oqimi nima?
47. Baytlar-bu... .
48. Input stream nima?
49. Output stream nima?
50. C++ dastur sarlavhasida qaysi fayldan foydalanish kerak?
51. Qiymat dastur orqali o‘qib olinganida nimalar qiymatlarning ajratuvchisi sifatida qabul qilinadi?
52. Qaysi kalit so‘z orqali kiritish oqimidagi bir nechta funksiyalardan foydalanish mumkin?
53. Berilganlar oqimidan faqat kerakli qismini kiritish kerak bo‘lsa, unda kiritish oqimining qaysi funksiyasidan foydalanish kerak?
54. *fixed* manipulyatori nimani chop etadi?
55. Mantiqiy qo‘shish operatori nechta ifoda orqali hisoblanadi?
56. Mantiqiy inkor operatori tekshirilayotgan ifoda yolg‘on bo‘lsa qanday qiymat qaytaradi?
57. *if* operatori nima?
58. C++ tilining qurilmalari operatorlarni blok ko‘rinishida tashkil qilishga imkon beradimi? Buni tushuntirib bering.
59. *Blok* nima?
60. Shart operatorida e‘lon qilish operatorlarini ishlatish mumkinmi?
61. <operator1> va <operator2> shartli operator bo‘lishi mumkinmi?
62. Agar tekshirilayotgan shart nisbatan sodda bo‘lsa nima ishlatish mumkin?
63. *switch* tarmoqlanish operatori nima?
64. *break* va *default* kalit so‘zlari nima uchun ishlatiladi?
65. *switch* operatorida e‘lon operatorlari ham uchrashi mumkinmi?
66. *switch* operatori bajarilishida “*sakrab o‘tish*” holatlari bo‘lishi hisobiga blok ichidagi ayrim e‘lonlar bajarilmasligi va buning oqibatida dastur ishida xatolik ro‘y berishi mumkinmi?
67. *switch* operatori nima uchun ishlatiladi?

68. Sanab o'tiluvchi turlar va shu turdagi o'zgaruvchilarga misol keltiring.
69. Mantiqiy operatorlarga nimalar kiradi?
70. <operand1> <taqqoslash amali> <operand2> - quyidagi amal nimani anglatadi?
71. "&&" "||" "!" operatorlari nimani anglatadi?
72. (x==3) && (y==5) agar x va y qiymatlari har xil bo'lsa qanday qiymat qaytaradi?
73. Mantiqiy ko'paytirish operatori qanday belgi orqali belgilanadi?
74. Mantiqiy qo'shish operatori qanday belgi orqali belgilanadi?
75. Beshta sonning o'rta arifmetigi qanday topiladi?
76. Cheksiz takrorlash uchun takrorlashni davom ettirish sharti qanday?
77. Takrorlash operatorida ham bloklardan foydalanish mumkinmi?
78. C++ tilining qurilmalari operatorlarni blok ko'rinishida tashkil qilishga imkon beradimi?
79. Agar <ifoda> *rost* qiymatli o'zgarmas ifoda bo'lsa, takrorlash qanday bo'ladi?
80. Takrorlash operatorlarining bajarilishida qanday holatlar yuzaga kelishi mumkin?
81. Takrorlash operatori ichma-ich joylashgan bo'lishi mumkinmi?
82. *do-while* takrorlash operatori qanday vazifani bajaradi?
83. *while* takrorlash shartini oldindan tekshiruvchi takrorlash operatori hisoblanadimi?
84. *else* operatori qanday vazifani bajaradi?
85. *continue* operatori qanday vazifani bajaradi?
86. *break* operatori qanday vazifani bajaradi?
87. *while* operatori qanday vazifani bajaradi?
88. *for* operatori qanday vazifani bajaradi?
89. *if* operatori qanday vazifani bajaradi?
90. *include* operatori qanday vazifani bajaradi?
91. Ayrim hollarda, *goto* operatorining «sakrab o'tishi» hisobiga xatoliklar yuzaga kelishi mumkinmi?
92. Takrorlash operatorida qavs ichidagi ifodalar bo'lmasligi mumkin, lekin sintaksisida ';' bo'lmasligiga ruxsat beriladimi?
93. Massiv deb nimaga aytiladi?
94. Massiv indeksi sifatida qanday son ishlatiladi?
95. Dasturda ishlatiladigan har bir konkret massiv qanday nomga ega?
96. Massiv elementiga murojaat qilish qanday amalga oshiriladi?
97. C++ tilida massivlar elementining turiga cheklovlar qo'yiladimi?
98. Ikki o'lchamli massivning sintaksisi qanday ko'rinishda bo'ladi?
99. *cout* operatori qanday vazifani bajaradi?
100. *cin* operatori qanday vazifani bajaradi?
101. *for* operatori qanday vazifani bajaradi?
102. *while* operatori qanday vazifani bajaradi?
103. *do-while* operatori qanday vazifani bajaradi?
104. *char* operatori qanday vazifani bajaradi?

105. *iostream* operatori qanday vazifani bajaradi?
106. Massiv – bu?
107. Palindrom deb nimaga aytiladi?
108. Misolda massiv elementlar soni keltirilmagan bo'lsa, massiv elementlar soni qanday aniklanadi?
109. *int* operatori qanday vazifani bajaradi?
110. *gets* operatori qanday vazifani bajaradi?
111. Funksiya bu...?
112. Funksiyalar modullar deb ham atalishi mumkunmi?
113. C++ tilida funksiya chaqirilganda ayrim argumentlarni tushirib qoldirish mumkunmi va bunga qanday erishish mumkun?
114. Lokal o'zgaruvchilar o'zlari e'lon qilingan funksiya yoki blok chegarasida ko'rinish sohasiga ega bo'ladimi?
115. Funksiyalar qanday turlarga bo'linadi?
116. Kelishuv bo'yicha qiymat berishning nechta sharti bor?
117. C++ tilidagi har qanday dasturda qaysi funksiya bosh funksiya bolishi kerak?
118. Funksiyalar qanday ko'rinishda bo'ladi?
119. Funksiya qanday aniklanadi?
120. Kompilyator ishlashi natijasida har bir funksiya qanday ko'rinishda bo'ladi?
121. Lokal o'zgaruvchi yashash vaqti qanday aniklanadi?
122. *include* operatori qanday funksiyani bajaradi?
123. *cout* operatori qanday funksiyani bajaradi?
124. *cin* operatori qanday funksiyani bajaradi?
125. *if* operatori qanday funksiyani bajaradi?
126. *else* operatori qanday funksiyani bajaradi?
127. Ayrim algoritmlar berilganlarning qanday turdagi qiymatlari uchun qo'llanishi mumkin?
128. Qayta yuklanuvchi funksiyalardan foydalanishda qanday qoidalarga rioya qilish kerak?
129. *inline* operatori qanday funksiyani bajaradi
130. *float* operatori qanday funksiyani bajaradi
131. Qiymatni hisoblash uchun funksiyaning «oldingi qiymati» ma'lum bo'lishi kerakmi?
132. Matematikada manfiy bo'lmagan butun sonlarning faktorialini aniqlash qaysi formula yordamida amalga oshiriladi?
133. Rekursiya deb nimaga aytiladi?
134. Rekursiya uchun qanday aniqlanishlar o'rinli?
135. Agar faktorial funksiyasiga $n > 0$ qiymat berilsa, qanday holat ro'y beradi?
136. Rekursiv funksiyalarning to'g'ri amal qilishi uchun qanday chaqirishlarning to'xtash sharti bo'lishi kerak?
137. Har bir rekursiv murojaat qo'shimcha xotira talab qiladimi?
138. Rekursiya chiroyli, ixcham ko'ringani bilan xotirani tejash va hisoblash vaqtini qisqartirish nuqtai nazaridan uni imkon qadar interaktiv hisoblash bilan almashtirilgani ma'qulmi?

139. Rekursiya qanday to'xtatiladi?
140. Har bir rekursiv formula nechta ifodaga ega bo'lishi kerak?
141. Tadqiq qilinadigan obyekt haqida tushuncha bering.
142. Jarayonni hal qilishdagi muammolar nimadan iborat?
143. Matematik model ta'rifini bering.
144. Diskret model nimadan iborat?
145. Masalani yechish usullari to'g'risida ma'lumot bering.
146. Dastur yaratish texnologiyalarini aytib bering.
147. Dasturni sozlash nimadan iborat?
148. Natija va uning tahlili nimadan iborat?
149. Hisoblash eksperimenti haqida tushuncha bering.
150. "Algoritm" tushunchasining ta'rifini bering.
151. Blok-sxemaning ta'rifini bering.
152. Blok-sxema asosiy figuralar ma'nosini aytib bering.
153. Qanday algoritmlar chiziqli deb ataladi.
154. Qanday algoritmlar tarmoqlanuvchi deb ataladi?
155. Qanday algoritmlar takrorlanuvchi deb ataladi?
156. Blok-sxemada ishlatiladigan asosiy bloklarni aytib bering.
157. Dastur tuzilishida qaysi bloklardan foydalaniladi?
158. Algoritmning xatolar qanday qilib to'g'rilanadi?
159. Chiziqli algoritmning ishlash prinsiplari.
160. Shartli algoritmning ishlash prinsiplari.

Test savollari

1-variant:

1. Axborot hajmining eng kichik o'lchov birligi nima?

- A) Bit V) Son S) Belgi D) Bayt

2. Algoritm so'zi qaysi olimning nomi bilan bog'liq?

- A) Beruniy V) Al-Xorazmiy S) Ulug'bek D) Ibn Sino

3. Kompyuter nima?

- A) Chop qilish qurilmasi B) Faqat yozishga mo'ljallangan qurilma C) Faqat o'qish uchun mo'ljallangan qurilma
D) Ma'lumotlar ustida turli amallar bajaruvchi kichik hajmdagi elektron hisoblash mashinasi.

