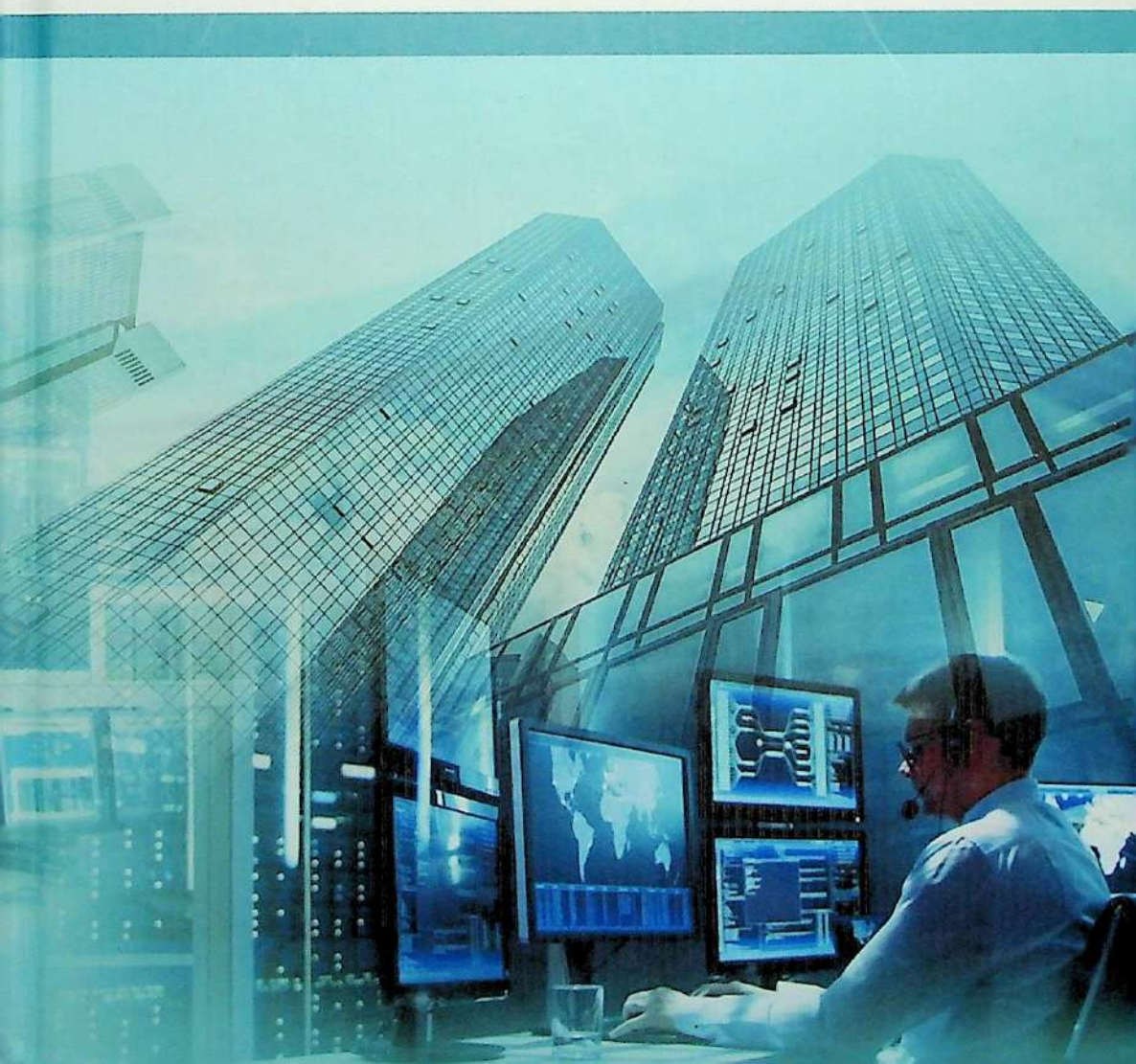


SH.A. AKBAROVA

TEXNIK TIZIMLARDA AXBOROT TEXNOLOGIYALARI



O'ZBEKISTON RESPUBLIKASI
OLIV VA O'RTA MAXSUS TA'LIM VAZIRLIGI

ISLOM KARIMOV NOMIDAGI
TOSHKENT DAVLAT TEXNIKA UNIVERSITETI

SH.A. AKBAROVA

TEXNIK TIZIMLARDA AXBOROT TEXNOLOGIYALARI

*O'zbekiston Respublikasi Oliy va o'rta maxsus ta'lim
vazirligi tomonidan texnik ta'lim yo'nalishlari
uchun darslik sifatida tavsiya etilgan*

TOSHKENT
«DONISHMAND ZIYOSI»

2022

TERMIZ MUHANDISLIK-
TEXNOLOGIYA INSTITUTI
AXBOROT-RESURS MARKAZI

INV. № 17718

• 27 • 05 2023 yil

UO'K 004(075)
KBK 32.81ya7
A 40

Taqrizchilar:

Qodirov M.M. – «Axborot texnologiyalari» kafedrası dotsenti,
PhD (TDTU);

Djumanov J.X. – «Kompyuter tizimlari» kafedrası professori,
texnika fanlari doctori (TATU).

Akbarova, Sh.A.

A 40 Texnik tizimlarda axborot texnologiyalari [Matn]: darslik /
Sh.A. Akbarova. – Toshkent: «Donishmand ziyosi», 2022. – 304 b.

Ushbu darslikda axborot-kommunikatsiya texnologiyalarining tarmoq sohalarda qo'llanilishi, intellektual boshqaruv tizimlari, avtomatlashtirilgan loyihalash tizimlari va ularning texnik sohalarda qo'llanilishi, imitatsion modellar yaratish, matematik modellash-tirish, «Komnac 3D» tizimida turli ko'rinishdagi modellar yaratish, «Bulutli texnologiya» modellari, texnik tizimlarda axborot xavf-sizligini ta'minlash usullari hamda zamonaviy dasturlash tillari va texnologiyalari, ulardan foydalanish xususiyatlari haqida ma'lumotlar berilgan.

ISBN 978-9943-8300-0-4

© SH.A. Akbarova, 2022

© «Donishmand ziyosi», 2022

KIRISH

Axborot texnologiyalarining hozirgi zamon taraqqiyoti hamda yutuqlari fan va inson faoliyatining barcha sohalarini axborotlashtirish zarurligini ko'rsatmoqda.

Jamiyatni axborotlashtirish deganda, axborotdan iqtisodni rivojlantirish, mamlakat fan-taxnika taraqqiyotini, jamiyatni demokratlashtirish va intellektuallashtirish jarayonlarini jadallashtirishni ta'minlaydigan jamiyat boyligi sifatida foydalanish tushuniladi.

Texnik tizimlardagi barcha muhandislik muammolari yangicha yondashishni talab etadi. Avtomatlashtirilgan loyihalash tizimlari, geometrik modellash dasturlari, tarmoqdagi ma'lumotlar bazalari ishlab chiqarish jarayonidagi va texnik tizimlardagi muammolarni yechishda samarali qo'llanilmoqda. Shuningdek, talabalarning yangi bilimlarni egallashda ta'lim jarayonida va mustaqil ta'limda axborot texnologiyalaridan unumli foydalanish va zamonaviy texnologiyalarni tatbiq etishi talab qilinadi.

Texnik boshqaruv tizimlarida va muhandislik masalalarini yechishda axborot texnologiyalarining o'rni katta. Jumladan, yo'nalishlar sohalaridagi masalalarni yechishda, mashinasozlikni loyihalashtirishda, mahsulot dizaynini ishlab chiqishda, sanoat mahsulotlarining tizimli tahlilida, tizim va kommunikatsiyani loyihalashda, muhandislik tahlilida, elektronika va energetika sohasida, mexanika-muhandislik sohasida loyihalash jarayonlarini avtomatlashtirishda, texnik tizimlarda ma'lumotlar almashish jarayonida zamonaviy dasturiy vositalardan va texnologiyalardan, shuningdek, internet texnologiyalaridan mukammal foydalanish kabilar katta ahamiyatga ega.

Ma'lumki, kompyuter texnologiyasining rivojlanishi bilan murakkab tizimlarni tadqiq qilishda mashinali modellash usuli eng samarali va bu usulning yordamisiz texnika va texnologiyaning ba'zi muammolarini hal etish qiyinlashib qoldi. Shu sababdan muhandis-texnologlarni tayyorlashdagi dolzarb masalalardan biri – MATLAB, MathCad, «Компас 3D» dasturlarida modellash nazariyasining asoslarini va usullarini o'zlashtirish hisoblanadi. Bu masala nafaqat o'rganilayotgan obyektlarning modellarini

qurish, ular dinamikasini tahlil qilish va model bilan mashinali eksperimentni boshqarish, balki o'rganilayotgan tizimlarga yaratilayotgan modellarning monandligini ma'lum miqdorda, qo'llanish chegarasida fikr yuritish mumkinligi hamda zamonaviy hisoblash texnika vositalarida tizimlarning modellashtirishni to'g'ri tashkil qilish imkonini beradi. Talabalarning «Texnik tizimlarda axborot texnologiyalari» fanini o'zlashtirishi uchun o'qitishning ilg'or va zamonaviy usullaridan foydalanish, yangi axborot va pedagogik texnologiyalarni tatbiq qilish muhim ahamiyatga egadir.

Texnik tizimlarda axborot texnologiyalari fanining vazifasi – ta'lim yo'nalishlari bo'yicha qo'llaniladigan avtomatlashtirilgan loyihalash tizimlari yordamida masalalarni yechish, modellashtirish, loyihalash va mahsulot dizaynini ishlab chiqish hamda dasturlashning mazmun-mohiyatini bilish, ularga nisbatan talabalarning ilmiy dunyoqarashini shakllantirish orqali axborot-kommunikatsiya texnologiyalari imkoniyatlarini namoyon etish. Shuningdek, hozirgi kun talabalarining yangi bilimlarni egallashida ta'lim jarayonida va mustaqil ta'limda axborot texnologiyalaridan unumli foydalana olishi va zamonaviy texnologiyalarni tatbiq etishi talab qilinadi.

Fanni o'rganish natijasida talabalar MATLAB, MathCad, «Компас 3Д» tizimlarda ishlashning mazmuni va mohiyati, ularni yaratish, kompyuter texnologiyalaridan foydalanish asoslarini hamda C++ Builder 6 dasturlash tili haqidagi bilimni egallab oladilar.

MATLAB dasturida imitatsion modellar yaratish tizimini talabalarga to'laqonli o'zlashtirishlari uchun muammoli metodlardan foydalanishni o'rganish, shu bilan birga MATLAB, Mathcad, «Компас 3Д» dasturida turli ko'rinishdagi modellarni yaratishning asosiy xossa va vazifalarining mazmun-mohiyatini o'rganish hamda uni nazariy va amaliy tadqiq etish muhim hisoblanadi.

Har bir mavzuni chuqur o'rganishda asosiy adabiyotlar bilan birgalikda internet saytlarining manzillari ham berilgan bo'lib, ushbu saytlardan talabalar o'zlariga kerak bo'lgan ma'lumotlarni olishlari va laboratoriya, amaliyot mashg'ulotlarida ham foydalanishlari mumkin.

1-BOB. TEXNIK TIZIMLARDA AXBOROT TEKNOLOGIYALARI FANINING PREDMETI VA VAZIFALARI

1.1. Axborot texnologiyalari va tizimlarining nazariy asoslari

Axborot texnologiyalari – bu ma'lumotlarni to'plash, yaratish, ishlov berish, tashkil etish, saqlash, izlash, tarqatish va foydalanishni ta'minlaydigan faoliyat hisoblanadi. Aynan jamiyat tomonidan talab qilinadigan ma'lumotlar (matn, raqamli, grafik, tovush, video, animatsiya) axborot texnologiyasining obyekti va natijasi bo'lib xizmat qiladi.

Axborot – shaxs va maxsus qurilmalar tomonidan predmetlar, faktlar, hodisalar va jarayonlarni aks etadigan ma'lumot hisoblanadi. Axborot signal (yorug'lik, tovush, elektr va boshqalar), axborot xabari (matn, grafik, nutq, vizual, audiovizual va boshqalar), rasmiylashtirilgan ma'lumotlar (belgilar, ko'rsatkichlar, parametrlar va boshqalar) shaklida mavjud bo'ladi. Axborot inson uchun mavjud bo'lgan har qanday usulda moddiy vositada qayd etilgan ma'lumotlarni hujjatlashtiradi.

Zamonaviy axborot ishlab chiqarish mustahkam moddiy bazaga tayanadi. Axborot ishlab chiqarishning texnik jihozlanishi rivojlangan axborot texnologiyalari sanoati tomonidan ta'minlanadi: kompyuterlar, maishiy elektronika, aloqa, ofis, bosmaxona, aloqa uskunalar, biznesni qo'llab-quvvatlash tizimlari, menejment, ta'lim va h.k.

Axborotni izlash, qayta ishlash va uzatishning kompyuter usulida bajarilishi uchun zarur shartlar yaratilgan.

Axborot texnologiyalari insonlarning samarali axborot ta'sirini, ularning dunyo axborot resurslaridan foydalanishini, axborot mahsulotlari va xizmatlariga bo'lgan ehtiyojlarini qondirishni ta'minlashga xizmat qiladi.

Axborot texnologiyalari – barcha turdagi ma'lumotlarni yaratish, ishlov berish, tashkil etish va tarqatish, zarur dasturiy ta'minot va texnik vositalarni ishlab chiqarish bilan bog'liq bo'lgan iqtisodiyotning bir bo'lagi hisoblanadi.

Shunday qilib, axborot texnologiyalari iqtisodiyotning tizimlashtirilgan tarmog'idir. U quyidagilarni o'z ichiga oladi:

- texnik vositalar (kompyuterlar, telefonlar, radiolar, magnitofonlar, musiqa markazlari, video va kinokameralar, televizorlar, pleyerlar va boshqalar);

- axborot tashuvchilar (fleshkalar, optik disklar, video, audio lentalar va boshqalar);

- telekommunikatsiya uskunalari va tarmoqlari (kabellar, simlar, sun'iy yo'ldoshlar, uzatish liniyalari, telekommunikatsiya tarmoqlari va boshqalar);

- matnli, grafik, tovushli, audiovizual, multimedia hujjatlari (ma'lumotlar bazalari va ma'lumotlar banklari, kutubxona va arxiv fondlari, hujjatlar bazalari, axborot qidirish tizimlari va boshqalar) shaklidagi ma'lumotlar;

- foydalanuvchilarga ma'lumotlarni boshqarish, katta hajmdagi ma'lumotlarga kirish va ular bilan ishlashga imkon beruvchi dasturiy ta'minot;

- axborot ishlab chiqarishni birlashtirishni, axborot tizimlari va tarmoqlari o'rtasidagi o'zaro aloqani, axborotni ruxsatsiz kirish, buzilish yoki yo'q qilishdan himoya qilishni ta'minlaydigan standartlar, internet protokollari va shartnomalari.

«Texnologiya» va «Axborot» tushunchalarining birligi, avvalo, ikkalasi ham belgilangan maqsadga erishishga qaratilgan harakatlarning ma'lum bir to'plami sifatida tushuniladigan jarayonga asoslanganligidadir.

Bundan tashqari, har qanday texnologik jarayon inson tanlagan va har xil usul va vositalar kombinatsiyasi yordamida amalga oshiriladigan strategiya bilan belgilanishi kerak.