4. Informatika fani nimani o'rgatadi?

- A) Axborotlarni o'lchashning usulini o'rgatadi B) Axborotlarni uzatish usullarini o'rgatadi
C) Axborotni qayta ishlash usullarini o'rgatadi.
D. Axborotlarni qabul qilish, saqlash, qayta ishlash va uzatish usullarini o'rgatadi.

5. Qanday chop qilish qurilmalari bor?

- A) Matritsali, purkagichli, lazerli. B) Ignali, lazerli C) Lazerli,
D) Purkagichli, lazerli

6. Algoritm so'zi qaysi olim nomidan olingan.

A) Al-Farg'oniy B) A. Navoiy C) Al Xorazmiy D) Pifagor

7. Windowsning ish stolida nimalar mavjud?

A) Yorliqlar, fayllar B) Timsollar, buyruqlar C) Yorliqlar, timsollar, Korzina, "Moy kompyuter", masalalar paneli D) Masalalar paneli

8. "Alt" + "F4" klaviatura tugmachalari birgalikda bosilganda qanday amal bajariladi?

A) dastur yuklanadi. B) dasturdan chiqiladi. C) Fayldan chiqiladi. D) Yangi fayl ko'riladi.

9. Qatorlar orasidagi intervallarni belgilash *Format* menyusining qaysi bandi yordamida bajariladi?

A) SHrift. V) Abzats. S) Tabulyatsiya. D) Stil.

10. Kiritilayotgan ma'lumotlarni kompyuter xotirasiga olish uchun qaysi tugmachalardan foydalaniladi?

A) F1. V) F4. S) Shift + F12. D) F5.

2- variant:

1. Saxifalab pastga yurish tugmasi qaysi?

A) PageUp. V) Home. S) End. D) Page Down

2. Xat boshiga joy tashlash uchun qaysi tugmadan foydalaniladi?

A) End V) Esc S) Delete D) Tab.

3. Format buyruqlar to'plami nima vazifani bajaradi?

A) matn, formula va grafiklarni formatlovchi buyruqlar to'plami;
B) hisoblash jarayonini boshqaruvchi buyruqlar to'plami;
C) xujjatga turli ob'ektlarni joylashtirish uchun xizmat qiluvchi buyruqlar to'plami
D) matn, formula va grafiklarni formatlovchi buyruqlar to'plami;

4. DELETE tugmasining vazifasi nima?

A) Matn boshiga borish B) Abzats boshiga borish C) Ko'rsatkichdan oldingi (chap tomondagi) belgini o'chirish
D) Ko'rsatkich turgan belgini va undan o'ng tomondagi belgilarni o'chirish.

5. End tugmasining vazifasi ?

A) Sahifalab quyiga yurish B) Xat boshiga borish C) Sahifalab yuqoriga yurish D) qator oxiriga borish

6. Qator boshiga borish uchun qaysi yozuvli tugmacha kerak bo'ladi?

A) Nome V) Yend S) Page Up D) Esc

7. "Pusk" tugmasining vazifasi nimadan iborat?

A) OS ni ishga tushirishdan.
B) Windows ning barcha dasturlarini ishga tushirishdan.
C) Word ni ishga tushirishdan. D) Excel ni ishga tushirishdan.

8. WORD taxrirlagichi ishga tushganda, kompyuterning o'zi faylga qanday nom beradi?

A) PAPKA 1 V) Dokument 1 S) tekst1 D) WORD 1

9. Matnda kursor qanday ko'rinishda bo'ladi?

A) Ko'rsatkich B) Gorizontal chiziqcha C) Vertikal chiziqcha
D) qum soat

10. Kursorning joyini o'zgartirish uchun nimadan foydalaniladi?

- A) Sichqoncha. B) Asboblar paneli C) Menyu satri
D) Holat satri

3- variant:

1. Menyu satridan so'ng qanday satr joylashgan?

- A) Asboblar paneli B) Sarlavha satri C) Holat satri
D) CHizg'ich

2. CHizg'ichlar qaysi menyu orqali o'rnatiladi va o'chiriladi?

- A) Vid va Pravka B) Pravka va Vid C) Vid va lineyka D) Fayl va Vid

3. Asboblar panelidagi birinchi tugma qanday tugma?

- A) Saqlash tugmasi B) CHop etish tugmasi C) Yangi fayl hosil qilish tugmasi. D) SHrift turi

4. Qanday dasturlash tillarini bilasiz?

- A) Word, Excel, Paint B) Delphi , Paskal, C++ C) Windows, NC
D) Matlab, Mathcad, Avtacad

5. Windows operatsion tizimining asosiy menyusiga qanday chiqiladi?

- A) «Moy kompyuter » papkasidan; B) «Pusk» tugmachasidan;
C) «Moi dokumenty» papkasidan; D) Masalalar panelidan

6. Skanerning vazifasi nima?

- A) qog'ozdagi ma'lumotlarni kompyuterga tasviriy ko'chirish.
B) Qog'ozdagi matnlarni chop qilish.
C) Diskdagi fayllarni kompyuterga o'tkazish.
D) Ma'lumotlarni magnit lentalariga yozish.

7. Qaysi printerda tasvir nuqtalardan zarbli usul bilan shakllanadi?

- A) Matrisali printer B) purkagichli printer C) lazerli printer

8. Qanday chop qilish qurilmalari bor?

- A) Rangli, planshetli, purkagichli
B) Matrisali printer , purkagichli printer , lazerli printer
C) oq-qora, rangli
D) matrisali

9. Birinchi avlod EHMLarning asosiy element bazasini nimalar tashkil etadi.

- A) Elektron lampalar B) yarim o'tkazgichli tranzistorlar
C) chiplar D) integral sxemalar

10. Monitorlar necha xil ko'rinishda bo'ladi?

- A) 5 B) 3 C) 4 D) 2

4- variant:

1. Printerning vazifasi nima?

- A) Diskda fayllar bilan ishlash. B) Kataloglar bilan ishlash.
C) Ma'lumotlarni ekranda aks ettirish. D) Komp'yuterdagi ma'lumotlarni qog'ozga ko'chiruvchi qurilma.

2. Modem - bu ... uchun mo'ljallangan qurilma

- A) axborotni chop etish
B) axborotni saqlash

- C) axborotni shu vaqtda qayta ishlash
D) axborotni telefon kanallari orqali uzatish

3. Lampali sxemalarga asoslangan birinchi raqamli xisoblash mashinalari qachon, qayerda ishlab chiqildi?

- A) Fransiya 1945 y.
B) AQSH 1946-1948 y.y.
C) Kiyek 1951 g.
D) Angliya 1930-1940 y.y.

4. Printerlar necha turga bo'linadi?

- A) 5 B) 3 C) 4 D) 2

5. Yacheyka deganda nimani tushunasiz?

- A) maydon B) qator C) ustun D) qator va ustun kesishmasi

6. Excel da diagrammalarni tanlash qanday amalga oshiriladi?

- A) Fayl. V) Tablitsa. S) Pravka. D) Vstavka.

7. MS Excel ishga tushganda avtomatik ravishda uning ishchi hujjati qanday nomlanadi?

- A) Dokument 1 B) Kniga 1 C) Untitled 1 D) Prezentsatsiya 1

8. Microsoft Excel jadvallarida rasmda keltirilgan holatda Enter tugmasi bosilsa B2 katagida qanday qiymat hosil bo'ladi

	A	B	C
1			
2		$=5+6*2$	
3			
4			
5			

- A) 17 B) 5 C) 12 D) 16

9. Tizimli qism odatda qanday shaklda ishlab chiqiladi?

- A) uzun va monitorli.
B) gorizontal va vertikal.
C) tezkor va doimiy.
D) hamma javob to'g'ri.

10. Kompyuterlarning qaysi xotirasida berilganlar avvaldan joylashtirilgan bo'ladi?

- A) registr xotira B) doimiy xotira C) videoxotira D) tezkor xotira

5- variant:

1. Ma'lumotlarni kiritadigan va kompyuter bilan muloqat qilish vazifasini o'taydigan qurilmalar?

- A) klaviatura, sichqoncha.
B) monitor, modem
C) sichqoncha, printer.
D) skaner, mikroprotsessor.

2. (3.5) dyuymli disketaga maksimal darajada qancha bayt ma'lumot sig'adi?

- A) 650 Mbayt
B) 1.44 Mbayt
C) 700 Mbayt
D) 144 Mbayt

3.Xotira qanday hajm birligi bilan o'lganadi.

- A) bayt B) Mbayt C) Kbayt D) hamma javob to'g'ri.

4.Kompyuterlarning qaysi guruhi dunyoda eng ko'p tarqalgan?

- A)Noutbuk
B)Shaxsiy EHM
C)Minikompyuter
D)Hammasi

5.Mikroprotsesssor bu...

- A)Xotira, shinalar, har xil elektron sxemalar joylashtiriladi.
B)Kompyuterning arifmetik va mantiqiy amallarini bajarib, ma'lumotlarni berilgan dastur asosida qayta ishlaydi.
C)Kompyuterda ma'lumotlarni saqlaydi
D)Kompyuterning kiritish va chiqarish portlari joylashadi.

6.Caps Lock tugmasining vazifasi.

- A)Katta yoki kichik harflarga o'tishni ta'minlaydi.
B)Kursorni sahifa boshiga o'tkazish uchun.
C)Funksional tugma bo'lib, dastur tuzilishiga qarab har xil vazifani bajaradi.
D)Kompyuter o'z ishini vaqtinchalik to'xtatadi.

7.Ma'lumotlar ketma-ketligining xotirada biror nom bilan saqlanishiga nima deb ataladi?

- A)Katalog B) Yorliq C)Fayl D) Papka

8.Fayl kengaytmasi nimani bildiradi?

- A)Fayl atributi
B)Fayl hajmini
C)Fayl turini
D)Fayl yaratilish sanasi

9.Kompyuterning hisoblash blokiga nimalar kiradi?