Axborot texnologiyalari vositalari – bu axborot texnologiyalari amalga oshiriladigan texnik, dasturiy ta'minot, axborot va boshqa

vositalar. Zamonaviy axborot texnologiyalari sanoat texnologiyalari bilan umumiy xususiyatlarga ega.

Umuman olganda, axborot texnologiyalarining asosiy xususiyatlarini ajratib ko'rsatish mumkin:

- axborot texnologiyasi jarayonining maqsadi axborot olish;
- texnologik jarayonning predmeti (ishlov berish predmeti) ma'lumotlar hisoblanadi;
- texnologik jarayonni amalga oshiradigan vositalar – bu turli xil hisoblash tizimlari (dastur, apparat, dasturiy ta'minot va uskunalar);
- tanlangan predmet sohasiga muvofiq ma'lumotlarga ishlov berish jarayonlari operatsiyalarga bo'linadi;
- jarayonlar bo'yicha nazorat harakatlari tashkilotning boshqaruv xodimlari tomonidan amalga oshiriladi;
- axborot texnologiyasi jarayonining maqbulligi mezonlari foydalanuvchilarga axborotni o'z vaqtida yetkazib berish, uning ishonchligi, xavfsizligi va to'liqligi hisoblanadi.

Axborot texnologiyalari axborot resurslaridan maqsadga muvofiq foydalanishga va ularni tashkiliy tuzilmaning barcha elementlariga yetkazib berishga qaratilgan.

Avtomatlashtirilgan axborot texnologiyalarining asosiy maqsadi boshlang'ich ma'lumotlarga ishlov berish orqali yangi sifatli ma'lumotlarni olish, buning asosida optimal boshqaruv qarorlari ishlab chiqishdan iborat.

Axborot texnologiyalarining asosiy maqsadi quyidagilar orqali amalga oshiriladi:

- axborot integratsiyasi;
- ma'lumotlarning dolzarbligi va izchilligini ta'minlash;
- boshqaruv apparati faoliyatini axborot bilan qo'llab-quvvatlashning sifat jihatidan yangi shakllarini joriy etish va faoliyat yuritish uchun zamonaviy texnik vositalardan foydalanish.

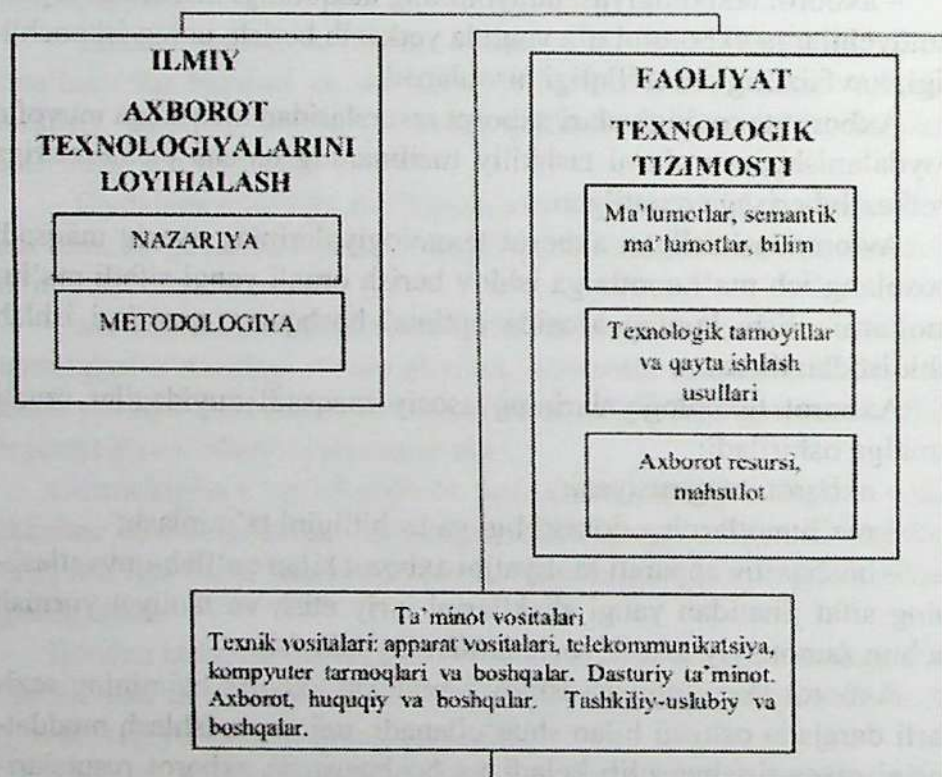
Axborot texnologiyasi ishlov beradigan axborot hajmining sezilarli darajada oshishi bilan shug'ullanadi, uni qayta ishlash muddatlarini qisqartirishga olib keladi va boshqaruvda axborot resurslaridan foydalanish jarayonining eng muhim tarkibiy qismi hisoblanadi.

Inson faoliyatining turli sohalari resurs ta'minoti tarkibida axborot komponentining ahamiyati ortib bormoqda. Axborot muhiti ijtimoiy va ekologik muhit bilan bir qatorda insonning yangi yashash muhitiga aylanadi.

Axborot texnologiyalari odamlar o'rtasida axborot almashinuvini ta'minlashda hal qiluvchi rol o'ynaydi; jamiyatni intellektualizatsiya qilishning texnologik asosi, ta'lim va madaniyat tizimini rivojlantirish, yangi bilimlarni olish va to'plash jarayonlarida muhim rol o'ynaydi.

Axborot texnologiyasi ilm-fan va axborotni ishlov berish bo'yicha maqsadli faoliyat texnologik va ta'minot qismlariga bo'linadi.

AXBOROT TEXNOLOGIYALARI



1.1-rasm. Axborot texnologiyalari modeli.

Axborot texnologiyasi amaliy faoliyat texnologiyasining barcha xususiyatlariga ega va o'z maqsadlari, mavzusi, resurslari, jarayonlari va natijalari bilan tavsiflanadi.

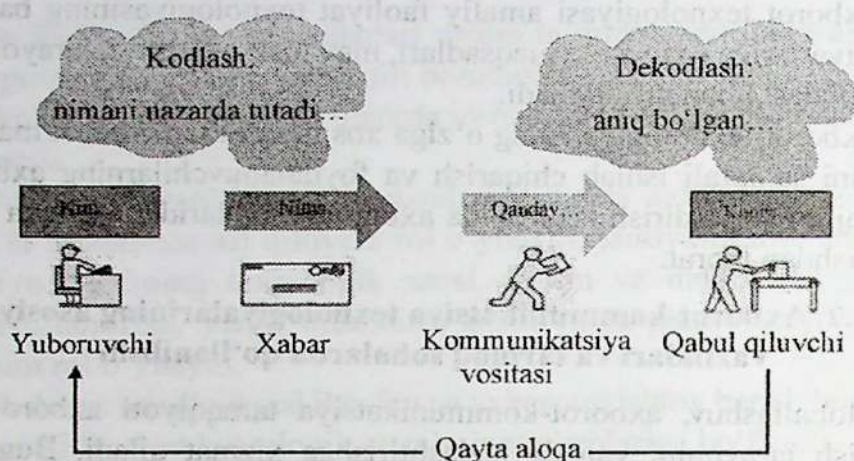
Axborot texnologiyasining o'ziga xos maqsadi – axborot mahsulotlarini samarali ishlab chiqarish va foydalanuvchilarning axborot ehtiyojlarini qondirish jarayonida axborot resurslaridan oqilona foydalanishdan iborat.

1.2. Axborot-kommunikatsiya texnologiyalarining asosiy vazifalari va tarmoq sohalarda qo'llanilishi

Globalashuv, axborot-kommunikatsiya taraqqiyoti axborot almashish jarayonini yanada tezlashtirishga xizmat qiladi. Bugungi kunda maktab o'quvchisi, hatto bog'cha bolasi qo'lida ham mobil telefon borligi hech kimni ajablantirmay qo'ydi. Uyali aloqa vositalari, planshet, kompyuter texnologiyalari, internetning imkoniyatlari had-hududsiz ekanligi hammaga ayon. Kundalik turmushimizda oddiy voqelikka aylangan ana shunday zamonaviy texnologiyalar imkoniyatlaridan jamiyatimizning barcha a'zolari, jumladan, jismoniy va yuridik shaxslar unumli foydalanishga intilmoqda.

Texnik sivilizatsiyaning rivojlanish masalasini hal etishda axborotlashtirishni kompyuter va kommunikatsiya texnologiyalari asosida olib borishdan boshqa muqobil yo'l yo'q. Axborot eng muhim va bebaho resurs bo'lgan axborotlashgan jamiyatda mamlakatning rivojlanish darajasi uni axborotlashtirish darajasi bilan baholanmoqda. Shuni ta'kidlash mumkinki, hozirgi kunda axborot yurituvchi kuchning asosiy mahsuloti sifatida qaraladi, axborot texnologiyalari esa jamiyatning axborot resurslarini faollashtirish va samarali foydalanishning muhim vositasidir. Shuning uchun AKT (axborot-kommunikatsiya texnologiyalari)ni jadal rivojlantirish va takomillashtirish hozirgi kunda mamlakat va jahon miqyosida strategik jihatdan muhim masalaga aylandi.

Kommunikatsiya – bu insonlar o'rtasida og'zaki va yozma xabarlar, nutq parametrlari ko'rinishidagi muloqotni va ma'lumotlarni uzatish jarayoni.



1.2-rasm. Kommunikatsiyaning uzatilishi.

Axborot-kommunikatsiya texnologiyalari – bu shaxsiy, ommaviy yoki ishlab chiqarish aloqalari uchun zarur bo‘lgan ma’lumotlarni tayyorlash va uzatish jarayonida ishlatiladigan obyektlar, harakatlar va qoidalar majmuyi.

Bunday texnologiyalarning maqsadi odamlarga ma’lum qarorlar qabul qilish uchun axborot bazasini taqdim etishdir.

Qayta ishlanayotgan axborot turiga qarab, aloqa tizimlari ma’lum ma’lumotlarni qayta ishlash uchun mo‘ljallangan bo‘lishi mumkin (masalan, algoritmik tillar yoki dasturlash tizimlari). Masalan: ishlov berish uchun aloqa tizimlari;

- matnli ma’lumotlar (masalan, gipermatnli tizimlar);
- grafikalar (bularga vektorli grafikani qayta ishlash uskunolari kiradi);
- animatsiya, shu jumladan video tasvirlar va ovoz (multimedia ilovalarini yaratish tizimlari);
- bilim.



1.3-rasm. Axborot-kommunikatsiya tizimlarining turlari.

Axborot-kommunikatsiya texnologiyalari o‘zaro ta’sir darajasiga qarab tasniflanadi. Ular maxsus texnik vositalar yordamida amalga oshirilishi mumkin. Media darajasida o‘zaro ta’sir o‘tkaziladigan tizimlar va axborotni qayta ishlash va saqlash tushunchalari qo‘llaniladigan tizimlar mavjud.

1.3. Kompyuter aloqa tizimlari

Tizim (sistema) deganda, yagona maqsad yo‘lida bir vaqtning o‘zida ham yaxlit, ham o‘zaro bog‘langan tarzda faoliyat ko‘rsatuvchi elementlar (obyektlar) majmuasi tushuniladi. Demak, har qanday tizim biror-bir aniq maqsad yo‘lida xizmat qiladi. Masalan, sizga ma’lum bo‘lgan shahar telefon tarmoqlari tizimi, insondagi yurak-qon tomir tizimi, asab tizimi va boshqalar sun’iy yaratilgan va tabiiy tizimlarga misol bo‘la oladi. Ularning har biri tizimga qo‘yiladigan barcha shartlarga javob beradi, ya’ni har biri o‘ziga xos yagona maqsad yo‘lida faoliyat ko‘rsatadi va tizimni tashkil etuvchi elementlardan iborat.

Axborot texnologiyalarini yangi tarmoq jamiyatini boshqarish tizimini tartibga solish, saqlash va takomillashtirishga qaratilgan axborot jamiyatining elementi va vazifasi sifatida qarash mumkin.

Zamonaviy axborot texnologiyalarining asosiy xarakteristikasi – raqamli texnologiyalar orqali uzoq masofalarga kerakli formatdagi ma’lumotlarni saqlash va uzatish uchun kompyuter texnologiyalaridan keng foydalanish.

Axborot texnologiyalari yangi aloqa turlarining paydo bo‘lishiga sabab bo‘ldi: SMS-xabarlar, ijtimoiy tarmoqlar, Skype orqali mulo-

qot, intranet va boshqalar. Axborot texnologiyalari kompyuter tarmoqlari bilan uzviy bog'liqdir.

Kompyuter aloqasi tizimlari bir qator vazifalarni hal qilish uchun mo'ljallangan, shu jumladan yuqori tezlikdagi tarmoqlardan foydalanish orqali yuqori sifatli aloqani ta'minlashdan iborat. Axborot texnologiyalari ko'pchilik ta'lim muassasalari amaliyotining bir qismiga aylandi. Axborot-kommunikatsiya texnologiyalari yordamida, multimediali ilovalardan foydalanib, o'qituvchilar taqdimotlar va tajribalar o'tkazadilar.

Ta'limda ishlatiladigan axborot-kommunikatsiya texnologiyalarining yana bir aniq afzalligi – bu moddiy resurslarni hisoblash algoritmlari bilan birlashtirishdir. Bu esa, o'z navbatida, murakkab muammolarning soddaga yechimlarini topishga imkon beradi.

Har xil turdagi axborot kommunikatsiyalarining kombinatsiyasi ma'lum bir axborot makonini yaratadi. Axborot makoni – bu axborot kommunikatsiyalari tarmog'i yetarlicha rivojlangan jamiyat sohasi hisoblanadi.

Ma'lumotni bitta mavzudan uzatish usuli aloqa taraflari uchun umumiy bo'lgan signal tizimining mavjudligi bilan belgilanadi (aloqa tili, belgilar tizimi): og'zaki (og'zaki muloqot shunday paydo bo'ladi) yoki og'zaki bo'lmagan – rasmlar, grafikalar, statistik ma'lumotlar, imo-ishoralar, yuz ifodalari va boshqalar.

1.4. Raqamli texnologiyalar tushunchasi va uning ahamiyati

Raqamli texnologiya virtual olamdan obyektlarni haqiqiy dunyoga qo'shish imkonini beruvchi kengaytirilgan haqiqat texnologiyasi hisoblanadi.

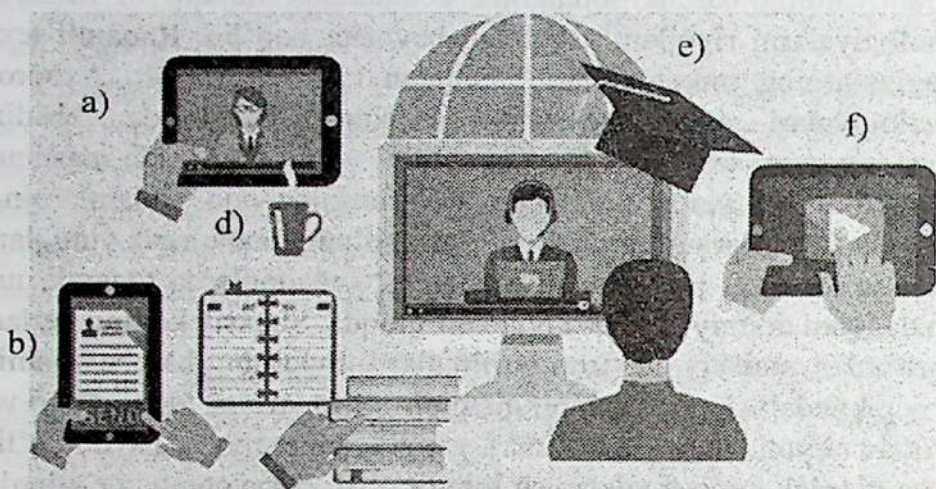
Raqamli yechimlar – uzluksiz spektr sifatida emas, balki analog darajadagi diskret diapazonlarda signallarni namoyish etishga asoslangan texnologiyalardir. Ushbu texnologiyalarning barcha darajalari tarmoqli ichida bir xil signal holatini ifodalaydi.

Raqamli texnologiyalar – bu axborot ma'lumotlarini kodlash va efirga uzatish usullariga asoslangan, nisbatan qisqa vaqt ichida turli xil muammolarni hal qilishga imkon beruvchi diskret tizim.

Jamiyatning barcha sohalarida kelajakdagi taraqqiyot raqamli texnologiyalarni rivojlantirish bilan bevosita bog'liq. Raqamli texnologiyalarning imkoniyatlari haqiqatan ham cheksizdir. Axborot texnologiyalari raqamli texnologiyalarning tarmoqlaridan biridir. Ularning yuqori tezligi va ko'p qirraliligi ularni nihoyatda mashhur qiladi. Biznes va sanoat ishlab chiqarish sohasida, oddiy odamlarning kundalik ehtiyojlarida hamma joyda eng so'nggi texnologik yutuqlardan foydalaniladi. Har qanday sohada va xonadonlarda raqamli qurilmalar soni doimiy ravishda o'sib bormoqda. Shaxsiy kompyuterlar, smartfonlar, maishiy elektron qurilmalar – bularning barchasi jamiyatning kundalik hayotining bir qismidir. Olimlarning tadqiqotlari va hayot amaliyoti shuni ko'rsatadiki, yangi texnologiyalar paydo bo'lishi va ulardan keng foydalanish tobora o'sib borayotgan sur'atlarda rivojlanib bormoqda. Masalan, yigirmanchi asrda elektr energiyasini joriy etish taxminan o'ttiz yil davom etdi va aytaylik, planshetlar deyarli to'rt yilda hamma joyda ishlatilgan. Raqamli texnologiyalar analog texnologiyalardan farqli o'laroq uzluksiz signallar bilan emas, balki diskret ishlaydi.

Raqamli texnologiyalar asosan raqamli hisoblash elektronikasi-da, birinchi navbatda, kompyuterlarda, o'yin mashinalari, robototexnika, avtomatika, o'lchash uskunolari, radio va telekommunikatsiya uskunolari va boshqa ko'plab raqamli qurilmalar kabi elektrotexnikaning turli sohalarida qo'llaniladi.

Hozirgi vaqtda raqamli tibbiyot texnologiyasidan ham keng foydalanilmoqda. Raqamli tibbiyot texnologiyasi aslida ko'plab odamlarning hayotini saqlab qolmoqda. Bugungi ilmiy izlanishlar yuqori texnologiyali qurilmalarni amaliyotga tatbiq etishga imkon berdi, ularning yordamida ko'plab kasalliklarni tashxislash, tahlil qilish va davolash ishlari olib borilmoqda. Klinik tadqiqotlar yangi dori vositalarini yaratish uchun keng imkoniyatlar yaratadi. Farmakologiya va boshqa sohalarining uslubiy asoslarini takomillashtirish o'lim darajasini pasaytiradi va turmush darajasini oshiradi. Bundan tashqari, ta'lim tizimida ham raqamli texnologiyalardan keng foydalanilmoqda.



1.4-rasm. Ta'limda raqamli texnologiyaning qo'llanilishi: a) planshet; b) mobil aloqa; d) elektron kutubxona; e) televizor; f) YouTube.

Raqamli tizimlarning kamchiliklari: ba'zan raqamli tizimlar xuddi shu vazifalar uchun analog tizimlarga qaraganda ko'proq quvvat sarf qiladi.

Bunday holda, ko'proq issiqlik energiyasi chiqariladi, bu qurilmaning sovishini talab qiladi (masalan, sovitish ventilyatori o'rnatiladi). Bu batareyalar bilan ishlaydigan qurilmalarda bunday tizimlardan foydalanishni cheklaydi.

Ba'zan, agar bitta raqamli ma'lumot yo'qolsa, xabarning ma'nosini to'liq o'zgartirish mumkin. Raqamli texnologiyalarni qo'llash sohalari internetda har kim topishi mumkin bo'lgan juda katta hajmdagi axborot oqimlari o'quv jarayonini bunday istagi bo'lgan har bir kishiga taqdim etdi.

Bugungi kunda keng tarqalgan virtual aloqa qisqa vaqt ichida kasallikni masofadan turib aniqlashga imkon beradi. Protezlarni ishlab chiqarishni zamonaviy 3D printerlarda amalga oshirish mumkin, bu ham katta imkoniyatlarni ochib beradi.

Doimiy ravishda o'sib borayotgan ishlab chiqarish hajmi, boshqa narsalar qatori, sayyoramiz aholisining o'sishi bilan bog'liq bo'lib,

ayrim sohalarda ustuvor vazifalardan biriga aylandi. Va raqamli texnologiyalar sanoatdagi deyarli barcha texnologik jarayonlarni avtomatlashtirish va optimallashtirishga qodir.

1.5. Raqamli iqtisodiyot tushunchasi va uning ahamiyati

Raqamli iqtisodiyot nima? Raqamli iqtisodiyot – bu iqtisodiy, ijtimoiy va madaniy aloqalarni raqamli texnologiyalarni qo'llash asosida amalga oshirish tizimidir. Ba'zida u internet iqtisodiyoti, yangi iqtisodiyot yoki veb-iqtisodiyot degan terminlar bilan ham ifodalanadi.

Raqamli iqtisodiyotni joriy etishda va rivojlantirishda axborot texnologiyalarining o'rni katta. Raqamli iqtisodiyot bilan aloqa o'rnatishda va undan foydalanishda turli ko'rinishdagi dasturlash tillarida tuzilgan dasturlar kerak bo'ladi.

Raqamli iqtisodiyot – bu noldan boshlab yaratilishi lozim bo'lgan qandaydir boshqacha iqtisodiyot emas. Bu yangi texnologiyalar, platformalar va biznes modellari yaratish va ularni kundalik hayotga joriy etish orqali mavjud iqtisodiyotni yangicha tizimga ko'chirish deganidir.

Belgilari:

- yuqori darajada avtomatlashtirilganlik;
- elektron hujjat almashinuvi;
- buxgalterlik va boshqaruv tizimlarining elektron integratsiyalashuvi;
- ma'lumotlar elektron bazalari;
- CRM (mijozlar bilan o'zaro munosabat tizimi) mavjudligi;
- korporativ tarmoqlar.

Qulayliklari:

- to'lovlar uchun xarajatlar kamayadi (masalan, bankka borish uchun yo'lkira va boshqa resurslar tejiladi);
- tovarlar va xizmatlar haqida ko'proq va tezroq ma'lumot olinadi;
- raqamli dunyodagi tovar va xizmatlarning jahon bozoriga chiqish imkoniyatlari katta;
- iste'molchi fikrini tez olish hisobiga tovar va xizmatlar jadal takomillashtiriladi;

- tezroq, sifatliroq, qulayroq.

Raqamli iqtisodiyot insonlarning turmush darajasini sezilarli darajada oshiradi, bu uning asosiy foydasidir.

Davlatning raqamli iqtisodiyotni rivojlantirish yo'lini tanlaganligi axborot texnologiyalari sohasida va umuman, elektron hujjatlar aylanmasi sohasida yangi yo'nalishlar ochib beradi. «Raqamli texnologiyalar» tomon burilishga butun jahon internet tarmog'i va sifatli aloqaning rivojlanishi sababchi bo'ldi.

Raqamli iqtisodiyotning zamonaviy ko'rinishi sifatida mobil to'lov tizimlarini ko'rsatish mumkin. Bunda hammamizga ma'lumki, mijozlarning smartfonlari bankdagi hisob raqamlarga bog'langan bo'ladi. Bu esa mijozlarga onlayn xaridlarni amalga oshirish, mobil qurilmalar yordamida pul mablag'larini erkin o'tkazish imkonini beradi.

To'lov tizimlari va bank axborot tizimlari raqamli aktivlardan keng foydalanadi. Raqamli aktivlarning asosiy komponentlari sifatida intellektual mulk obyektlari hisoblanadi. Intellektual faoliyat natijalariga egalik huquqiga faqatgina kompaniya ega bo'ladi. Bunday aktivlarga misol qilib, ixtiro patentlar, kompyuter dasturlariga mutlaq huquq va ma'lumotlar bazasini keltirish mumkin.

RAQAMLI IQTISODIYOT

9 yo'nalish:



1.5-rasm. Raqamli iqtisodiyotning yo'nalishi bo'yicha turlari.

Raqamli iqtisod axborot texnologiyalaridan korxonalarning turli bo'linmalarini boshqarishda foydalanish ishlab chiqarishni tashkil qilishni takomillashtirishga imkon beradi. Qisqa vaqt ichida ko'proq mahsulot chiqarish orqali ishlab chiqarish korxonalari deyarli barcha mamlakatlarda o'z mahsulotlarini sotishlari mumkin.

Mumkin bo'lgan chegaralarni oshirib, raqamli texnologiyalar iqtisodiy o'sishni tezlashtirishga yordam beradi. Sanoat sohasida ishlash uchun zarur bo'lgan inson resurslarini qisqartirish ma'naviy va madaniy rivojga yo'naltirish imkonini yaratadi.

Sanoat kompaniyalari axborotni boshqarish va uzatish sohasida turli xil ma'lumot va raqamli texnologiyalarni o'zlashtirish darajalariga ega. Raqamli texnologiyalar sanoat ishlab chiqarishi va tadbirkorligini rivojlantirishning asosiy yo'nalishi ekanligi allaqachon isbotlangan.

Raqamli texnologiya har qanday kompaniyada ishlab chiqarish jarayonlarini avtomatlashtirish, aniq statistik ma'lumotlarning haqiqiy ma'lumotlariga asoslangan moliyaviy buxgalteriya hisobini yuritishga imkon beradi. Optimal boshqaruv usullaridan foydalanish ishlab chiqarish jarayonlarini diversifikatsiya qiladi va oqilona qarorlar ishlab chiqishga ko'maklashadi.

Bugungi kunda biznesni qurish modellari sezilarli darajada o'zgarib bormoqda. Har bir yirik kompaniya global internet imkoniyatlaridan foydalangan holda o'z faoliyat doirasini sezilarli darajada kengaytirishi mumkin. Sayyoradagi turli xil geografik joylarga deyarli bir zunda kirish imkoniyati sizning biznesingizni maksimal darajada samarali boshqarish imkonini beradi.

Raqamli axborot texnologiyalari sohasidagi investitsiya sarmoyalari savdo bozorlari va odamlarning dolzarb ehtiyojlari to'g'risida real baho olishga imkon beradi.

Raqamli iqtisodiyotning afzalliklari – axborot-kommunikatsiya texnologiyalarining rivojlanishi, zamonaviy texnologiyalarning hayotimizga tatbiq etilishi har bir inson hayotida ko'plab ijobiy imkoniyatlar berishi mumkin. Raqamli texnologiyalar rivojlanishi ortidan inson, unga kerakli xizmatdan tezroq foydalanishi, internet orqali o'ziga

kerakli mahsulotlarni arzon sotib olish bilan ko'plab pul mablag'larini tejashi mumkin. Masalan, biron-bir kitobning elektron ko'rinishda sotib olish sizga, shu kitobni chop etilgan ko'rinishini sotib olish ancha arzonga tushishi mumkin. Yoki bo'lmasa oddiy iste'molchi o'zi ham tadbirkor bo'lishi, uyidan chiqmagan holda onlayn savdo-sotiq bilan shug'ullanishi mumkin.

Raqamli iqtisodiyotning rivojlanishini boshqa afzalliklari quyidagicha bo'lishi mumkin:

- ishlab chiqarishda mehnat samaradorligining oshishi;
- kompaniyalar raqobatbardoshligining o'sishi;
- ishlab chiqarishdagi xarajatlarning kamayishi;
- yangi ish o'rinlari yaratilishi;
- yangi zamonaviy kasblarning paydo bo'lishi;
- kambag'allikni yengish va ijtimoiy tengsizlikning yo'qolishi.

Bular raqamli iqtisodiyotning bor-yo'g'i bir nechta afzalliklari, xolos. Raqamli iqtisodiyotning rivojlanishi bizning kundalik hayotimizga ijobiy ta'sir qiladi, oddiy foydalanuvchiga ko'plab qo'shimcha imkoniyatlar beradi va qolaversa, bozorning o'sishi va rivojlanishini ta'minlab berishi mumkin.

Nazorat savollari

1. Axborot texnologiyalari tushunchasiga ta'rif bering.
2. Axborot-kommunikatsiya texnologiyalari deganda nimani tushunasiz?
3. Axborot tizimi qanday ketma-ketliklardan tashkil topgan?
4. Raqamli iqtisodiyotning qulayliklariga ta'rif bering.
5. Raqamli texnologiyaning afzalliklari qanday?

2-BOB. INTELEKTUAL TIZIMLARNING ASOSIY TUSHUNCHALARI

2.1. Intellektual tizimlarning maqsadi, funktsiyalari va uslubi

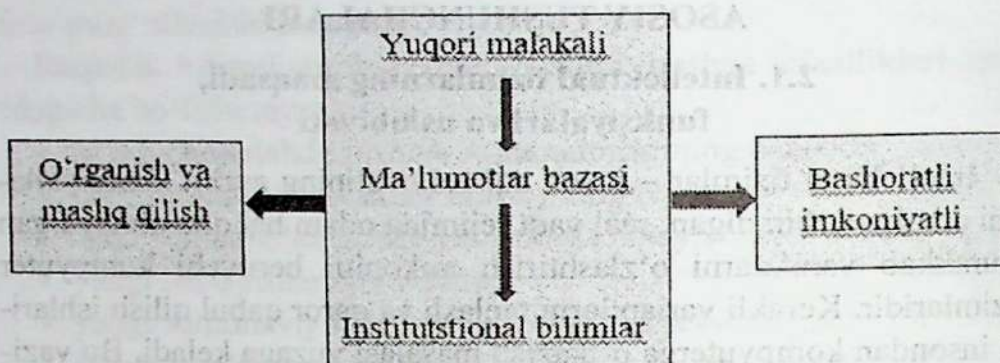
Intellektual tizimlar – inson aql-zakovatining ayrim xususiyatlarini amalga oshiradigan, real vaqt rejimida odam hal qila olmaydigan murakkab vazifalarni o'zlashtirish imkonini beruvchi kompyuter tizimlaridir. Kerakli variantlarni tanlash va qaror qabul qilish ishlari insondan kompyuterga o'tkazish masalasi yuzaga keladi. Bu vazifani yechish yo'llardan biri – bu ekspert tizimlarini yaratish va foydalanish sanaladi. Ekspert o'zidan kelib chiqib sharoitni tahlil etadi va nisbatan foydali axborotni aniqlab oladi, chorasiz yo'llardan voz kechgan holda qaror qabul qilishning eng maqbul yo'llarini vujudga keltiradi. Intellekt tizimlarni yaratishdagi markaziy muammo mutaxassislarning bilimlarini kompyuter xotirasida xuddi shunday aks ettirishdir. Ekspert tizimida ma'lum bir predmet sohasini ifodalaydigan bilimlar bazasidan foydalaniladi.