- A)Protsessor va xotira
B)Tashqi xotira va klaviatura
C)Xotira va monitor
D)Protsessor

10.Kompyuter yoqilgan vaqtda bajarilayotgan dasturlar va ishlanayotgan ma'lumotlarni vaqtincha saqlash qurilmasi bu –

- A)Videoxotira
B)Tashqi xotira
C)Tezkor xotira
D)Doimiy xotira

Oraliq nazorat:

1.C++ tilida necha turdagi leksemalar ishlatiladi?

- A) 5
- B) 4
- C) 3
- D) 6

2.C++ tilida identifikator nima?

- A) arifmetik amallar;
- B) funksiyalar;
- C) lotin harflari, ostki chiziq belgisi va sonlar ketma ketligidan iborat;
- D) o'zgarmas sonlardan iborat;

3. C++ tilida xizmatchi so'zlar deb nimaga aytiladi?

- A) Lotin harflari, ostki chiziq belgisi va sonlar ketma ketligidan iborat ifoda;
- B) Tilda ishlatiluvchi ya'ni dasturchi tomonidan o'zgaruvchilar nomlari sifatida ishlatish mumkin bo'lmagan identifikatorlar ;
- C) C++ kompilyatorida matnlarni formatlovchi maxsus belgilar;
- D) O'zgarmas e'lon qilinishi bilan unga qiymat beruvchi belgilar;

4. C++ tilida maxsus belgilar deb nimaga aytiladi?

- A) Lotin harflari, ostki chiziq belgisi va sonlar ketma ketligidan iborat ifoda;
- B) Tilda ishlatiluvchi ya'ni dasturchi tomonidan o'zgaruvchilar nomlari sifatida ishlatish mumkin bo'lmagan identifikatorlar ;
- C) O'zgarmas e'lon qilinishi bilan unga qiymat beruvchi belgilar;
- D) C++ kompilyatorida matnlarni formatlovchi belgilar;

5. Eskeyp simvollar deb nimaga aytiladi?

- A) '\' simvolidan boshlangan simvollar;
- B) '*' simvolidan boshlangan simvollar;
- C) '=' simvolidan boshlangan simvollar;
- D) ';' simvolidan boshlangan simvollar;

6. Ma'lumotlarning haqiqiy son turi necha qismdan iborat bo'lishi mumkin?

- A) 3
- B) 4
- C) 5
- D) 6

7. C++ tilida funksiyalarni yozish uchun qanday belgidan foydalaniladi?

- A) fugalali qavs – “{ “
- B) burchak qavs – “<<”
- C) to'g'ri burchakli qavs – “[“
- D) kichik qavs – “(“

8. C++ tilida funksiya tanasi deb nimaga aytiladi?

- A) Barcha o'zgarmaslar;
- B) Barcha leksemalar;
- C) Figurali qavslarni ichida joylashgan barcha satrlar;
- D) Barcha operatorlar;

9. Chiqarish operatori ko'rsatilgan qatorni tanlang

- A) (<<)

B) (=)

C) (+)

D) (;)

10.C++ tilida qanday o'zgaruvchilar ishlatiladi?

A) cout, cin;

B) tab, bell

C) main

D) int, float, char

Yakuniy nazorat (1-variant)

1.C++ tilida to'g'ri yakunlangan ifodani ko'rsating.

A) x=a+b;

B) #include <iostream>

C) x=a+b.

D) int main()

2. "endl" operatori qachon ishlatiladi?

A) satr boshida

B) funksiya tanasida

C) dastur oxirida

D) satr oxirida

3.Funta (#) belgisi nima vazifani bajaradi?

A) preprotessorga signal uzatadi;

B) dasturni ishga tushiradi;

C) hisoblashni bajaradi;

D) hech qanday ish bajarmaydi;

4. C++ tilida funksiya nima?

A) o'zgaruvchilar;

B) bir yoki bir necha amalni bajaruvchi dastur bloki;

C) hisob ishlarini bajaruvchi operator;

D) dasturni ishga tushiruvchi buyruq;

5.C++ tilida ifoda yakunida qanday belgi qo'yiladi?

A) "#"

B) "<<"

C) "? "

D) "; "

6.C++ tilida funksiya nomi oldidagi "int" kalit so'zi nimani bildiradi?

A) Funksiya bajarilishi natijasida qaytaraladigan qiymat turini;

B) Sarlavha sartini;

C) Funksiya tanasini;

D) Dasturni boshlashni;

7.C++ tilida "<<" – amalining ma'nosi?

A) o'zlashtirish amali;

B) ko'paytirish amali;

C) burchak qavs;

D) ma'lumot uzatish amali.

8. Haqiqiy sonning butun va kasr qismlari qanday belgi bilan ajratiladi?

A) '.' belgisi bilan;

B) "<<"

C) "? "

D) "; "

9. "a!=b" amalining ma'nosi nima?

A) "a katta yoki teng b"

B) "a teng b"

C) "a teng emas b"

D) "a kichik yoki teng b"

10. C++ tilida " = " – amalining ma'nosi?

A) o'zlashtirish amali;

B) ko'paytirish amali;

C) burchak qavs;

D) ma'lumot uzatish amali.

Yakuniy nazorat (2-variant)

1. "log(x)" – funksiya qanday amal bajaradi?

A) x butun sonining absolyut qiymatini qaytaradi;

B) x haqiqiy sonining absolyut qiymatini qaytaradi;

C) x sonining natural logarifmini qaytaradi;

D) x sonining kvadrat ildizini qaytaradi;

2. "a==b" amalining ma'nosi nima?

A) "a katta yoki teng b"

B) "a teng b"

C) "a teng emas b"

D) "a kichik yoki teng b"

3. Mantiqiy ko'paytirish operatori qanday belgi orqali belgilanadi?

A) "*"

B) "!"

C) "||"

D) "&&"

4. Inkrement amali nima uchun ishlatiladi?

A) operand qiymatini bittaga oshirish uchun ;

B) operand qiymatini bittaga kamaytirish uchun ;

C) sonlarni qo'shish uchun;

D) mantiqiy qo'shish uchun;

5. Takrorlash operatorlari to'g'ri ko'rsatilgan qatorni tanlang.

A) if, else;

B) cout, for;

C) for, while;

D) break, switch;

6. "abs(x)" – funksiya qanday amal bajaradi?

A) x butun sonining absolyut qiymatini qaytaradi;

- B) x haqiqiy sonining absolyut qiymatini qaytaradi;
- C) x sonining natural logarifmini qaytaradi;
- D) x sonining kvadrat ildizini qaytaradi;

7. Global o'zgaruvchilar deb nimaga aytiladi?

- A) Funksiya ichida e'lon qilingan o'zgaruvchilar;
- B) Dastur ishlashida o'zgaruvchilar;
- C) funksiya aniqlanishidan tashqarida e'lon qilingan o'zgaruvchilar;
- D) Boshlang'ich qiymat berilmagan o'zgaruvchilar;

8. Qanday belgilar bilan tugagan barcha belgilar ketma-ketligi izoh hisoblanadi?

- A) “//” dan boshlanib, “//” belgilar oralig'ida joylashgan barcha belgilar ketma-ketligi;
- B) “!” dan boshlanib, “!” belgilar oralig'ida joylashgan barcha belgilar ketma-ketligi;
- C) “<<” dan boshlanib, “>>” belgilar oralig'ida joylashgan barcha belgilar ketma-ketligi;
- D) “/*” dan boshlanib, “*/” belgilar oralig'ida joylashgan barcha belgilar ketma-ketligi;

9. Kompilyatsiya jarayonini necha bosqichdan tashkil topadi?

- A) Ikki bosqichdan
- B) Uch bosqichdan
- C) To'rt bosqichdan
- D) Besh bosqichdan

10. Figurali qavslar nima uchun ishlatiladi?

- A) Natijani chiqarish uchun;
- B) Takrorlashni to'xtatish uchun;
- C) Matematik hisoblarni bajarish uchun;
- D) Funksiya tanasini belgilash uchun;

Yakuniy nazorat (3-variant)

1. “sqrt(x)” – funksiya qanday amal bajaradi?

- A) x butun sonining absolyut qiymatini qaytaradi;
- B) x haqiqiy sonining absolyut qiymatini qaytaradi;
- C) x sonining natural logarifmini qaytaradi;
- D) x sonining kvadrat ildizini qaytaradi;

2. Mantiqiy qo'shish operatori qanday belgi orqali belgilanadi?

- A) “*”
- B) “!”
- C) “||”
- D) “&&”

3. Dekrement amali nima uchun ishlatiladi?

- A) operand qiymatini bittaga oshirish uchun ;
- B) operand qiymatini bittaga kamaytirish uchun ;
- C) sonlarni qo'shish uchun;
- D) mantiqiy qo'shish uchun;

4. “fabs(x)” – funksiya qanday amal bajaradi?

- A) x butun sonining absolyut qiymatini qaytaradi;
- B) x haqiqiy sonining absolyut qiymatini qaytaradi;
- C) x sonining natural logarifmini qaytaradi;
- D) x sonining kvadrat ildizini qaytaradi;

5. Lokal o'zgaruvchilar deb nimaga aytiladi?

- A) Funksiya ichida e'lon qilingan o'zgaruvchilar;
- B) Dastur ishlashida o'zgarmaydigan o'zgaruvchilar;
- C) funksiya aniqlanishidan tashqarida e'lon qilingan o'zgaruvchilar;
- D) boshlang'ich qiymat berilmagan o'zgaruvchilar;

6. Yuqori darajadagi dasturlash tillari nima?

- A) Matematik hisoblarni bajaruvchi tillar;
- B) Dastur tuzish uchun mo'ljallangan tillar;
- C) Dastur ta'minotini yaratish jarayonini osonlashtirish uchun yaratilgan tillar;
- D) Tasvirlar yaratish uchun mo'ljallangan tillar;

7. “break” kalit so'zi nima uchun ishlatiladi?

- A) Natijani chiqarish uchun;
- B) Takrorlashni to'xtatish uchun;
- C) Matematik hisoblarni bajarish uchun;
- D) Tasvirlar yaratish uchun;

8. “continue” operatori qanday vazifani bajaradi?

- A) takrorlashni to'xtatish va boshqarishni keying qadamga o'tkazish ;
- B) Matematik hisoblarni bajarish;
- C) Shartni tekshirish;
- D) Natijani chiqarish;

9. “else” operatori qanday vazifani bajaradi?

- A) Shart rostligini aniqlovchi operator;
- B) Takrorlashni to'xtatuvchi operator;
- C) Natijani chiqaruvchi operator;
- D) shart yolg'onligini aniqlovchi operator;

10. Mantiqiy inkor operatori qanday belgi orqali belgilanadi?

- A) “*”
- B) “!”
- C) “||”
- D) “&&”

Yakuniy nazorat (4-variant)

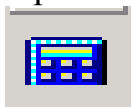
1. MATHCAD dasturida funksiyalarni ishlatish uchun qaysi muloqot oynadan foydalaniladi?

- A) MATHCAD dasturida funksiyalarni ishlatish uchun matn maydonidan foydalaniladi.
- B) MATHCAD dasturida funksiyalarni ishlatish uchun matematik maydonidan foydalaniladi.
- C) MATHCAD dasturida funksiyalarni ishlatish uchun “Qo'yish” maydonidan foydalaniladi.

D)MATHCAD dasturida Insert Function (Funksiyani qo'yish) tugmasiga bog'langan muloqot oynasidan foydalaniladi.

2.O'zgaruvchi diapazonini ko'rsatishda qanday belgi kiritiladi?

- A)O'zgaruvchi diapazonini ko'rsatishda klaviaturadan (+) kiritiladi
- B)O'zgaruvchi diapazonini ko'rsatishda klaviaturadan (:) kiritiladi
- C)O'zgaruvchi diapazonini ko'rsatishda klaviaturadan (;) nuqta vergul kiritiladi.
- D)O'zgaruvchi diapazonini ko'rsatishda klaviaturadan (<<) kiritiladi.



3.MathCadda tugmasi qanday vazifani bajaradi?

- A)Arifmetik hisob kitoblarni hal qilish.
- B) munosabat va mantiqiy amallardan foydalanish mumkin.
- C)Sonlarni formatlash mumkin.
- D)Hisoblash natijalarini o'zgartiradi.

4. MATHCADda diskret o'zgaruvchilar nima.

- A)MATHCADda diskret o'zgaruvchilar sikl operatori.
- B)MATHCADda diskret o'zgaruvchilar for operatori.
- C)MATHCADda diskret o'zgaruvchilar while operatori.
- D)MATHCADda diskret o'zgaruvchilar if operatori.

5. Insert buyruqlar to'plami nima vazifani bajaradi?

- A) xujjatga turli ob'ektlarni joylashtirish uchun xizmat qiluvchi buyruqlar to'plami;
- B)matn, formula va grafiklarni formatlovchi buyruqlar to'plami;
- C) hisoblash jarayonini boshqaruvchi buyruqlar to'plami;
- D)simvolli hisoblashlar bo'yruqlari to'plami;

6.MathCad dasturida abzas uzunligini aniqlash uchun qaysi buyruq ishlatiladi?

- A) Result
- B) Tabs
- C) Text
- D) Equation

7.MATHCAD ishga tushganda avtomatik ravishda uning ishchi hujjat qanday nomlanadi?

- A) Dokument 1
- B) Kniga 1
- C) Untitled 1
- D) Prezentatsiya 1

8.Asosiy matematik amallar shablonini ko'rsating

- A)Calculator (Kalkulyator)
- B) Calculus (Hisoblash)
- C)Matrix (Massiv)
- D)Evoluation (Baholash)

9.MathCadda bajarishni bekor qilish uchun qaysi tugmalardan foydalaniladi?

- A) Ctrl+X; B) Ctrl+C; C) Ctrl+V; D) Ctrl+Z

10.MathCad matematik ifodalarni nechta belgi aniqligida ifodalaydi?

- A) 200 belgi B) 20 belgi C) 120 belgi D) 10 belgi

Yakuniy nazorat (5-variant)

1.Mathcadda bosmaga chiqarish uchun qaysi buyruq beriladi?

- A) Ctrl+V B) Ctrl+C C) Ctrl+P D) Alt+C

2.For sikl operatorining vazifasi nimadan iborat?

- A)For sikl operatorini takrorlanishlar soni oldindan aniq bo'lganda ishlatiladi.
B)For sikl operatorini takrorlanishlar soni oldindan aniq bo'lmaganda ishlatiladi.
C) Agar shart yolg'on bo'lsa D)Qiymatlarni lokal yuborishda ishlatiladi

3. While sikl operatori qachon ishlatiladi?

- A)While sikl operatori takrorlanishlar soni oldindan aniq bo'lgan hollarda
B) Agar shart yolg'on bo'lsa
C)While sikl operatori takrorlanishlar soni oldindan aniq bo'lmagan hollarda
D)Qiymatlarni lokal yuborishda ishlatiladi

4.Massiv va massiv amallarini bajarish shabloni ko'rsating

- A)Calculator (Kalkulyator)
B) Calculus (Hisoblash)
C)Matrix (Massiv)
D)Evaluatoin (Baholash)

5.MathCadda nusxa olish uchun qaysi tugmalardan foydalaniladi?

- A) Ctrl+X; B) Ctrl+C; C) Ctrl+V; D) Ctrl+Z

6.Mathcad dasturida nechta menyu bor?

- A) 5 B) 6 C) 9 D) 8

7.MathCad dasturida matnlarni formatlash uchun qaysi buyruq ishlatiladi?

- A) Result
B) Tabs
C) Text
D) Equation

8. O'zgaruvchilarga qiymat berish uchun qanday operator ishlatiladi?

- A)O'zgaruvchilarga qiymat berish uchun qo'shish operatori "+" ishlatiladi
B)O'zgaruvchilarga qiymat berish uchun yuborish operatori "==" ishlatiladi
C)O'zgaruvchilarga qiymat berish uchun ko'paytirish operatori "*" ishlatiladi
D)O'zgaruvchilarga qiymat berish uchun ayirish operatori "-" ishlatiladi.



9. belgi nima vazifani bajaradi?

- A) Belgilangan yoki endi yoziladigan absatsni o'rtadan joylashtirish.
B) Belgilangan yoki endi yoziladigan absatsni chap tomondan joylashtirish.
C) Har bir abzatsni marker (biror belgi) bilan belgilash.
D) Belgilangan yoki endi yoziladigan absatsni o'ng tomondan joylashtirish.

10. MATHCAD dasturida funksiyalarni ishlatish uchun qaysi muloqot oynadan foydalaniladi?

- A)MATHCAD dasturida funksiyalarni ishlatish uchun matn maydonidan foydalaniladi.

B)MATHCAD dasturida funksiyalarni ishlatish uchun matematik maydonidan foydalaniladi.

C)MATHCAD dasturida funksiyalarni ishlatish uchun “Qo’yish” maydonidan foydalaniladi.

D)MATHCAD dasturida Insert Function (Funksiyani qo’yish) tugmasiga bog’langan muloqot oynasidan foydalaniladi.

Yakuniy nazorat (6-variant)

1.Differensiallash, integrallash, summani hisoblash shablonini ko’rsating.

A)Calculator (Kalkulyator)

B) Calculus (Hisoblash)

C)Matrix (Massiv)

D)Evolution (Baholash)

2.O’zgaruvchi diapazonini ko’rsatishda qanday belgi kiritiladi?

A)O’zgaruvchi diapazonini ko’rsatishda klaviaturadan (+) kiritiladi

B)O’zgaruvchi diapazonini ko’rsatishda klaviaturadan (:) kiritiladi

C)O’zgaruvchi diapazonini ko’rsatishda klaviaturadan (;) nuqta vergul kiritiladi.

D)O’zgaruvchi diapazonini ko’rsatishda klaviaturadan (<<) kiritiladi.

3.MathCadda nusxani qo’yish uchun qaysi tugmalardan foydalaniladi?

A) Ctrl+X; B) Ctrl+C; C) Ctrl+V; D) Ctrl+Z

4. MathCad ishchi oynasida hisoblash jarayonini boshqaruvchi buyruqlar to’plami berilgan qatorni ko’rsating.

A) View

B) Edit

C) Insert

D) Math

5. belgi nima vazifani bajaradi?

A) Belgilangan yoki endi yoziladigan matn shriftining o’lchovi ko’rsatiladi.

B) Berilgan yoki endi yoziladigan matnning shriftini tanlash.

C) Belgilangan yoki endi yoziladigan matnning qalin yoki oddiyligini belgilash.

D) Belgilangan yoki endi yoziladigan matnning tagi chizilgan yoki oddiyligini belgilash.

6.MathCad bu -

A) Grafiklar bilan ishlovchi sistema.

B) Buxgalteriya dasturi hisoblanadi.

C) Tenglama va tenglamalar sistemasini yechish uchun mo’ljallangan dastur.

D) Matematikaning turli sohalaridagi masalalarini yechishga mo’ljallangan sistema.

7.MathCad dasturi nechta xarakterli interfeyslardan iborat?

A) 5 ta B) 6 ta C) 4 ta D) 3

8.MathCad dasturida ishni yakunlash ketma-ketligi berilgan qatorni ko’rsating.