Ekspert tizimini yaratishdagi eng murakkab masalalardan biri bilimlar bazasi tizimini qurish va uni to'ldirish hisoblanadi. Bu ikki bosqich o'zaro bir-biriga bog'liq hisoblanadi.

Bilimlar bazasi – bu ayrim predmet sohalari murakkab vazifalar yechimini topish uchun tahlil va xulosalarni yuzaga keltiruvchi model, qoida, omillar (ma'lumotlar) majmuyidir.

Axborot ta'minotining alohida yaxlit strukturasi ko'rinishida yaqqol ko'zga tashlangan va tashkil etilgan predmet sohasi haqidagi bilim boshqa bilim turlaridan, masalan, umumiy bilimdan ajralib turadi. Bilimlar bazasi asosiy ekspert tizimi sanaladi. Bilimlar fikrlash va vazifalarni hal etish usuliga imkon beruvchi aniq ko'rinishda ifodalanadi va qaror qabul qilishni soddalashtirishga ko'maklashadi.

Ekspert tizimining asosligini ta'minlovchi bilimlar bazasi tashkilotning bo'linmalaridagi mutaxassislar bilimini, tajribasini o'zida mujassamlashtiradi va institutsional bilimlarni (ixtisoslashganlar majmuyini, yangilanayotgan strategiyalar, qarorlar uslublari) ifodalaydi (2.1-rasm).



2.1-rasm. Bilimlar bazasining asosiy xususiyatlari.

Ekspert tizimi – ayrim sohalarda bilimlarni to'plash va qo'llash, uyushtirish usullari hamda vositalari majmuyidir. Ekspert tizimi mutaxassislarning yuqori sifatli tajribasiga suyanagan holda qarorni tanlash chog'ida muqobil variantlar ko'pligi uchun yanada yuqori samaraga erishadi. Strategiyaning tuzish paytida yangi omillarni baholab, ularning ta'sirini tahlil etadi. Ekspert tizimlari sun'iy intellektdan foydalanishga asoslangan.

Sun'iy intellekt – kompyuterda inson tafakkurini simulatsiya qiluvchi dasturiy ta'minot tizimidir. Sodda qilib aytganda, sun'iy intellekt kompyuterlarni odam kabi o'ylash va yechim topishga yo'naltirilgan texnologiya.

Ushbu maqsadga erishish uchun uchta asosiy komponent talab qilinadi:

- hisoblash tizimlari (katta quvvatli kompyuterlar);
- katta ma'lumotlar va ularning boshqaruvi;
- SI algoritmlari (dasturiy kod).

Natija inson aqliga qancha yaqin bo'lishi istalsa, shuncha ko'p ma'lumotlar va hisoblash resurslari (kuchi) talab etiladi.

2.2. Ekspert tizimlari. Ekspert tizimlarining sinflanishi va struktura tuzilishi

Ekspert tizimlari (ET)ning asosililigini qarab chiqamiz.

2.1-jadval

Inson tafakkurini va sun'iy intellekt tizimini solishtirish

Inson tafakkur tizimi	Sun'iy intellekt tizimi
Kamchiliklari 1. Barqaror emas 2. Qiyin o'tkazuvchi (ifodalovchi) 3. Qiyin hujjatlashtiriluvchi 4. Oldindan aytib bo'lmaydigan 5. Qimmatli	Ustunliklari 1. Doimiy 2. Oson ifodalovchi 3. Oson hujjatlashtiriluvchi 4. Doim bir xil 5. Maqbul
Ustunliklari 1. Ijod qiluvchi 2. Moslashuvchan 3. Hissiy idrokdan foydalanadi 4. Har tomonlama 5. Keng qamrovli bilimdan foydalanadi	Kamchiliklari 1. Sun'iy oldindan dasturlashtiriladi 2. Aytib turish kerak 3. Belgili idrokdan foydalanadi 4. Tor yo'nalishli 5. Maxsus bilimdan foydalanadi

Ushbu tizimlarning afzalliklari va kamchiliklari tahlil qilinganda, inson ekspertning asosiy afzallik tomoni, u ko'p sohada, masalan, ijodkorlikda, topqirlikda, ma'lumot uzatishda va umuman, mazmunan Sldan ustunlikka ega.

Oddiy axborotni qayta ishlovchi dasturlardan ETning farqli tomonlari:

1) an'anaviy kompyuter dasturlari – ixtiyoriy qo'yilgan masalaga ular doim bir xil jarayonlar ketma-ketligida yondashishadi; ET har bir qo'yilgan masalaga xususiy yechim daraxtini quradi;

2) ET ixtiyoriy simvolli ifodani (masalan, konseptual, makon va zamon munosabatlari) qayta ishlaydi. Agar oddiy dasturlarda maqsad – sonli qiymatlarni hisoblash, o'zgarmaslarni to'plash va xotiradan chiqarish bo'lsa, ET uchun maqsad – obyektlar va hodisalar oqimini kuzatishda oldindan asosli ko'rsatmalar va tavsiyalar berishdan iborat;

3) agar an'anaviy dasturlar matematik qoidalaridan kelib chiqsa, ET ishlashi esa belgilarni qayta ishlashga va evristik mulohazalarga asoslanadi. ET inson tafakkurini, aniqlangan muammolarning faraz qilingan yechimlarini imitatsiya qiladi, keyin ulardan eng mos, to'g'ri keladiganini tanlaydi. Bunda eng oldin keraksiz yechimlarni tashlab yuboriladi. Bundan-da ko'proq, u egallangan subyektiv bilimlarga bog'liq bo'lmagan tarkibiy tuzilishidan foydalanadi, tadqiq qilingan inson tizimining hayotiy muammolar yechimiga ekspertiza o'tkazilishi qabul qilinadi. Muammolarni turli tomondan qarashning tizimli tahlili tufayli, u shundaygina to'g'ri keladigan emas, balki eng yaxshi yechimni beradi. ET butunlay insoniy ekspertizaga bog'liq.



2.2-rasm. Ekspert tizimlarining strukturasi.

Ekspert tizimlarini axborot tizimlari sinfi sifatida ko'rib chiqish mumkin. U foydalanuvchining roziligidan qat'i nazar ma'lumotlarni tahlil va tahrir eta oluvchi, qarorni tahlil etib qabul qiladigan, tahliliy-tasnifiy vazifalarni bajara oladigan ma'lumotlar va bilimlar bazasiga ega. Jumladan, ekspert tizimlari keladigan axborotlarni guruhlariga bo'lib tashlay oladi, xulosa chiqaradi, identifikatsiyalaydi, tashxis qo'yadi, bashoratlashga o'rgatadi, sharhlab beradi va hokazo.

Ekspert tizimining boshqa axborot tizimlaridan afzalliklari quyidagicha:

- yaqin davrlargacha kompyuterda yechish qiyin yoki umuman yechib bo'lmaydigan deb sanaluvchi murakkab masalalarning yangi sinfini yechish, optimallashtirish va (yoki) bahosini olish imkoniyati;

• dasturchi bo'lmagan foydalanuvchiga (pirovard foydalanuvchilar) o'z tilida suhbat yuritish va kompyuterdan samarali foydalanish uchun axborotni vizualizatsiyalash usullarini qo'llash imkoniyatini ta'minlash;

• yanada ishonchli va malakali xulosa chiqarish yoki qaror qabul qilish uchun ekspert tizimini mustaqil o'rganish, bilimlardan foydalanish qoidalari, ma'lumotlar, bilimlarning to'planishi;

• foydalanuvchi axborot yo'qligi tufayli yoki axborotning haddan ziyod rang-barangligi, yoki hatto kompyuter yordamida ham odatdagi qarorni qabul qilishning cho'zilib ketilishi tufayli yecha olmaydigan savollar yoki muammolarni hal etish;

• takomillashgan asboblari va ushbu tizimdagi foydalanuvchi mutaxassisning shaxsiy tajribasidan foydalanish hisobiga yakka tartibdagi ixtisoslashgan ekspert tizimlarini yaratish imkoniyati;

• ekspert tizimining asosi qaror qabul qilish jarayonini shakllantirish maqsadida tuzilgan bilimlar majmuyi (bilimlar bazasi) sanaladi.

2.3. Ekspert tizimining afzalliklari va tizimni yaratish moduli

Tizimni yaratish moduli qoidalar to'plamini yaratish uchun xizmat qiladi.

Tizimni yaratish modulining asosi bo'lgan ikkita yondashuv mavjud: dasturlashtirishning algoritmik tilidan foydalanish va ekspert tizimi qobig'idan foydalanish.

Bilimlar bazasini tasavvur etish uchun maxsus LISP va PROLOG tillari ishlab chiqilgan, garchi bundan boshqa har qanday ma'lum algoritmik tildan foydalanish mumkin bo'lsa ham.

Ekspert tizimi qobig'i. Tegishli bilimlar bazasini yaratish orqali ma'lum bir muammoni hal etishga moslashgan tayyor dasturiy muhitni ifodalaydi. Ko'pgina hollarda qobiqdan foydalanish dasturlashdan ko'ra tezkor va osonroq tarzda ekspert tizimini yaratish imkonini beradi.

Ekspert tizimining afzalliklarini tajribali mutaxassislarga qiyoslab shunday bayon etish mumkin:

1) erishilgan puxta bilim, asos yo'qolmaydi, u hujjatlashtirishi, uzatilishi, ijro etilishi va ko'payishi mumkin;

2) nisbatan mustahkam natijalarga erishiladi, insondagi hissiy va shu kabi boshqa ishonchsiz omillar bo'lmaydi;

3) tizimning ishlab chiqish qiymati yuqori, lekin ekspluatatsiya qiymati past. Umuman qiyoslaganda esa u yuqori malakali mutaxassislardan ko'ra arzonroq tushadi.

Yangi qoida va konsepsiyalarga, ijodkorlik va ixtirochilikka unchalik moslashmaganligi hozirgi ekspert tizimining kamchiligidir. Ko'p hollarda bu tizim yuqori malakali mutaxassislar o'rnini bosa oladi, ammo ba'zan past malakali ekspertga muhtoj joylar ham bo'lib turadi. Ekspert tizimi eng oxiridagi foydalanuvchining kasb imkoniyatlarini kengaytirish va ko'paytirish vositasi bo'lib xizmat qiladi.

Ochig'i, bu tizim muayyan bir predmet sohasida mutaxassis-ekspertlar darajasidagi bilimni namoyish etmog'i kerak. Tizim yaxshi yechimlarni kerakli darajada topa olmaydi, lekin predmetni keng anglaydi.

Rejalashtiruvchi ekspert tizimlari ma'lum bir maqsadlarga erishish uchun zarur bo'lgan dasturlarni ishlab chiqishga mo'ljallangan.

Bashoratlovchi ekspert tizimlari o'tmish va bugunning voqealariga asoslanib kelajak ssenariysini oldindan aytib bermog'i, ya'ni berilgan vaziyatdan ishonchli natijalar chiqarishi kerak. Buning uchun bashoratlovchi ekspert tizimlarida dinamik parametrik modellar qo'llaniladi.

Tashxislovchi ekspert tizimlari kuzatiladigan hodisalarning normal emasligi sabablarini topish xususiyatiga ega. Ma'lumotlar to'plami tahlil uchun asos bo'lib xizmat qiladi. Ular yordamida etalon xatti-harakatdan chetlanish aniqlanadi va tashxis qo'yiladi.

O'rgatuvchi ekspert tizimlari foydalanuvchilarga berilgan sohada tashxis qo'yish va tahlil etish imkoniyatini berishi lozim. Bunday tizimdan bilim va xatti-harakat to'g'risidagi farazni yaratish, tegishli ta'lim uslubini va harakat usullarini aniqlash talab etiladi. Ekspert tizimini yaratishda kamida uchta muammo yuzaga keladi:

- xotiraga kiritiladigan axborotning yetarli darajada to'liq bo'lishini ta'minlash. Bu eng asosiy bilimlarni ajratish va ma'lumotlar tuzilmasida ularning o'zaro aloqasini o'rnatish, shuningdek, kodlashtirishning bunday tizimini yaratish va foydalanishni talab etadi;

- ekspert tizimi faoliyati sifatining samarali bahosini olish va tegishli mezonlarni ishlab chiqish. Qiyinchilik shundaki, mutaxassislar bilimi – bu shunchaki ma'lumot va faktlar yig'indisi emas. Ayrim elementlar munosabatini tasavvur etish uchun aloqalar qonuniyatlarini hisobga olishga formal urinish tizimni o'ta darajada «keskin» qilib qo'yadi va u yangi elementlarni qo'shish uchun «yopiq» bo'lib qoladi;

- yechiladigan masala tuzilmasining ehtimollik xususiyati va bilimlarning uyg'unlashuvi tufayli ishonchsiz natijalar olish mumkinligi.

Ekspert tizimini yaratish quyidagi talablar mavjud holatda maqsadga muvofiqdir:

- tizimga o'z bilimini berishni istagan ekspertlar mavjudligi;

- ekspertlar vazifani hal etishning o'z uslublarini bayon etishi mumkin bo'lgan muammoli sohaning mavjudligi;

- ko'pchilik ekspertlarning mazkur muammoli sohada yechimlar o'xshashligining bo'lishi;

- muammoli sohadagi vazifaning ahamiyati, ya'ni ular yoki murakkab bo'lishlari, yoki mutaxassis bo'lmagan foydalanuvchi hal eta olmasligi yoki hal etish uchun ancha vaqt talab qilishi;

- masalani yechish uchun katta hajmdagi ma'lumot va bilimning bo'lishi;

- predmet sohasida axborotning to'liq bo'lmasligi va o'zgaruvchanligi tufayli evristik uslublarni qo'llash.

Yuqorida qayd etilgan uchta muammoni hal etish va sanab o'tilgan talablarni bajarish ekspert tizimini qo'llashning zarur hamda yetarli sharti sanaladi.

Ekspert tizimini yaratish bosqichlari. Ekspert tizimini yaratishning nisbatan muhim bosqichlariga quyidagilarni kiritish mumkin: konseptualizatsiya, realizatsiya, testdan o'tkazish, joriy etish, kuzatib borish, modernizatsiyalash.

Konseptualizatsiya bosqichida ekspert tizimini ishlab chiqish bo'yicha mutaxassis ekspert bilan hamkorlikda tanlangan predmet sohasidagi muammoni yechishning uslublarini bayon etish uchun qanday tushuncha, munosabat va tadbirlar zarurligini hal etadi. Bosqichdagi asosiy vazifa masalani yechish jarayonida yuzaga keluvchi vazifa strategiyasi va cheklovlarni tanlashdan iborat. Konseptualizatsiya muammoni to'liq tahlil etishni talab qiladi.

Identifikatsiya bosqichida vazifa turi, tavsifi, o'lchami, ishlanma jarayonidagi ishtirokchilar tarkibi aniqlanadi. Modelning yaroqliligi ko'rib chiqiladi, talab etiladigan vaqt-mashina resurslari baholanadi, ekspert tizimini yaratish maqsadi belgilanadi.