A) ALT+F4

- B) ALT+F3
- C) ALT+F5
- D) ALT+F2

9. Matnlarni tahrirlash uchun mo'ljallangan bo'yuqlar to'plami berilgan qatorni ko'rsating.

- A) View
- B) Edit
- C) Insert
- D) Math

10. MathCad dasturida asboblari paneli necha qismdan iborat?

- A) 4
- B) 3
- C) 2
- D) 5

Yakuniy nazorat (7-variant)

1. Qiymatlarni yuborish operatori va natijalarni chiqarish operatorini ko'rsating.

- A) Calculator (Kalkulyator)
- B) Calculus (Hisoblash)
- C) Matrix (Massiv)
- D) Evaluation (Baholash)

2. MathCadda belgilangan sohani qirqib olish uchun qaysi tugmalardan foydalaniladi?

- A) Ctrl+X;
- B) Ctrl+C;
- C) Ctrl+V;
- D) Ctrl+Z

3. MathCad dasturi ishga tushganda undagi ish varag'i nima deb nomlanadi?

- A) Dokument 1
- B) List 1
- C) Prezentatsiya 1
- D) Worksheet

4. Operator deb nimaga aytiladi?

- A) Ma'lumotlar ustida bajariladigan ma'lum bir amalni bajarish uchun ishlatiladigan belgi.
- B) Barcha elementlari noldan iborat bo'lgan matrisa
- C) Hisoblash jarayonini boshqaruvchi buyruqlar to'plami
- D) Mexanika, matematika, fizika, muxandislik va boshqaruv masalalarini yechish

5. Hujjatga turli ob'ektlarni joylashtirish uchun xizmat qiluvchi buyruqlar to'plami berilgan qatorni ko'rsating.

- A) View
- B) Edit
- C) Insert
- D) Math



6. belgi nima vazifani bajaradi?

- A) Belgilangan yoki endi yoziladigan matn shriftining o'lchovi ko'rsatiladi.

- B) Berilgan yoki endi yoziladigan matnning shriftini tanlash.
- C) Belgilangan yoki endi yoziladigan matnning qalin yoki oddiyligini belgilash.
- D) Belgilangan yoki endi yoziladigan matnning tagi chizilgan yoki oddiyligini belgilash.

7.MATHCAD hujjatiga matn kiritish uchun qaysi menyudan foydalaniladi?

- A) Matn kiritish uchun Bosh menyudan Insert→Text Region (Qo'yish→Matn maydoni) buyrug'ini berish;
- B) matn kiritish uchun Bosh menyudan Math buyrug'ini berish;
- C) matn kiritish uchun Bosh menyudan Graph buyrug'ini berish;
- D) matn kiritish uchun Bosh menyudan Insert buyrug'ini berish;

8.MathCad dasturida faylni xotiraga olish buyruqlar ketma-ketligini ko'rsating.

- A) File-Save As
- B) File-Exit
- C) ALT+F6
- D) Shift + F5

9. MathCad dasturida tugma nima vazifani bajaradi?

- A) Vektorlar va matritsalar ustida amallar bajarish.
- B) Tenglamalarning ildizi topiladi.
- C) Ifodalarni analitik hisoblash.
- D) Dasturlashda kerak bo'ladigan operatorlarni tanlash.

10.MATHCAD dasturining nomlanishi nechta so'zdan iborat?

- A) 2 ta MATHematika (matematika) va CAD (avtomatik loyihalash sistemasi)
- B) 3 ta MATHematika (matematika) va CAD (avtomatik loyihalash sistemasi) va tizim so'zlaridan iborat;
- C) 1 ta

Yakuniy nazorat (8-variant)

1.“MATLAB” dasturida qator deb nimaga aytiladi?

- A) amallar bajarish ketma-ketligi
- B) apostrof ichidagi simvollar ketma-ketligi
- C) diagonali birga teng matrisa
- D) diagonali nolga teng matrisa

2.“MATLAB” tizimining asosiy oynasi necha bo'limdan iborat?

- A) 9
- B) 6
- C) 7
- D) 8

3.“MATLAB” dasturida (./) arifmetik amal qanday vazifani bajaradi?

- A) Massiv mos elementlari bo'yicha ko'paytirish
- B) Massiv mos elementlari bo'yicha qo'shish
- C) O'lchovlar bir xil massiv mos elementlari bo'yicha bo'lish
- D) Darajaga ko'tarish

4. Matlab tizimining asosiy manyular satrini ko'rsating.

- A) Fayl, Edit, Math, Insert, Help
- B) Fayl, Edit, View, Web, Windows, Help
- C) Fayl, Edit, Math, Insert, Calculus, Help
- D) Fayl, Edit, Math, Insert, Matrix, Help

5. "MATLAB" dasturida buyruq satri qanday simvol bilan boshlanadi?

- A) (+) simvoli
- B) (:) simvoli
- C) (>>) simvoli.
- D) (=) simvoli

6. "MATLAB" dasturida o'zgaruvchi nomi nechta belgidan oshmasligi shart?

- A) 63 B) 200 C) 8 D) 42

7. "MATLAB" dasturida qachon buyruq natijasi ans(inglizcha answer-javob) o'zgaruvchisi orqali beriladi?

- A) o'zgaruvchi nomi yozilsa
- B) o'zgaruvchi nomi yozilmay bajarilsa
- C) o'zgaruvchi nomi sig'may qolsa
- D) o'zgaruvchi nomi tushunarsiz bo'lsa

8. "MATLAB" dasturida (.*) arifmetik amal qanday vazifani bajaradi?

- A) ko'paytirish
- B) qo'shish
- C) Massiv mos elementlari bo'yicha ko'paytirish
- D) Massiv mos elementlari bo'yicha bo'lish

9. "MATLAB" dasturida "log(z)" funksiyasi qanday vazifani bajaradi?

- A) sonning natural logarifmini hisoblaydi;
- B) sonning o'nli logarifmini hisoblaydi;
- C) sonning logarifmini hisoblaydi;
- D) kompleks sonining modulini hisoblaydi;

10. "MATLAB" dasturida "%" belgisi qanday vazifani bajaradi?

- A) o'zgaruvchilarni belgilashda ishlatiladi;
- B) hech qanday vazifa bajarmaydi;
- C) foizlarni hisoblashda ishlatiladi;
- D) izohlarni yozishda ishlatiladi;

Talabalar bilimini baholash mezonlari

Kimyo injinerida axborot texnologiyalari kursini semestrda o'qish va o'zlashtirish davomida olinadigan eng yuqori baho miqdori **“5”** baho.

Baholash semestr davomida talabaning o'quv ishlarni bajarish va o'zlashtirish darajasini aniqlash uchun joriy, oraliq nazoratga va yakuniy nazoratga ajratilgan.

Joriy va oraliq ballarni to'plash talabaning o'quv ishlarini o'z vaqtida va sifatli bajarish va topshirishi, shuningdek o'quv mashg'ulotlarida, ilmiy-metodik ishlarda va boshqa o'quv tadbirlarda faol qatnashishiga bog'liq.

Semestr mobaynida **joriy** nazorat kunlik baholab boriladi.

Semestr mobaynida 2 ta oraliq nazorat belgilangan. (haftasiga 2 soat o'qitiladigan fanlarda 1 ta oraliq).

Birinchi oraliq nazorat o'quv semestrning 9-xaftasida o'tkaziladi.

Ikkinchi oraliq nazorat esa o'quv semestrning 18-xaftasida o'tkaziladi.

Oraliq nazoratlar kurs dasturidagi bo'limlar tugallangandan keyin yozma ish yoki test tarzida o'tkaziladi.

Yozma ish tarkibida kurs dasturining asosiy mavzularini o'z ichiga olgan bo'lib, 3 yoki 4 ta savoldan iborat bo'ladi. Har bir savolning murakkabligi va hajmiga qarab 2,3,4,5 bahogacha berish nazarda tutilgan.

Test shaklida o'tkazilsa variant 30 ta savoldan iborat bo'lsa quyidagicha baholanadi:

27-30 “5” (a'lo)

21-26 “4” (yaxshi)

18-20 “3” (qoniqarli)

17 va undan kam bo'lsa “2” (qoniqarsiz)

ON turini topshirmagan va “2” (qoniqarsiz) baho olgan talaba YN ga kiritilmaydi va akademik qarzdor sanaladi. O'qish davrida dars qoldirgan talabalar mazkur darslar buyicha referat sifatida hisobot topshirgan holdagina oraliq va yakuniy nazoratlarga kiritiladi.

Yakuniy nazoratda joriy va oraliq nazoratda to'plangan baholar miqdori hisobga olinmaydi. Talaba sababli holatda ON va YaN turlariga ishtirok etmasa, ushbu talabaga tegishli nazorat turini qayta topshirishga fakultet dekani farmoyishi bilan ruxsat beriladi;

Semestr davomida kurs bo'yicha to'plagan ballar yig'indisi quyidagi o'zlashtirish ko'rsatkichi bo'yicha baholanadi:

Baho	Talabaning bilim darajasi
a'lo	talaba mustaqil qaror qabul qilsa, ijodiy fikrlasa, bilimini amalda qo'llay olsa va savol mohiyatini bilib, xulosa chiqarganligi uchun
yaxshi	talaba qisman mustaqil qaror qabul qilsa va ijodiy fikrlasa hamda bilimini amalda qo'llay olsa, berilgan savolning mohiyati haqida tushunchaga ega bo'lsa
qoniqarli	talaba bilimini amalda qisman qo'llay olsa, savol mohiyatini tasavvur qilsa
qoniqarsiz	Aniq tasavvurga ega emaslik, bilmaslik

*Yakuniy nazorat turi fakultet dekani, o'quv-uslubiy boshqarma bilan kelishgan holda va o'quv ishlari bo'yicha prorektor tomonidan imzolangandan so'ng tegishli fan bo'yicha o'quv mashg'ulotini olib borgan o'qituvchi ishtirokisiz o'tkaziladi.

Yakuniy nazorat shakli fan xususiyati va o'quv auditoriya soatidan kelib chiqib, kafedra belgilaydi.

Yakuniy nazorat

Test shaklida o'tkazilsa variant 30 ta savoldan iborat bo'lsa quyidagicha baholanadi:

27-30 "5"(a'lo)

21-26 "4"(yaxshi)

18-20 "3"(qoniqarli)

17 va undan kam bo'lsa "2"(qoniqarsiz)

"Yozma" shaklda o'tkazilsa savollar tarkibi 3 tadan iborat bo'ladi va har bir savol "5" baholi tizimda baholanib o'rtacha baho hisobga olinadi.

Yakuniy nazorat test shaklida o'tkazilsa O'qituvchi jurnalga bahoni shu kunning o'zida qayd etiladi. ON va YaN lar yozma ish shaklida bo'lsa, o'qituvchi 3 kun muddatda baholarni jurnalga qayd etadi.

"5"(a'lo), 4(yaxshi), 3(qoniqarli) baho olgan talabaga qayta topshirishga ruxsat berilmaydi;

Talaba sababsiz ravishda nazorat turiga qatnashmasa "0" belgisi qo'yiladi;

YaN turi bo'yicha baho jurnalga qayd etilganda, shu kuni talabaning baholash daftariga ham qayd etilishi shart;

YaN turi bo'yicha 2 (qoniqarsiz) baho yoki jurnalga "0" belgisi yozib qo'yilganda ushbu baho va belgi talabaning baholash daftariga yozilmaydi;

Qo'shimcha materiallar.

«Aqliy hujum» metodining asosiy qoidalari:

- Auditoriyada olg'a surilgan fikr va g'oyalar tanqid ostiga olinmaydi va baholanmaydi;
- Har bir shaxs tomonidan taklif qilinayotgan fikr va g'oyalar qanchalik fantastik va aniq bo'lsa ham, uni baholashdan o'zingizni tiying!
- Tanqid qilmang - hamma bildirilgan fikrlar bir xilda bebahodir.
Fikr bildirilayotganda notiqni bo'lma!
Maqsad - mazmunli fikr va g'oyalar sonini ko'paytirish.
Qanchalik ko'p fikr va g'oyalar bildirilsa, shunchalik yaxshi. Yangi va bebaho fikr va g'oyaning paydo bo'lish ehtimoli ko'payadi.
Agar fikrlar qaytarilsa asabiylashmang va hayron bo'lmang.
Bu muammo faqatgina ma'lum usullar yordamidagina hal bo'lishi mumkin, deb o'ylamang.
Fikrlar «hujumi» ni o'tkazish vaqti aniqlanadi va unga qat'iy rioya qilinishi shart.
Berilgan savolga qisqacha (1-2 so'zdan iborat) javob beriladi.

Guruhlarda ishlash qoidasi

Sherigingizni diqqat bilan tinglang.

Guruh ishlarida o'zaro faol ishtirok eting, berilgan topshiriqlarga javobgarlik bilan yondashing.

Agar Sizga yordam kerak bo'lsa, albatta guruh a'zolariga murojaat qiling.

Agar Sizdan yordam so'rasha, albatta yordam bering.

Guruhlar faoliyati natijalarini baholashda hamma ishtirok etishi shart!

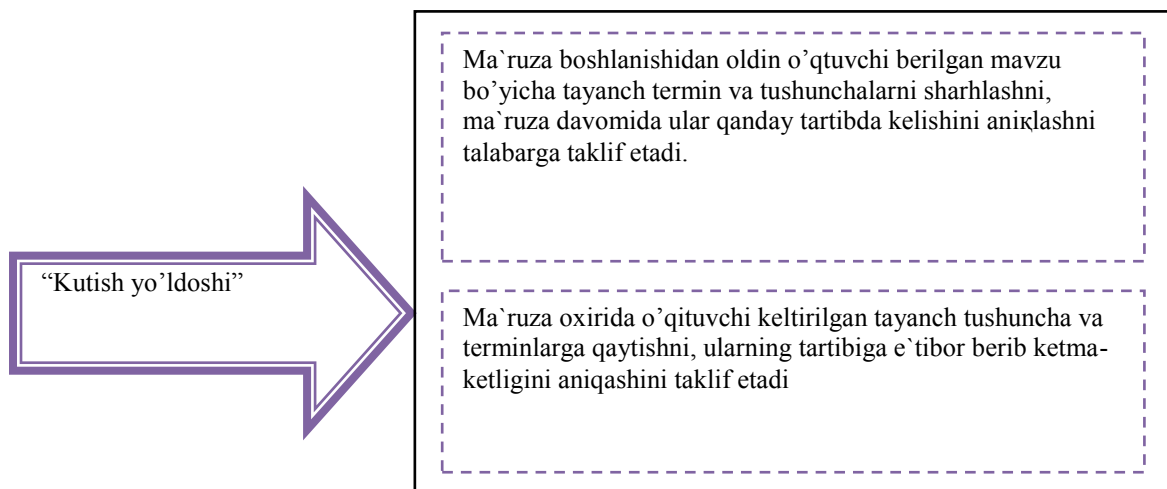
Shuni aniq tushuningiz lozim:

Boshqalarga o'rgatish orkali o'zimiz o'rganamiz;

Biz bitta kemadamiz: yoki birgalikda suzib chiqamiz, yoki birgalikda cho'kib ketamiz.

1.2-ilova

“Kutish yo’ldoshi” usulini qo’llash qoidasi

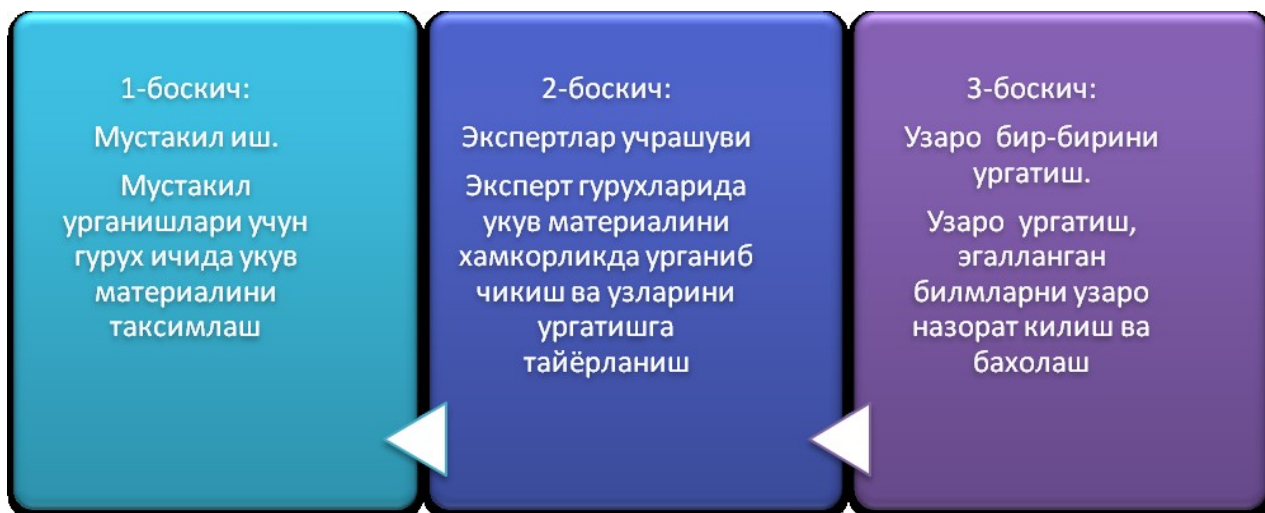


1.3-ilova

«Zigzag» texnikasi

O’zaro o’qitish yagona tamoyilga asoslangandir: o’quv guruhi kichik guruhlariga bo’linadi. Har bir guruh a’zosi o’rganilayotgan mavzuning ma’lum sohasi bo’yicha ekspert bo’ladi va boshqalarga o’rgatadi.

Har bir guruhning asosiy maqsadi: har bir talaba mavzuni to’liq o’zlashtirishi kerak.



1.4-ilova

Klaster tuzish qoidalari

Hayolingga nima kelsa shuni yoz. Fikrlarning sifatiga e’tibor bermang.
Yozuvning orfografik va boshqa xatolariga e’tibor bermang.
Ajratilgan vakt tugamaguncha yozishni to’xtatmang.
Agar fikrlar hech kelavermasa, to yangi fikrlar kelguncha qog’ozga rasmlar chizing.
Iloji boricha ko’proq bog’lanishlarni qurishga harakat qiling.
Fikrlar sonini va ular orasidagi bog’lanishlar sonini chegaralamang.

1.5-ilova

3x3 usulini qo'llash qoidasi

1. Talabalarni 3 ta guruhga ajratish lozim.
2. Uchta guruhga 3 ta savol beriladi.
3. Ma'lum bir vaqtdan so'ng topshiriqlarni yig'ib olish kerak.
4. Topshiriqlarni guruhlararo almashtirish kerak. (3-marta)
5. Topshiriqlarni birinchi holatdagi guruhlariga qaytarish lozim.
6. Prezentsiya qilinadi.
7. Kamchilik va yutuqlar aytiladi.

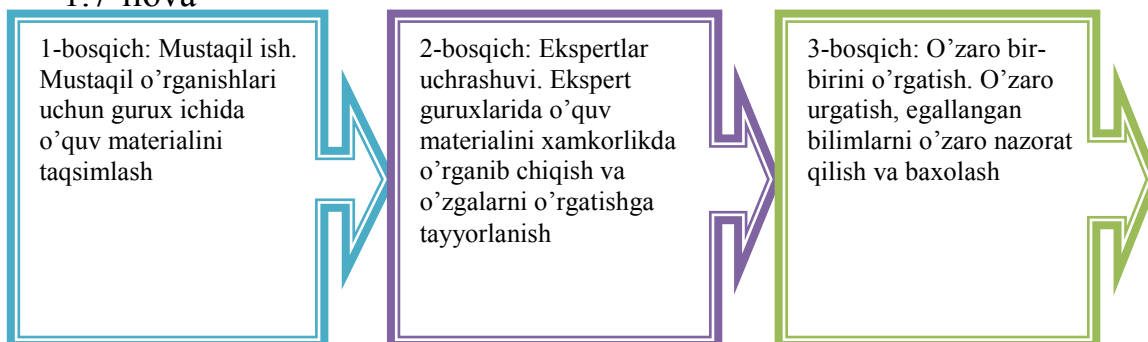
1.6-ilova

Munozara ishtirokchisiga eslatma
Munozara munosabatlarni hal etish emas, balki muammolarni echimini hamkorlikda topish metodidir.
Boshqalar xam fikr bildirishlariga imkoniyat yaratish uchun uzoq gapirma.
Aqli fikrlarning maqsadga etishi uchun so'zlaringni taroziga sol, o'ylab, so'ngra gapir, his-tuyg'ularingni nazorat qil.
Taqrizchi va opponent fikrini to'g'ri tushunishga harakat qil. Uning fikrini hurmat qil.
Faqat munozara mavzusi bo'yichagina, aniq fikr bildir.
O'z taqdimoting bilan kimgadir yoqishga yoki aksincha xafa qilishga urinma.