Formallashtirish bosqichida asosiy tushunchalar va munosabatlarni bilimlarni ifodalashning o'ziga xos rasmiy tiliga o'tkaziladi. Bu yerda ko'rib chiqilayotgan vazifa uchun modellar yoki ma'lumotlarni taqdim etishning o'xshash usullari tanlanadi.

Amalga oshirish bosqichida yuklatilgan vazifalarni bajarishga qodir bo'lgan ekspert tizimining jismoniy «qobig'i», yuzasi yaratiladi.

Ekspert tizimi faoliyatining to'g'riligini testdan o'tkazish bosqichida tekshirish mumkin.

2.4. Sun'iy intellektning asosiy tushunchalari

Sun'iy (kompyuter) aql – bu axborot texnologiyasi va kompyuter muhandisligining eng istiqbolli yo'nalishlaridan biri. Sun'iy intellekt sohasidagi ish an'anaviy intellektual muammolarni hal qilish uchun kompyuter tizimlarini (o'qitish, ekspert, konsalting, robot va boshqalar) loyihalash uchun usullar, vositalar va texnologiyalarni yaratishga qaratilgan.

Intellekt so'zi lotincha «intellectus» so'zidan olingan bo'lib, u bilish (aniqlash), tushunish yoki fahmlash (aql) ma'nosini beradi. Intellekt maqsadga erishishda zarur bo'ladigan faktlar va metodlar to'plamidan iborat. Maqsadga erishish – bu faktlarga zaruriy qoidalarini qo'llash hisoblanadi.

Sun'iy intellekt (SI) – bu dasturiy muhitning shunday tizimiki, unda inson tafakkuri kompyuter jarayoniga imitatsiyalangan. Sun'iy

intellekt atamasi 1956-yilda Stenford universiteti (AQSH) tomonidan taklif qilingan.

Sun'iy intellekt kompyuterni intellektning alomatlari bilan «jihozlashni» ko'zda tutadi. Sun'iy intellekt usullari dasturlarni birlashtirishni soddalashtiradi va tizimga o'z-o'zini o'qitish va yangi axborotlarni jamlash qobiliyatini kiritish imkoniyatini beradi.

Odam bilimlarni fikrlash usulini o'zgartirmasdan, ma'lum bo'lgan dalillarni esdan chiqarmasdan jamlashi mumkin. Sun'iy intellekt tizimi xuddi shunday ishlab chiqiladi. Bunda inson xotirasining bloklariga o'xshab dasturlarning ayrim qismlarining yuqori mustaqilligiga erishiladi. Odamning miyasi kerakli axborotni tanlab turib, faqat ushbu muammoga tegishli bo'lgan dalillarni ulaydi, bunda u kirishi mumkin bo'lgan barcha ma'lumotlardan foydalanmaydi. Inson faoliyatining asosida fikrlash yotadi va ushbu holda fikrlash jarayonining maqsadi yakuniy natija bo'ladi. Bitta maqsadga erishilgandan keyin yangi maqsad qo'yiladi va erishiladi.

Sun'iy intellekt tizimi rivojlanishining quyidagi bosqichlarini ko'rib chiqishimiz mumkin:

1) XX asrning 70-yillari masalalarni yechish metodlarini izlash va ularni universal dasturlarni qurishda foydalanish bilan xarakterlanadi;

2) XX asrning 80-yillari axborotlarni tasavvur qilishning umumiy metodlarini izlashga va ularni maxsus dasturlarga qo'llash usullarini qidirish bilan xarakterlanadi;

3) XX asrning 90-yillari bir qancha fan sohalari bo'yicha maxsus dasturlarni yaratish uchun katta hajmli yuqori sifatli maxsus bilimlarning qo'llanishi bilan xarakterlanadi. XX asrning 90-yillari boshlarida butunlay yangi konsepsiya qabul qilindi. Intellektual dasturni tuzish uchun, izlanish fan sohasining yuqori sifatli maxsus bilimlari bilan ta'minlash kerak. Shuning uchun loyihalashtirilayotgan SI tizimi yuqori bosqichdagi bilimlari bazasiga ega bo'lishi kerak;

4) ma'lum fan sohasidagi sun'iy intellekt tizimi ekspert deb ataladi. Ekspert tizimi (ET) – bu aniq fan sohasidagi mutaxassislarning bilimlari to'plovchi va kam ixtisoslashgan foydalanuvchilarga kon-

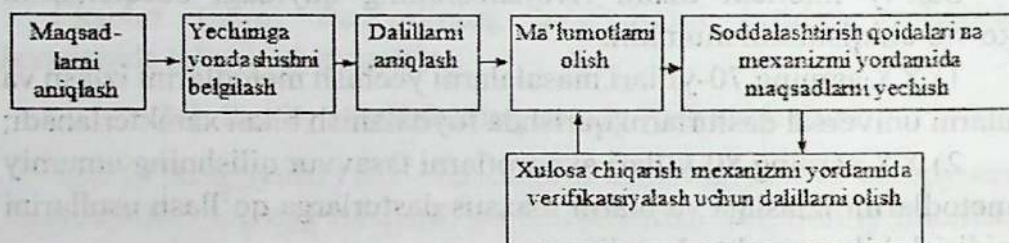
sultatsiya uchun empirik tajribasini adadlashtiruvchi (nusxa yozish) murakkab dasturlar kompleksi;

5) ET uchun bilimlarni qabul qilishda shu fan sohasidagi ekspertlar xizmat qiladi.

Bugungi kunda ayrim davlatlarda robot-hamshiralari, haydovchisiz transport vositalari, buyurtmani yetkazib beruvchi dronlar xizmatidan foydalanish yo'lga qo'yilgan. Hatto politsiya xodimlarining ba'zi vazifalarini ham maxsus robotlar bajarishyapti. Olimlar ularning tashqi ko'rinishini imkon qadar odamlarnikiga o'xshatishga urinishmoqda.

2.5. Sun'iy intellekt strukturasi tuzilishi

Sun'iy intellekt haqida gapirish uchun dasturiy tizim inson tomonidan qaror qabul qilish jarayonini tashkil qiluvchi barcha elementlar, maqsadlar, dalillar, qoidalar, mexanizmlar, xulosalar va soddalashtirishga ega bo'lishi kerak. Bunday elementlarning asosiy tarkibiy qismlari 2.3-rasmda ko'rsatilgan.



2.3-rasm. Sun'iy intellekt tizimining tarkibiy qismlari.

Sun'iy intellekt tizimining an'anaviy dasturiy tizimlardan asosiy farqi shundan iboratki, uning tuzilishining tarkibiy qismlari bo'lingan holda aniqlanadi va uning istalgan qismini zamonaviylashtirish umumiy tuzilmaga ta'sir qilmaydi.

Sun'iy intellekt tizimlarini loyihalashtirishda birinchi bosqichda unga erishish uchun mo'ljallangan maqsad aniqlanadi, yechilayotgan vazifalarni yirik atamalarda bayon qilishni bilish uchun zarur sinfi belgilaniladi. Dalillar sun'iy intellekt tizimining muhim qismi bo'la-

di, ularsiz maqsadga erishish mumkin emas. Har bir maqsadning dalillari bor. Har bir dalil o'zining salmog'iga ega, ya'ni har bir dalilga nisbatan muhimlik xosdir. Ushbu vazifani yechish uchun dalil qanchalik katta ahamiyatga ega bo'lsa, uning salmog'i shunchalik ulkan. Maqsadlarga erishish uchun zarur bo'lgan umumiy dalillar aniqlangandan keyin, aniq ma'lumotlarni olish kerak. Ma'lumotlarni olish uchun tegishli savollar shakllantiriladi, ularga javoblar tizimining yakuniy qarorga kelishiga yordam beradi. Sun'iy intellekt tizimlari uchun dasturni ishlab chiqish quyidagi bosqichlardan iborat:

- 1) maqsadlarni aniqlash;
- 2) bu maqsadlarga tegishli dalillarni aniqlash;
- 3) ushbu vaziyat uchun xos bo'lgan dalillarga mos ma'lumotlarni olish;
- 4) xulosa chiqarish qoidalari va mexanizmidan foydalanish bilan ma'lumotlarni olish.

Ma'lum dalillar, qoidalarga ko'ra berilgan vaziyatga nisbatan qo'llaniladi. Qoidalar ma'lumotlarni to'g'ri baholash va maqsadga erishishga yordamlashadi.

Nazorat savollari

1. Sun'iy intellekt qanday vazifalarni bajaradi?
2. Sun'iy intellekt tizimining tarkibiy qismlari nimalardan iborat?
3. Ekspert-tizimi nima?
4. Bilimlar muhandisining vazifalari nimalardan iborat?
5. Ekspert tizimining instrumental muhitiga nimalar kiradi?

3-BOB. ZAMONAVIY AVTOMATLASHTIRILGAN LOYIHALASH TIZIMLARI VA ULARNING TEXNIK SOHALARDA QO'LLANILISHI

3.1. Loyihalash jarayonlari va bosqichlari

Kompyuter yordamida loyihalash va boshqarish tizimlari eng murakkab zamonaviy sun'iy tizimlar qatoriga kiradi. Tizimli yondashuvsiz ularni loyihalash va texnik xizmat ko'rsatish mumkin emas. Shuning uchun tizim muhandisligi g'oyalari va qoidalari zamonaviy avtomatlashtirilgan tizimlar va ularni qo'llash texnologiyalarini o'rganishga bag'ishlangan fanlarning ajralmas qismi hisoblanadi.

Texnik obyektlarni loyihalash – hali mavjud bo'lmagan obyekt obrazini yaratish, o'zgartirish va taqdim etish jarayonidir. Obyekt yoki uni tashkil qilgan qismlar obrazi inson hayolida ijodkorlik natijasida yaratilishi yoki kompyuter va insonning o'zaro aloqasi jarayonida ba'zi bir algoritmlarga mos ravishda amalga oshirilishi mumkin. Nima bo'lishdan qat'i nazar muhandislik loyihalash jamoatning ba'zi bir texnik obyektlarga bo'lgan ehtiyoji borligi tufayli boshlanadi. Texnik obyektlar qurulish obyektlari, ishlab chiqarish mahsulotlari yoki jarayonlar bo'lishi mumkin. Barcha loyihalash jarayonini yoki uning bir qismini yechish inson va kompyuterning o'zaro aloqasi orqali amalga oshiriladigan loyihalash avtomatlashtirilgan deb ataladi. Avtomatlashtirilgan loyihalashni amalga oshiradigan tizim avtomatlashtirilgan loyihalash tizimini tashkil qiladi (ingliz tilidan olinganda CAD System – Computer Aided Design System).

Murakkab obyektlarni loyihalash bir qator nazariya va yondashuvlarda ifodalangan g'oya va prinsiplarni qo'llashga asoslangandir. Murakkab tizimlarni loyihalashning turli usullarini o'zida qamragan g'oyalardan tashkil topgan tizimli yondashuv har tomonlama umumiy yondashuv hisoblanadi. Tizimli yondashuvning asosiy umumiy prin-

sipi murakkab tizimlarni ularning o'zaro aloqasini hisobga olgan holda qabul qilishdan iborat. Tizimli yondashuv tizim tuzilishini yaratish, aloqalarni turlarga bo'lish, atributlarni aniqlash, tashqi muhit ta'sirini tahlil qilish kabi jarayonlarni qamrab olgan.

Avtomatlashtirilgan tizimning vazifalari:

- hujjatlarni avtomatlashtirish;
- qabul qilish jarayonining ma'lumotlarini qo'llab-quvvatlash va avtomatlashtirish;
- parallel loyihalash texnologiyasidan foydalanish;
- loyiha yechimlari va loyiha jarayonlarini birlashtirish;
- loyiha ma'lumotlarini va vazifalarini qayta ishlash;
- loyihani matematik modellash;
- loyihani boshqarish sifatini yaxshilash;
- ixtiyoriy loyiha va optimallashtirish usullarini qo'llash.

3.2. Avtomatlashtirilgan loyihalash tizimlarining strukturasi

Avtomatlashtirilgan loyihalash tizimi – avtomatik tizim funksiyalarini va loyiha qilish jarayonlarini, tashkilot va texnik tizim loyihalash jarayonini avtomatlashtirishni, kompleks texnik va dasturiy ta'minotni avtomatlashtirish vositalarining majmuasini o'zida aks ettiradi.

Loyihalash bosqichlari – vaqt o'tishi bilan rivojlanib boradigan jarayon kabi loyihalashning katta bo'lagidir. Umumiy holda bosqichlar ilmiy-tadqiqot ishlari, loyiha eskizi yoki tajriba-konstruktorlik ishlari, texnik, ishchi loyiha, tajriba namunalarini sinab ko'rish kabi bosqichlarga bo'linadi. Ilmiy-tadqiqot ishlari bosqichi ayrim holda loyiha oldi tadqiqoti yoki texnik takliflar bosqichi deb ham ataladi. Bir bosqichdan ikkinchisiga o'tish jarayonida loyihani ishlab chiqishdagi aniqlik va puxtalik darajalari o'sib boradi va ishchi loyiha tajriba namunalarini tayyorlash uchun yetarli darajaga keltiriladi.

Loyiha turlari:

- avtomatlashtirilmagan loyiha – loyihalash jarayoni (kompyuter-ni ishlatmagan holda);

- avtomatlashtirilgan loyiha – loyihalarning hamma yechimlari va ularning bajarilish qismi inson va kompyuter harakatida bajariladi;
- avtomatik loyiha – loyiha inson qatnashuvisiz kompyuterdagi dasturlarga asoslangan holda amalga oshiriladi.

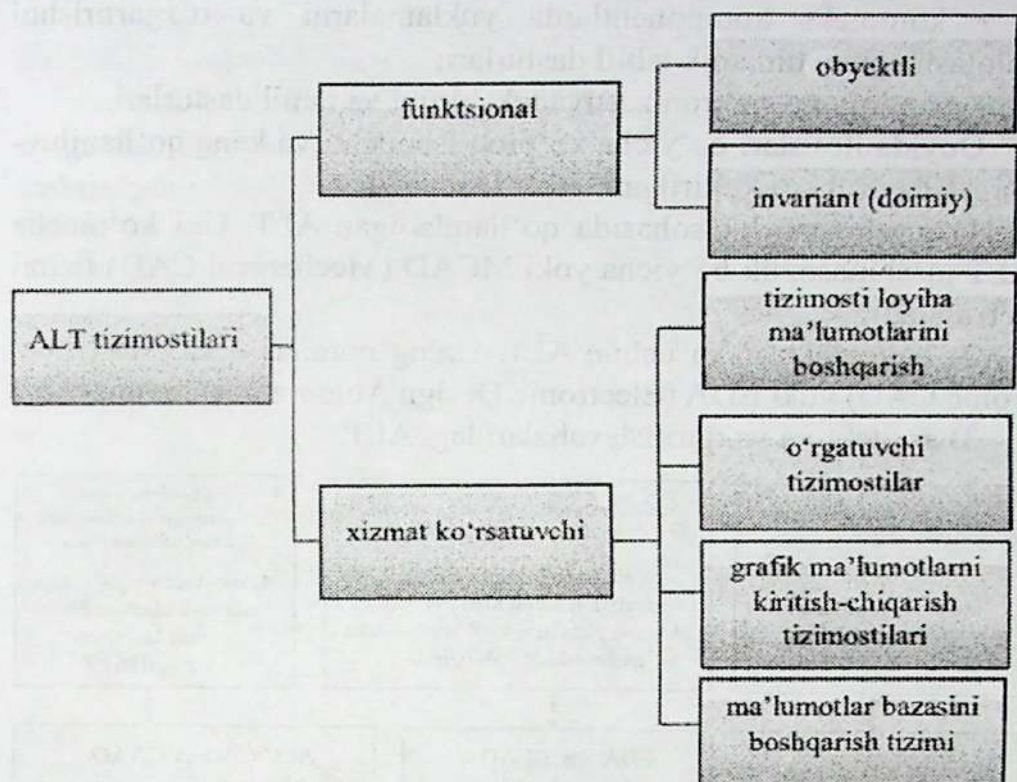
Har qanday murakkab tizim kabi ALT (avtomatlashtirilgan loyihalash tizimlari) ham tizimostidan tashkil topadi. Loyihalash va xizmat ko'rsatish tizimosti mavjud. Loyihalash tizimostilari bevosita loyihalash tartib-taomillarini bajaradi. Loyihalash tizimostiga misol qilib mexanik obyektlarni geometrik uch o'lchovli modellashtirish, konstruktorlik hujjatlarini tayyorlash, sxemotexnik tahlil qilish tizimostilari kiradi.