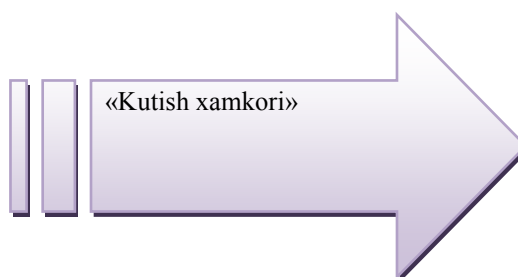
BBB usuli jadvali

Bilaman	Bilishni xoxlayman	Bilib oldim
1	2	3

1.7-ilova



1.8-ilova



O'qituvchi ma'ruza boshlanishidan oldin talabalarga mavzu buyicha yozdirgan asosiy tushuncha na iboralarga umumiy tushuncha berishni va ularning ma'ruza jarayonida kullanilishining taxminiy ketma-ketligini aniqlashni taklif qiladi.

O'qituvchi ma'ruza yakunida talabalarga asosiy tushuncha va iboralarga qaytishni, hamda uni asl natijalar bilan solishtirishni taklif qiladi.

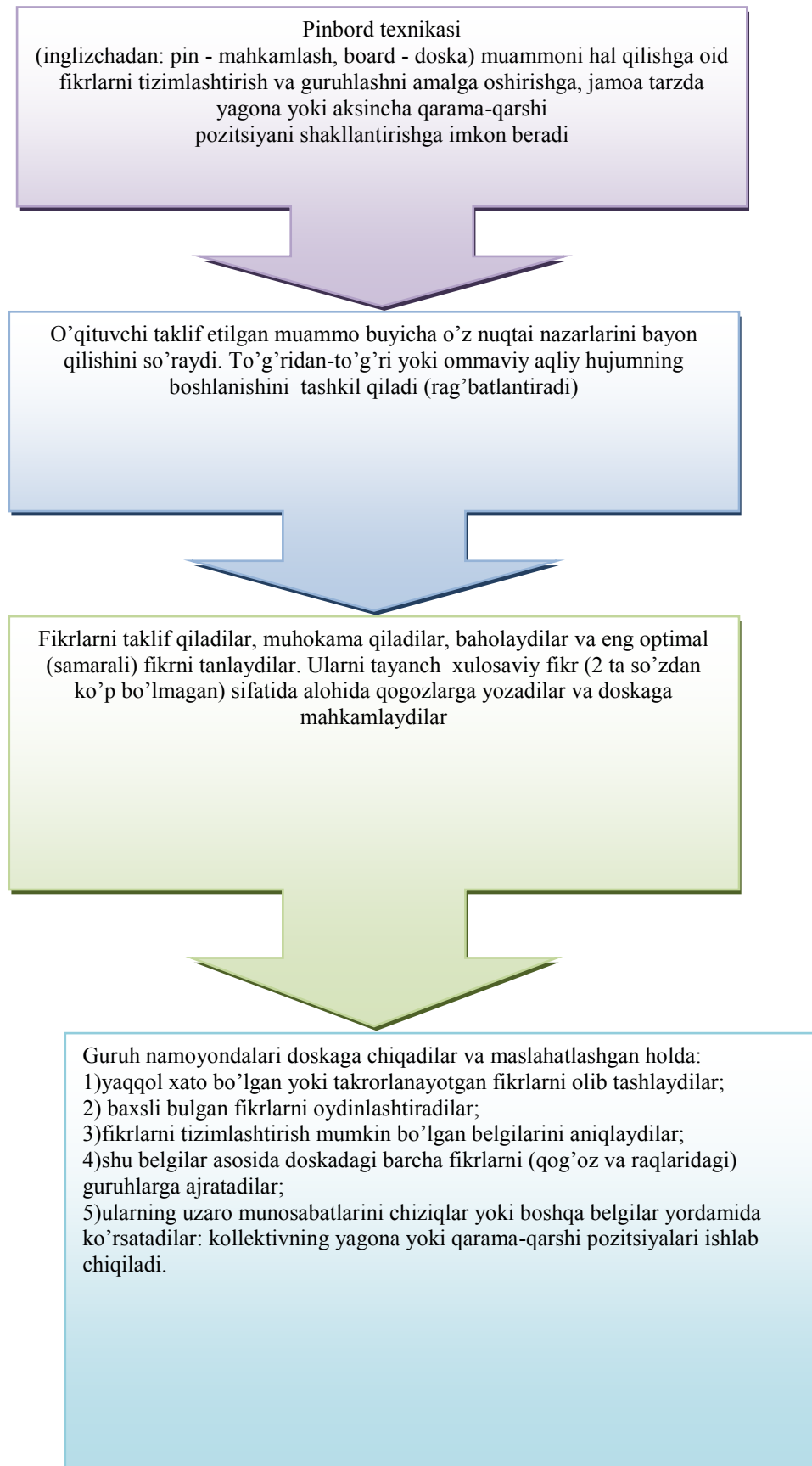
1.9-ilova

Munozara ishtirokchisiga eslatma.

1. Munozara munosabatlarni hal etish emas, balki muammolarni yechimini hamkorlikda topish metodidir.
2. Boshqalar xam fikr bildirishlariga imkoniyat yaratish uchun uzoq gapirma.
3. Aqli fikrlaring maksadga etishi uchun so'zlaringni taroziga sol, uylab, so'ngra gapir, his-tuyg'ularingni nazorat qil.
4. Taqrizchi va opponent fikrini to'g'ri tushunishga harakat qil. Uning fikrini hurmat qil. Faqat munozara mavzusi bo'yichagina, aniq fikr bildir. O'z taqdimoting bilan kimgadir yoqishga yoki aksincha xafa qilishga urinma.

1.10-ilova

Pinbord texnikasi

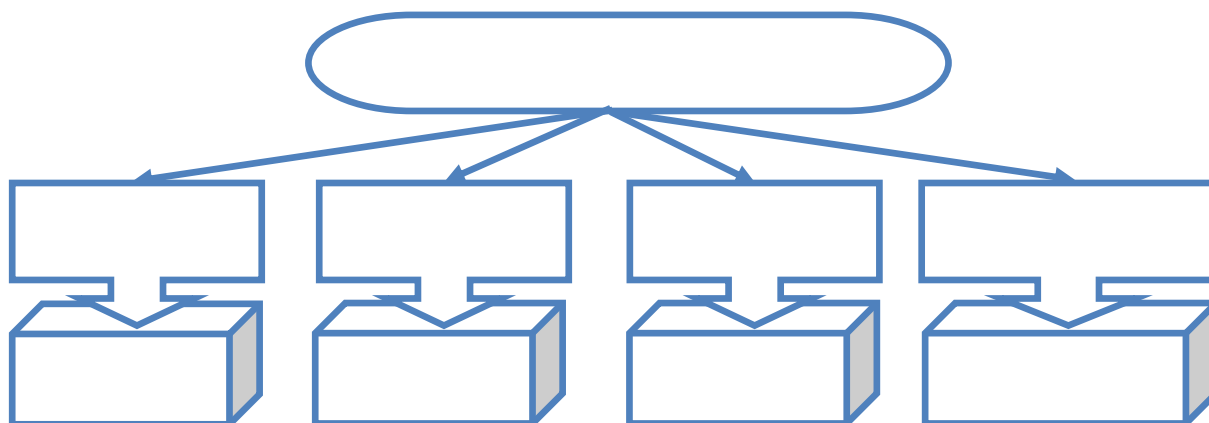


1.11-ilova Insert usuli



Insert jadvali

	+	-	?



1.12-ilova

Delfi texnikasi –
muammoni
yechimlaridan eng
sarasini tanlab olish
va baholash
imkoniyatini beradi.

Гурухнинг ҳар бир аъзоси таклиф килинган вариантларни муҳимлик даражасидан келиб чиққан ҳолда гурухлайди (1- энг муҳими, унча муҳим эмас) ва натижаларни ведомостнинг «Д» категориясига ёзиб қўяди.

Ҳар бир вариант оптималлик/ тежамкорлик /самарадорлик ва бошқа курсаткичлар даражаси асосида 10 балли тизим бўйича баҳоланади (1- энг юқори балл, 10- энг паст балл).
Натижалар «Б» каторига ёзилади.