Xizmat ko'rsatuvchi tizimostilar loyihalanayotgan tizim ostini ishlashini tashkillashtiradi, ularning birlashmasi ko'pincha ALTning tizim muhiti (yoki qobiq) deb ataladi. Xizmat ko'rsatuvchi tizimostilari bo'lib loyiha kattaliklarini boshqarish (PDM – Product Data Management), loyihalash jarayonini boshqarish (DesPM – Design Process Management), kompyuter bilan mutaxassis o'rtasida aloqa o'rnatadigan foydalanuvchi interfeysi, ALTning dasturiy ta'minotini ishlab chiquvchi – CASE (Computer Aided Software Engineering) ALTda qo'llaniladigan foydalanuvchi texnologiyasini o'rganuvchi o'quv tizimostilari xizmat qiladi.

Loyihalash tizimlari avtomatlashtirilgan loyihalash (CAD), avtomatlashtirilgan ishlab chiqarish (CAM) va avtomatlashtirilgan konstruktorlik (CAE) yo'nalishlarga bo'linadi.

Avtomatlashtirilgan loyihalash CAD (computer-aided design) tizimi:

- loyihalarni tuzatish, o'zgartirish, tahlil qilish va optimizatsiyasini yengillashtirish uchun kompyuter tizimlaridan foydalanish texnologiyasi;
- kompyuter grafikasi bilan ishlaydigan dastur, shuningdek, muhandislik hisob-kitobida qo'llaniladigan tizim;
- chizma chizish va geometrik modellashtirish tizimi;



3.1-rasm. ALT tizimostilari.

Avtomatlashtirilgan ishlab chiqarish CAM (computer-aided manufacturing) tizimi:

- kompyuter ishlab chiqarish resurslari bilan to'g'ridan to'g'ri yoki o'zaro bog'liqlik orqali ishlab chiqarishni rejalashtirish, boshqarish va boshqarish uchun kompyuter tizimlaridan foydalanishni o'z ichiga olgan texnologiya;

- stanoklar uchun uskunalar va ishlov beriladigan qismlarni tanlash va o'rnatish orqali moslashuvchan maydonlarda ishlashga qodir avtomatlashtirilgan robotlarni dasturlash.

Avtomatlashtirilgan konstruktorlik CAE (computer-aided engineering) tizimi:

- mexanizmalarda harakat trayektoriyalarini va ularning tezliklarini aniqlay oladigan kinematik hisoblar uchun dasturlar;

- kompleks komponentlarda yuklamalarni va o'zgartirishni aniqlash uchun dinamik tahlil dasturlari;
- mantiqiy va sinxronizatsiya tekshiruvi va tahlil dasturlari.

Quyida ilovalari bo'yicha ko'proq ishonchli va keng qo'llaniluvchi ALT guruhlari keltirilgan:

1) mashinasozlik sohasida qo'llaniladigan ALT. Uni ko'pincha ALT mashinasozlik bo'yicha yoki MCAD (Mechanical CAD) tizimi deb ataladi;

2) radioelektronika uchun ALT. Uning nomlari – ECAD (Electronic CAD) yoki EDA (Electronic Design Automation) tizimlar;

3) arxitektura va qurilish sohalaridagi ALT.



3.2-rasm. Loyihalash tizimlarida ishlovchi ilovalar.

Belgilangan maqsad bo'yicha loyihalashning turli jihatlarini ta'minlovchi ALT yoki ALT tizimostilari ajratiladi. Shunday qilib, MCAD tarkibida CAE/CAD/CAM tizimlar paydo bo'ladi:

- funksional loyihalovchi ALT, boshqacha aytganda, ALT-F yoki CAE (Computer Aided Engineering) tizimlari;

- umumiy mashinasozlikning konstruktorlik SAR – ALT-K, u ko‘pincha CAD-tizim deb ataladi;

- umumiy mashinasozlikning texnologik ALT – boshqacha qilib aytganda, ishlab chiqarishni texnologik tayyorlashning avtomatlashtirilgan tizimlari yoki CAM (Computer Aided Manufacturing) tizimlari.

Qo‘llanilish ko‘lami bo‘yicha ALTning alohida dasturiy-uslubiy komplekslari mavjud. Masalan, so‘nggi elementlar usuli bilan mos ravishda mexanik buyumlarning chidamlilik tahlili kompleksi yoki elektron sxemalar tahlili kompleksi, dasturiy (software) va texnik (hardware) ta‘minotning noyob arxitektura tizimlari kompleksi.

Baza tizimosti xarakteri bo‘yicha ALTning quyidagi ko‘rinishlari mavjud:

1) mashina grafikasi va geometrik modellash tizimosti bazasidagi ALT. Bu ALT konstruktorlash loyihalashning asosiy protsedurasi hisoblangan ilovalarga yo‘naltirilgandir. Konstruktorlash jarayoni fazoviy forma va obyektlarning o‘zaro joylanishini aniqlashdir. Shuning uchun bu guruhga mashinasozlik sohasidagi ALTning ko‘pgina grafik yadrolari kiradi;

2) MMBT bazasidagi ALT. Ular qiyin bo‘lmagan matematik hisoblashlar orqali ko‘p miqdordagi kattaliklarni qayta ishlovchi ilovalarga mo‘ljallangan. Bunday ALT iqtisodiy-texnik ilovalarda ko‘proq uchraydi. Masalan, biznes-rejalarni loyihalashda. Umuman, bular mustaqil ishlatiluvchi dasturiy-uslubiy komplekslardir. Bularga, masalan, ishlab chiqarish jarayonini imitatsion modellash tizim, oxirgi elementlar usuli bilan chidamlilikni hisoblash, avtomatlashtirilgan boshqaruv tizimini sintez va tahlil qilish komplekslari kiradi. Ko‘pincha bunday ALT CAE tizimiga kiradi. Unga VHDL bazasida yaratilgan mantiqiy loyihalash dasturlari, MathCAD matematik paketi misol bo‘ladi;

3) kompleks (integrallashgan) ALT – avvalgi turdagi tizimostilarni birlashmasidan tashki topgan. Kompleks ALTga mashinasozlikdagi CAE/CAD/CAM-tizimlari yoki ALT BIS kiradi. ALT BIS MBBT va prinsipial, mantiqiy va funksional sxemalar, kristallar topologiya-

si, mahsulotning yaroqliligini tekshirish testlari komponentlarini loyihalash tizimostilaridan tashkil topgan. Bunday murakkab tizimlarni boshqarish uchun ixtisoslashtirilgan tizimli muhitlar ishlatiladi.

3.3. Loyihalash jarayonlaridagi CAE/CAD/CAM-tizimlari

Mashinasozlikda CAD-tizim vazifasi ikki o'lchamli (2D) va uch o'lchamli (3D) loyihalashdan iborat. 2D loyihalashga chizmachilik, konstruktorlik hujjatlarini rasmiylashtirish, 3D loyihalashga uch o'lchamli modellarni olish, metrik hisoblashlar, real vizuallashtirish, 2D va 3D modellarini o'zaro o'zgartirish funksiyalari kiradi.

CAD-tizimlari «yengil» va «og'ir» tizimlarga bo'linadi. Yengil tizimlar 2D grafikaga yo'naltirilgan bo'lib, arzon va hisoblash resurslariga nisbatan talab kamroq. Og'ir tizimlar 3D geometrik modellashtirishga yo'naltirilgan bo'lib, universal, qimmat hisoblanadi. Ularda chizma hujjatlarni rasmiylashtirish uch o'lchamli geometrik modellarni qayta ishlash orqali amalga oshiriladi.

CAM-tizimning asosiy vazifasi texnologik jarayonlarni ishlab chiqish, sonli boshqaruv dasturli texnologik qurilmalar uchun boshqaruvchi dasturlar sintezi, qayta ishlash jarayonlarini modellashtirish, shu bilan birga asbobning nisbiy harakat trayektoriyasini qurish, ishlov berish jarayonini tayyorlash, ishlov vaqti normalarini hisoblashdan iborat.

CAE-tizim vazifalari turli-tumandir. Sababi ular tahlil protseduralarini loyihalash, modellashtirish, loyiha qarorlarini optimallashtirish bilan bog'liqdir. Mashinasozlik CAE tizimi tarkibiga quyidagi protseduralar uchun tuzilgan dasturlar kiradi:

- chidamlilik tahlili kabi fizik kattalik maydonlarini modellashtirish;
- makro darajada o'tish jarayonlari va holatini hisoblash;
- ommaviy xizmat ko'rsatish modellari asosida murakkab ishlab chiqarish tizimlarini imitatsion modellashtirish.

ALTning aniq takliflarga bo'lgan ehtiyojini qulay moslashtirish, uni rivojlantirish uchun ALT tarkibida moslashtirish asboblari vositasi bo'lish maqsadga muvofiqdir. Bu vositalar kengaytirish tillarini

qo'shgan holda u yoki bu CASE-texnologiyalarida taqdim etilgan. Ayrim ALTLarda original asboblarni muhiti ishlatiladi.

3.4. CALS-texnologiyalarni qo'llash sohalari

CALS-texnologiya – bu sanoat ishlab chiqarishini kompleks kompyuterlashtirishdir. Uning maqsadi sanoat mahsulotini uning faoliyat davri barcha bosqichlarida o'ziga xos xususiyatlarini standartlashtirish va unifikatsiyalashdir. Asosiy o'ziga xos xususiyatlar loyiha, texnologik, ishlab chiqarish, marketing, ekspluatatsiya hujjatlarida taqdim etilgan. CALS-tizimlarda kompyuter muhitida ma'lumotlarni saqlash, qayta ishlash va uzatish ko'zda tutilgan. Avtomatlashtirishning mos tizimlari avtomatlashtirilgan mantiqiy tizimlar yoki CALS (Computer Aided Logistic Systems) deb nomlangan. Sababi mantiqiy deganda odatda zaxiralarni boshqarish va ta'minlashga qaratilgan masalalar tushuniladi, CALS vazifalari esa kengroq bo'lib, sanoat mahsulotlari faoliyat davrining barcha bosqichlari bilan bog'liqdir.

«Ishlab chiqarish» bosqichiga nisbatan CALS texnologiyalaridan foydalanish quyidagilar tufayli ishlab chiqarish samaradorligini oshirishga imkon beradi:

1) mahsulotlarning ishlab chiqarish jarayonlarini tavsiflovchi moddiy, axborot va moliyaviy oqimlarni modellashtirish, mahsulotning ko'rsatilgan texnik-iqtisodiy parametrlariga erishilishini ta'minlaydigan texnologik jarayonlarning optimal to'plamini tanlash;

2) ishlab chiqarish xarajatlarini minimallashtirishni ta'minlaydigan, mahsulot aylanishining barcha bosqichlarida mahsulotni qo'llab-quvvatlashning yaxlit axborot tizimi;

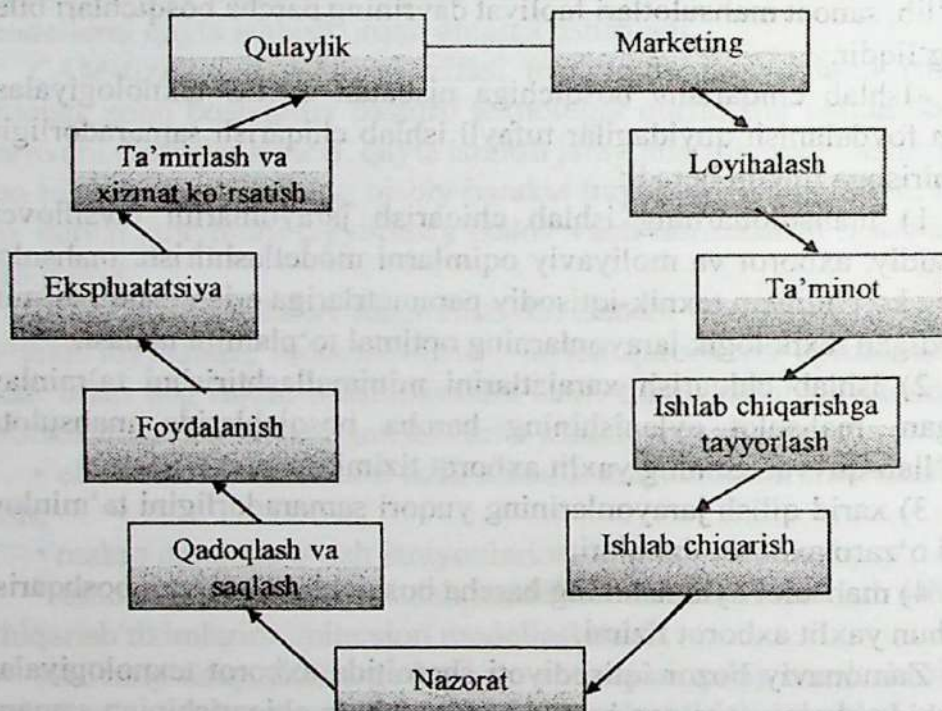
3) xarid qilish jarayonlarining yuqori samaradorligini ta'minlovchi o'zaro axborot tizimlari;

4) mahsulot aylanishining barcha bosqichlarida sifatni boshqarish uchun yaxlit axborot tizimi.

Zamonaviy bozor iqtisodiyoti sharoitida axborot texnologiyalari alohida ahamiyatga ega bo'lib, sanoat ishlab chiqarishining samaradorligini oshirishga yordam beradi.

CALS texnologiyalari doirasida mahsulotning aylanishi bosqichlari:

- marketing tadqiqotlari;
- mahsulot dizayni;
- rejalashtirish va jarayonlarni ishlab chiqish;
- xomashyo yoki butlovchi qismlarni sotib olish;
- ishlab chiqarish yoki xizmat ko'rsatish;
- tekshirish;
- qadoqlash va saqlash;
- sotish va tarqatish;
- o'rnatish va ishga tushirish;
- texnik yordam va xizmat ko'rsatish;
- maqsadga muvofiq ishlash;
- sotishdan keyingi faoliyat;
- amalga oshirish yoki qayta ishlash.



3.3-rasm. Mahsulotning aylanishi bosqichlari.

Hozirgi vaqtda CALS texnologiyalariga asoslangan mahsulotlarni boshqarishning tipik kompyuter tizimlari ishlab chiqilgan. Mahsulot sifatini boshqarishning kompyuterga asoslangan tizimlarini joriy qilish uchun ishlab chiqilgan dasturiy va texnik vositalardan foydalanish odatdagi elektron modelni ma'lum bir korxona tuzilmasiga moslashtirish orqali amalga oshiriladi.

CALS – bu eng yangi kompyuter texnologiyalari bo'lib, mahsulotlarni loyihalash, ishlab chiqarish, sinovdan o'tkazish, yetkazib berish va ishlatish jarayonlari uchun yagona axborot muhitini yaratishni ta'minlaydi.

Bularni e'tiborga olgan holda CALSni quyidagicha shifrdan chiqarish mumkin: CALS – Computer Acquisition and Life Cycle Support.

CALSni qo'llash loyihalash ishi ko'lamini qisqartirish imkonini beradi. Sababi oldindan loyihalangan jihozlar qismlari, mashinalar va tizimlar tarkibining tavsiflari CALS texnologiyasi foydalanuvchilariga kirish mumkin bo'lgan tarmoq serverlarining ma'lumotlar bazalarida saqlanadi. Turli tizim va sohalardagi mahsulotlarning yaroqliligi va integratsiyasi yechimi, ekspluatatsiya sharoitlarining o'zgarishiga moslashuvi, loyihalash tashkilotlarini ixtisoslashtirish va boshqa shu kabi muammolar ancha yengillashadi. Murakkab texnik mahsulotlarning bozordagi yutug'ini CALS texnologiyasiz tasavvur qilib bo'lmaydi.

Sanoatda loyihalash va boshqarish uchun ochiq taqsimlangan avtomatlashtirilgan tizimni qurish zamonaviy CALS-texnologiyasining asosini tashkil qiladi. Ularni qurishdagi asosiy muammo global masshtabga ega bo'lgan umumiy tizimda joy va vaqtdan qat'i nazar kattaliklar interpretatsiyasini va ularni bir xilda tavsiflashni ta'minlashdir. Loyiha, texnologik va ekspluatatsion hujjatlar tuzilmasi hamda uni tasvirlash tillari standartlashgan bo'lishi kerak. U holda turli CAE/CAD/CAM tizimlarni ishlatadigan, vaqt va fazoda bo'lingan turli kollektivlarning umumiy loyihasi ustida olib borilayotgan ishning muvaffaqiyatli bo'lishi muqarrardir. Bitta konstruktorlik hujjati turli loyihalarda ko'p marotaba ishlatilishi mumkin. Texnologik

hujjat esa turli ishlab chiqarish talablariga moslashgan. Bu loyihalash va ishlab chiqarish umumiy davrini qisqartirish va arzonlashtirishga olib keladi. Bundan tashqari, tizim ekspluatatsiyasi soddalashadi.

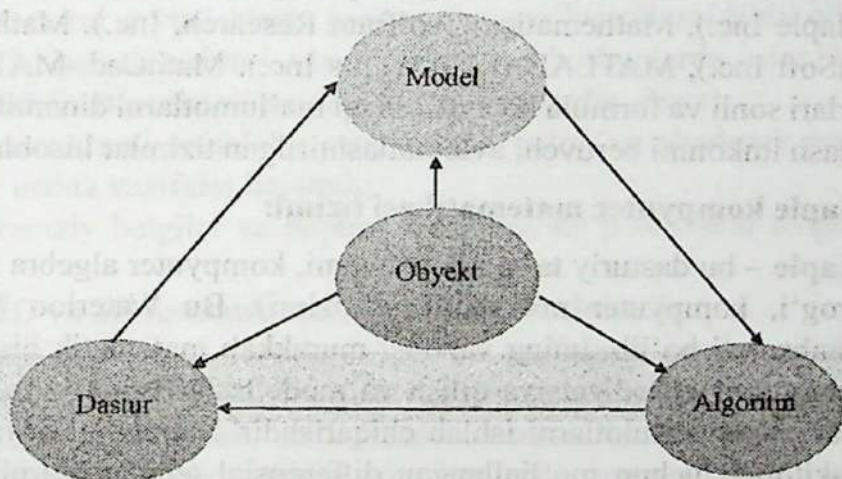
Nazorat savollari

1. Loyihalashning avtomatlashtirilgan tizimi nima?
2. Loyihalash bosqichlari qanday ketma-ketlikdan iborat?
3. ALT turlari va modellari?
4. Ilmiy ishlarni avtomatlashtirishning qanday vositalarini bilasiz?
5. CALS texnologiyasi qanday texnologiya?

4-BOB. MATEMATIK MODELLASHTIRISH, SONLI TAHLIL USULLARINI TIZIMLARDA AMALGA OSHIRISH

4.1. Matematik modelashtirish va ularning turlari

Matematik model – bu obyektning ekvivalenti bo‘lib, uning eng muhim xususiyatlarini, qonunlarini, uning tarkibiy qismlariga xos bo‘lgan aloqalarni va boshqalarni matematik shaklda aks ettiradi. Bundan tashqari, obyektning matematik modelashtirish muammosining o‘zi aniq harakatlar rejasini ishlab chiqaradi. Uni shartli ravishda uch bosqichga bo‘lish mumkin: model, algoritm, dastur (4.1-rasm).



4.1-rasm. Matematik modelning ekvivalentligi.

Birinchi bosqichda obyektning «ekvivalenti» tanlanadi, uning eng muhim xususiyatlarini matematik shaklda aks ettiradi, uning tarkibiy qismlariga xos bo‘lgan aloqalari qonunlarga bo‘ysunadi. Matematik model nazariy usullar bilan o‘rganiladi, bu bizga obyekt haqida oldindan muhim bilimlarni olish imkonini beradi.

Ikkinchi bosqichda modelni kompyuterda amalga oshirish algoritmi tanlanadi yoki ishlab chiqiladi. Model raqamli usullarni qo'llash uchun qulay shaklda taqdim etiladi. Kerakli aniqlikdagi kerakli miqdorlarni topish uchun bajarilishi kerak bo'lgan hisoblash va mantiqiy operatsiyalar ketma-ketligi aniqlanadi. Hisoblash algoritmlari modelning asosiy xossalarini buzmasligi va natijada asl obyekt bo'lishi va yechilayotgan muammolar va ishlatilayotgan kompyuterlarning o'ziga xos xususiyatlariga moslashishi kerak.

Uchinchi bosqichda model va algoritmi kompyuter uchun qulay bo'lgan tilga «tarjima qiladigan» dasturlar yaratiladi. Bundan tashqari, ular tejankor va moslashuvchan bo'lishi kerak. «Model – algoritmi – dastur» uzburchagini yaratgandan so'ng, tadqiqotchi qo'liga universal, moslashuvchan va qulay vositani oladi, u avval «sinov» hisoblash tajribalarida tuzatiladi va sinovdan o'tkaziladi.

Matematik ALTLar orasida eng ko'p tarqalgani Maple V (Waterloo Maple Inc.), Mathematica (Wolfram Research, Inc.), MathCAD (MathSoft Inc.), MATLAB (MathWorks Inc.), MathCad, MATLAB dasturlari sonli va formula ko'rinishidagi ma'lumotlarni dinamik qayta ishlash imkonini beruvchi avtomatlashtirilgan tizimlar hisoblanadi.

Maple kompyuter matematikasi tizimi:

Maple – bu dasturiy ta'minot to'plami, kompyuter algebra tizimi (aniqrog'i, kompyuter matematikasi tizimi). Bu Waterloo Maple Inc. mahsuloti bo'lib, uning vazifasi murakkab matematik hisoblar, ma'lumotlarni vizualizatsiya qilish va modellashtirishga yo'naltirilgan dasturiy mahsulotlarni ishlab chiqarishdir. Maple tizimi ramziy hisob-kitoblar uchun mo'ljallangan differensial tenglamalarni sonli yechish va integrallarni topish uchun bir qancha vositalarga ega. Shu bilan birga grafik vositalarga ham ega.

Maple tizimida, birinchidan, interaktiv grafikaning misli ko'ril-magan imkoniyatlari, xususan, 3D dinamik interaktiv grafikada kadrlar yaratish imkoniyati mavjud. Ikkinchidan, bu Maple dasturlash tili bo'lib, u aniq mantiqqa ega, uni professional bo'lmagan dasturchi tushunishi mumkin. Uchinchidan, bu oddiy usullar bilan amaliy das-

turlar paketlarini yaratish va Maple tizimiga birlashtirish qobiliyati mavjud.

Kompyuter matematikasi uch o'lovli grafikaga foydalanuvchiga uch o'lovli grafik muhit bilan o'zaro aloqa qilish imkoniyatini beradi hamda uch o'lovli grafik modellarning xususiyatlarini o'rganish uchun noyob imkoniyatlarni ochib beradi.

Mathematica tizimi:

Mathematica – bu fan, muhandislik, matematika va hisoblashda keng qo'llaniladigan kompyuter algebra tizimi hisoblanadi. Mathematica texnik hisoblashning barcha sohalarini qamrab oladigan 5000 ga yaqin o'rnatilgan funksiyalarga ega. Bularning barchasi Mathematica tizimiga to'liq kiritilgan.

O'ttiz yillik tajribaga tayanib, Mathematica texnik hisoblashning barcha sohalarida, jumladan, neyron tarmoqlari, mashinani o'rganish, tasvirni qayta ishlash, geometriya, ma'lumotlarni tahlil qilish va qayta ishlash nazariyasi, vizualizatsiya va boshqalarda ustunlik qiladi. Mathematica tizimi quyidagi xususiyatlardan iborat:

- muammoli masalalar, tenglamalar tizimi va algebraik munosabatlar ustida vazifalar bajaradi;
- ramziy belgilar va raqamli formatda ko'p vositalar to'plamiga ega;
- 2D va 3D formatda natijalarni vizuallashtiradi;
- ma'lumotlarni, rasmlar, video va audiolarni import va eksport qiladi.

Tizim, shuningdek, sonli hisob-kitoblarni ham amalga oshiradi. Funksiyalarning qiymatlarini (shu jumladan, maxsuslarini) ixtiyoriy aniqlik bilan aniqlaydi. Tizim chiziqli algebraik qobiliyatlarni ham o'z ichiga oladi – matritsalar bilan ishlash (qo'shish, ko'paytirish, teskari matritsani topish, vektorga ko'paytirish, eksponentni hisoblash, determinantni olish), qiymatlarini qidirish vazifalarini bajaradi.

Bu tizim natijalarni ham raqamli, ham grafik shaklda taqdim etadi. Xususan, funksional grafikalar, shu jumladan parametrligri egri chiziqlar va yuzalar, geometrik shakllarni qurish (chiziqlar, aylanalar, to'rt-

burchaklar va boshqalar), grafiklarni yaratish va o'zgartirish vazifalarini amalga oshiradi. Bundan tashqari, tovushni ham qayta ishlaydi. Uning grafigi analitik funksiya yoki nuqtalar to'plamini o'rnatadi.

Ixtisoslashtirilgan MathCad tizimi:

MathCad – matematik va ilmiy-texnik hisob-kitoblar shaxsiy kompyuterlarni qo'llashning muhim sohasi hisoblanadi. Hisob-kitoblar texnik ma'lumotlarning asosi hisoblanadi. Kompyuterning hisoblash quvvati uni ilmiy ishlarni avtomatlashtirish vositasi sifatida ishlatish imkonini beradi. Murakkab hisoblash masalalarini yechishda maxsus yozilgan dasturlar ishlatiladi. Shu bilan birga ilmiy ishlarda cheklangan murakkablikdagi keng spektrli masalalar uchraydi. Ularni yechish uchun universal vositalarni ishlatish mumkin. Bunday masalalarga quyidagilar kiradi:

- mutaxassislar uchun odatiy bo'lgan shaklda yozilgan matn va formuladan iborat ilmiy-texnik hujjatlarni yaratish;
- sonli o'zgarmlar, o'zgaruvchilar va o'lchanadigan fizik kattaliklar ishlatiladigan matematik amallar natijasini hisoblash;
- vektor va matritsalar ustida amallar;
- tenglama va tenglamalar tizimini yechish;
- statistik hisoblar va ma'lumotlar tahlili;
- ikki o'lchamli va uch o'lchamli grafik chizish;
- tenglama va tizimlarni analitik hisoblash;
- differensiallash va integrallash;
- differensial tenglamalarni yechish;
- boshlang'ich shartlarni turli qiymatda hisoblash.

Yechilayotgan masalalar murakkablik darajasining obyektiv sabablarga ko'ra o'sishi algoritmning murakkablashishiga va ularning C, Pascal, Fortran kabi algoritmik tillarda amalga oshirilishiga olib keladi. Yanada ko'proq vaqt dastur kodini ishga tushirishga ketadi. Bu sabablar algoritm biriktirilgan avtomatlashtirishni boshqarish tizimining (ALT) yaratilishiga olib keldi. Bunday tizimlar paydo bo'lganiga ancha vaqt bo'lgan va ular tor miqyosda ixtisoslashgan.

4.2. Hisobli eksperiment. MathCAD dasturida statik modellarni amalga oshirish

MathCAD elementar arifmetikadan boshlab, murakkab sonli usullarni qo'llash bilan tugallaydigan turli ilmiy va muhandislik hisob-kitoblarni amalga oshirish imkonini beradigan matematik muharriridir. MathCad foydalanuvchilari – talabalar, olimlar, muhandislar, turli texnik mutaxassislardir. Qo'llanilishdagi oddiylik, matematik harakatlarning yaqqolligi, sonli usullar va tizim funksiyalari kutubxonasining kattaligi, belgili hisoblashlar imkoniyati hamda natijalarni taqdim etishning a'lo darajadagi apparaturalari (har xil turdagi grafika, chop etiladigan hujjatlar tayyorlash, kuchli vositali va Web-sahifalar) mavjudligi tufayli MathCad ommabop matematik ilovaga aylandi. MathCad 2001 boshqa zamonaviy matematik ilovalardan farqli ravishda WYSIWYG («What You See Is What You Get» – «Siz nimani ko'rsangiz, shuni olasiz») prinsipi asosida qurilgan. Shuning uchun uni ishlatish jarayoni juda sodda, xususan, unda matematik hisob-kitobni amalga oshirish uchun dastur yozishga hojat bo'lmaydi. Buning o'rniga formula muharriri yordamida matematik ifodalarni shunchaki kiritish kifoya. Shuni ta'kidlash lozimki, natija shu zahotiyoq olinadi. Bundan tashqari, printerda hujjatning bosma nusxasini ishlab chiqish yoki MathCadda ishlash vaqtida hujjat kompyuter ekranida qanday ko'rinishda bo'lsa, shu ko'rinishda Internetda sahifasini yaratish mumkin. MathCad yaratuvchilari shunday ishlarni amalga oshirdilarki, natijada dasturlash bo'yicha maxsus bilimga ega bo'lmagan foydalanuvchilar ham zamonaviy hisoblash fanlari va kompyuter texnologiyalari yutuqlaridan bimalol foydalana oldilar. MathCad muharririda samarali ishlash uchun foydalanuvchining baza ko'nikmalari yetarli. Boshqa tomondan, professional dasturchilar MathCaddan turli dasturiy hisoblar yaratish, MathCad imkoniyatlarini kengaytirish orqali ko'proq ma'lumot oladilar. Real hayot muammolariga mos ravishda matematiklarga quyidagi masalalarni yechishga to'g'ri keladi:

- kompyuterga turli matematik ifodalarni kiritish (hujjatlarni yaratish yoki keying hisoblash uchun, prezentatsiya, Web-sahifa yaratish);

- matematik hisob-kitoblarni amalga oshirish;
- hisob-kitob natijalarini grafik ko'rinish tayyorlash;
- berilgan kattaliklarni kiritish va natijalarni matnli yoki boshqa formatdagi ma'lumotlar bazasi fayliga chiqarish;

- ish hisobotini bosma hujjat ko'rinishida tayyorlash;
- Web-sahifa tayyorlash va natijani Internetda chop etish;
- matematika sohasi bo'yicha turli ma'lumotlar olish.