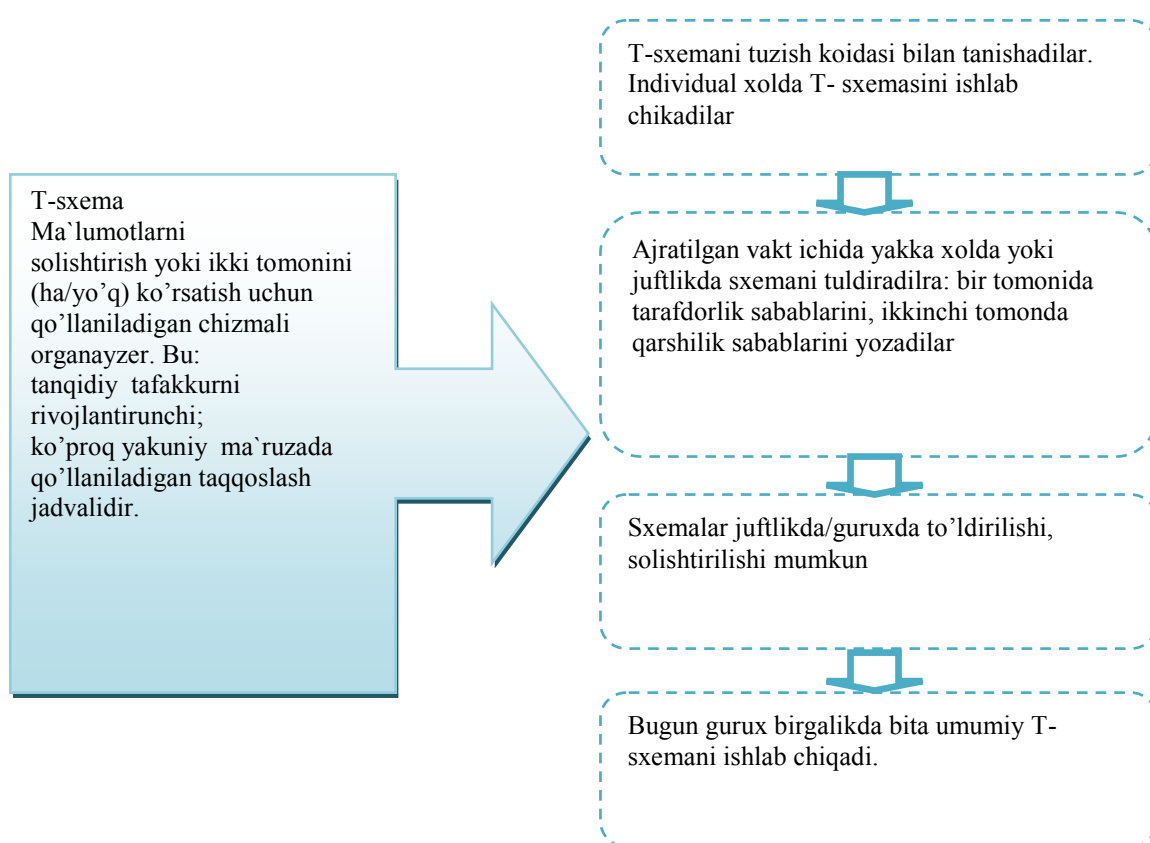
«Д» ва «Б» каторларининг натижалари умумлаштирилади ва якуний «Х» каторига ёзилади.

Баҳолаш варағи ишлаб чиқилади. Энг кам ҳажмдаги натижа энг оптимал вариантни курсатади.

Alternativ g'oyalarni (guruhlarda) baholash varag'i

Guruh	Alternativ g'oyalar								
	1-si			2-si			3-si		
1									
2									
3									
N									
Jami									

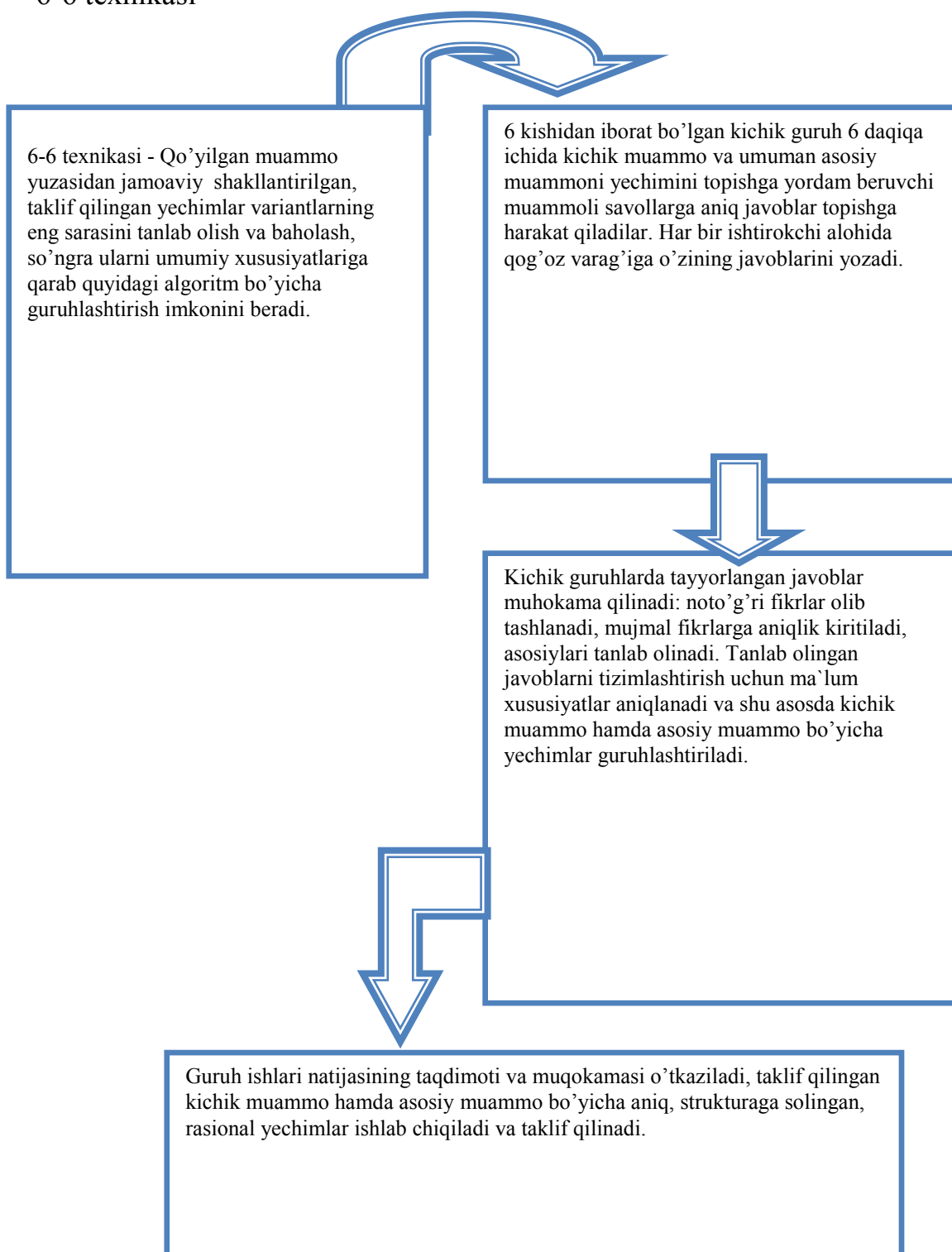
1.13-ilova



Usullari	Ijobiy tomonlari	Kamchiliklari

1.14-ilova

6-6 texnikasi



1.15-ilova

«Koop-koop» («Birgalikda o'qiymiz») texnikasini o'tkazish qoidasi

Faoliyat bosqichlari		
	Ta'lim beruvchi	Ta'lim oluvchi
1	Turli xil o'zlashtirish darajalariga ega bo'lgan, 3 va 5 kishidan iborat kichik guruhlarni tashkil qiladi.	
2	Har bir guruhga butun o'quv guruhi ishlashi lozim bo'lgan, mavzuning bir bo'lagi bo'lib hisoblangan, bitta topshiriq beradi. Ekspert varaqlarini tarqatadi.	Guruh a'zolari o'zaro topshiriqlarni bo'lib olishadi.
3	Topshiriqni muvaffaqiyatli bajarilishini va muammoli madaniyatini nazorat qiladi.	3.1 Hamma topshiriqni mustaqil bajaradi. 3.2 Guruh a'zolarining mini-ma'ruzalarini tinglaydi. Umumiy ma'ruza tayyorlanadi. Ma'ruzani o'qish uchun spiker tayinlanadi.
4	Guruhlarda ish yakunlanishini va ish natijasi taqdim etilishini e'lon qiladi.	
5	Natijalarni taxlil qiladi va baholaydi, g'olib guruhlarni aniqlaydi.	

Asosiy adabiyotlar

1. R. Hamdamov, U. Begimqulov, N. Tayloqov. Ta'limda axborot texnologiyalari. "O'zbekiston Milliy Ensiklopediyasi" Davlat ilmiy nashriyoti. Toshkent - 2010
2. S.S. G'ulomov, B.A. Begalov. Informatika va axborot texnologiyalari. Fan, T: 2010. 686 b.
3. Aripov M. M., Muxammadiyev J.U. Informatika. Axborot texnologiyalari. Oliy o'quv yurtlari talabalar uchun darslik. –Toshkent. 2005.
4. M.M. Aripov, T. Imomov va boshqalar Informatika. Axborot texnologiyalari. O'quv qo'llanma, 1-2-qism. -Toshkent. 2005, TDTU 334 b. va 394 b.
5. A.M. Polatov. Algoritmlar va C++ tilida dasturlash asoslari. O'quv qo'llanma. – Toshkent "Universitet" 2017. 58 b
6. Axborot kommunikatsiya texnologiyalari. O'quv qo'llanma. –Toshkent. 2012
7. Informatika va axborot texnologiyalari. O'quv qo'llanma. – Toshkent 2012

Qo'shimcha adabiyotlar

1. Mirziyoyev SH.M. Qonun ustvorligi va inson manfaatlarini ta'minlash-yurt taraqqiyoti va xalq farovonligining garovi. 48b. T.O'zbekiston. 2017 yil
2. Mirziyoyev SH.M. Erkin va farovon demokratik O'zbekiston davlatini birgalikda barpo etamiz. 56b, T.O'zbekiston. 2016 yil
3. Mirziyoyev SH.M. Buyuk kelajagimizni mard va olijanob xalqimiz bilan birga quramiz. 488b, T.O'zbekiston. 2017 yil
4. Aripov M.A., Maraximov A.R., Informatika, informatik texnologiyalar. Darslik., T: TDYUI., 2004 yil
5. S. S. G'ulomov, B.A. Begaliyev "Informatika va axborot texnologiyalari" Toshkent, FAN., 2010 yil.

Internet saytlari

1. Ensiklopediya poiskovix system [http:// www.searchingines.ru/](http://www.searchingines.ru/)
2. <http://moodle.edu.uz>
3. www.intuit.ru
4. <http://virlual-universily eurasia.org>
5. <http://www.opennet.ru>
6. www.linux.org.ru
7. <https://nashol.com>
8. https://www.youtube.com/video urok_Project_