MathCad bu barcha masalalar bilan bemaolol ishlay oladi:

- matematik ifoda va matn MathCad formulalar muharriri yordamida kiritiladi. U imkoniyatlarini ishlatishdagi soddaligi bilan Microsoft Word formulalar muharriridan qolishmaydi;

- matematik hisob-kitoblar kiritilgan formulalar yordamida zudlik bilan bajariladi;

- juda boy formatlash imkoniyatiga ega bo'lgan turli grafiklar bevosita hujjatlarga qo'yiladi;

- fayllarga turli formatlardagi ma'lumotlarni kiritish va chiqarish mumkin;

- hujjatlar bevosita MathCaddan foydalanuvchi kompyuter ekranida qanday ko'rsa, shunday ko'rinishda bosmaga chiqarilishi yoki keyinchalik tahrirlash uchun RTF formatda saqlanishi mumkin. Bu hujjatlar kuchli matn muharrirlarida tahrirlanishi (masalan, MS Wordda) hamda Web-sahifa formatlarda saqlanishi mumkin;

- belgili hisoblashlar turli yordamchi matematik ma'lumotlarni bir zumda olish, yordamchi tizim, Resurslar Markazi va elektron kitoblar esa turli kerakli ma'lumotlarni tez topish imkonini beradi.

Shunday qilib, MathCad tarkibiga bir-biri bilan o'zaro integrallashgan komponentlar kiradi. Bu komponentlar matn va formulalarni kiritish va tahrirlash imkonini beradigan quvvatli matn muharrirlari, kiritilgan formulalar asosida hisob-kitobni amalga oshiradigan hisoblash protsessori, sun'iy intellekt tizimi hisoblangan belgili protsessorlardir. Bu komponentlar birlashmasi ish natijasini hujjatlashtirish bilan birga, turli matematik hisob-kitoblar uchun qulay hisoblash muhitini yaratadi.

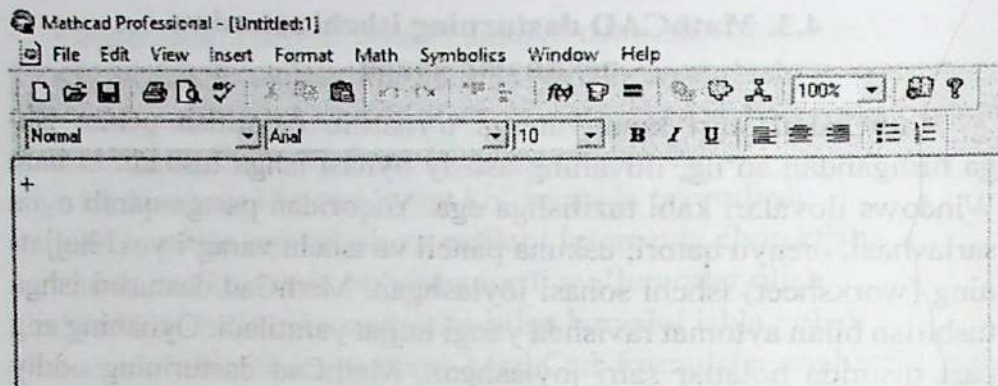
4.3. MathCAD dasturning ishchi sohasi va uskunalar paneli

MathCad dasturi kompyuterga o'rnatilib bajarilish uchun ishga tushgandan so'ng, ilovaning asosiy oynasi ishga tushadi. U ham Windows ilovalari kabi tuzilishga ega. Yuqoridan pastga qarab oyna sarlavhasi, menyu qatori, uskuna paneli va ishchi varag'i yoki hujjatning (worksheet) ishchi sohasi joylashgan. MathCad dasturini ishga tushirish bilan avtomat ravishda yangi hujjat yaratiladi. Oynaning eng past qismida holatlar satri joylashgan. MathCad dasturining oddiy matn tahrirlovchisi bilan o'xshashligi inobatga olinsa, siz asboblarni panelidagi ko'pgina tugmalarning vazifasini tushunib olasiz.

Dasturni ishga tushirish bilan old planda **Tip of the Day** (Kun maslahati) muloqot oynasi ham paydo bo'ladi. Uni **Close** (Yopish) tugmasini bosish bilan o'chirishimiz mumkin. MathCadning kun maslahati deb ataluvchi opsiyasini o'chirish ushuni, uning muloqot oynasidan **Show tips on startup** (Maslahatni ko'rsatish) bayroqcha ko'rinishini olib tashlash kerak. **Next Tip** (Keyingi maslahat)ni bosib, keyingi maslahatni ko'rish mumkin. MathCadda, shuningdek, **Resource Center** (Resurslar markazi) nomli yana bir oynani ko'rish mumkin. Bunda turli matematik, fizik, muhandislik masalalarini bajarish mumkin. O'z ishingizda siz bunga e'tibor berishingiz yoki bermasligingiz va undan yordamchi ma'lumot sifatida foydalanishingiz lozim.

Tip of the Day oynasini yopish bilan bo'sh hujjatga ifodalarni kiritishni boshlash mumkin. Kiritishning eng oddiy usuli formulalarni klaviatura orqali kiritishdir (lekin bu usul yangi o'rganayotganlar uchun yaxshi hisoblanmaydi). Formulalar yordamida oddiy hisob-kitoblarni bajarish uchun quyidagilarni bajaring:

- hujjatning kerakli nuqtasini sichqoncha bilan belgilab, ifoda paydo bo'lishi kerak bo'lgan joyni aniqlang;
- ifodaning chap tomonini kiriting;
- tenglik belgisini kiriting $\Leftarrow \Rightarrow$.



4.2-rasm. MathCADning oyna ilovasi.

Foydalanuvchining MathCad interfeysi intuitsiyaga asoslangan va Windowsning boshqa ilovalariga o'xshashdir. Uning tarkibiy qismlari:

- yuqori menyu yoki menyu qatori (menu bar);
- Standard (Standart) va **Formatting** (Formatlash) uskunalar paneli (toolbars);
- Math (Matematika) va matematik asboblarning panelining qo'shimcha asboblari;
- ishchi soha (worksheet);
- holatlar satri (status line yoki status bar);
- suzib chiquvchi yoki kontekst menyu (pop-up menus yoki context menus);
- muloqot oynasi yoki muloqotlar (dialogs).

Ko'pgina buyruqlarni menyu (yuqoridagi yoki kontekst) yordamida bajarish bilan birga, asboblarning paneli yoki klaviatura orqali ham bajarish mumkin.

Menyu:

Menyu qatori MathCad oynasining eng yuqori qismida joylashgan. Unda to'qqizta menyu nomlari joylashgan bo'lib, har birini shichqoncha bilan tanlash orqali harakatlari bo'yicha guruhlangan ro'yxatlarni chiqarish mumkin:

– File (Fayl) – hujjatlarni yaratish, ochish, saqlash, elektron pochta orqali jo‘natish va hujjatli fayllarni printerda chop etish bilan bog‘liq buyruqlar;

– Edit (Tahrirlash) – matnni tahrirlashga oid buyruqlar (nusxa olish, qo‘yish, o‘chirish);

– View (Ko‘rinish) – MathCad muharriri oynasida hujjatning tashqi ko‘rinishini boshqarish hamda animatsiya fayllarini yaratish buyruqlari;

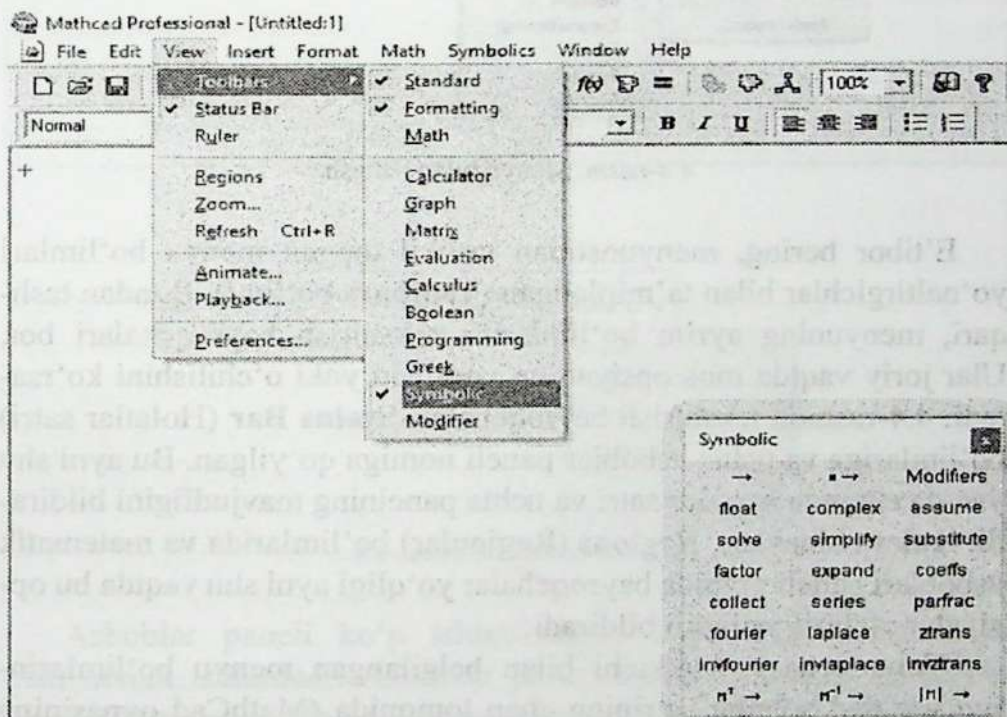
– Insert (Qo‘yish) – hujjatga turli obyektlarni qo‘yish buyruqlari;

– Format (Format) – matn, formula va grafikani formatlash buyruqlari;

– Math (Matematika) – hisoblash jarayonini boshqarish buyruqlari;

– Symbolics (Ramziy belgilar) – belgili hisoblash buyruqlari;

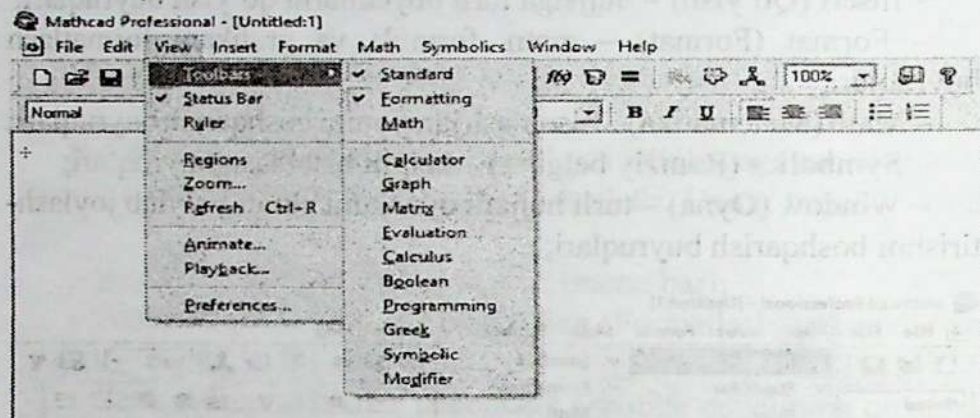
– Window (Oyna) – turli hujjatli oynalarni ekran bo‘ylab joylash-tirishni boshqarish buyruqlari;



4.3-rasm. Symbolic buyrug‘ini ekranga qo‘yish.

– Help (Yordam) – yordamchi ma'lumotlarni chaqirish buyruqlari.

Biror buyruqni tanlash uchun avval u joylashgan menyu ochiladi va kerakli element tanlanadi. Ayrim buyruqlar menyuning o'zida emas, balki ostmenyuda joylashgan. Bunday buyruqni bajarish uchun, masalan, ekranga **Symbolic** (ramziy belgilar) uskunalar paneli buyruqlarini chiqarish uchun, **View -> Toolbars** (Ko'rinish) bo'limi tanlanadi va paydo bo'lgan menyustidan **Symbolic** (ramziy belgilar) buyrug'i belgilanadi.

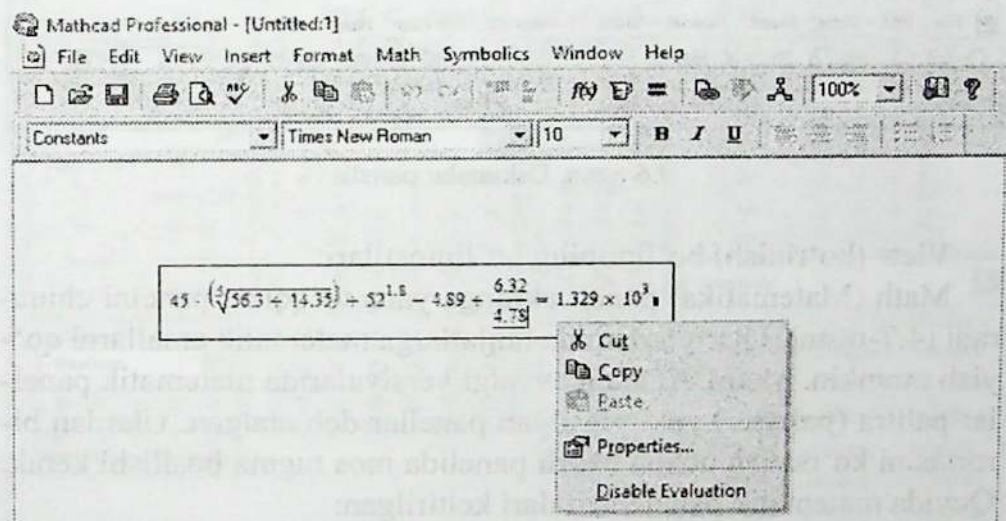


4.4-rasm. Menyu bilan ishlash.

E'tibor bering, menyustidan tashkil topgan menyu bo'limlari yo'naltirgichlar bilan ta'minlangan (**Toolbars** bo'limi). Bundan tashqari, menyuning ayrim bo'limlarida tekshirish bayroqchalari bor. Ular joriy vaqtda mos opsiyaning yopilishi yoki o'chilishini ko'rsatadi. 4.4-rasmda tekshirish bayroqchalari **Status Bar** (Holatlar satri) bo'limlariga va uchta asboblari paneli nomiga qo'yilgan. Bu ayni shu vaqtda ekranda holatlar satri va uchta panelning mavjudligini bildiradi. **Ruler** (Lineyka), **Regions** (Regionlar) bo'limlarida va matematik asboblari paneli nomida bayroqchalar yo'qligi ayni shu vaqtda bu opsiyalar o'chirilganligini bildiradi.

Sichqoncha ko'rsatkichi bilan belgilangan menyu bo'limlarining vazifasi holatlar satrining chap tomonida (MathCad oynasining pastki qismida) paydo bo'ladi. Ko'rsatkich **Symbolic** (ramziy bel-

gi) bo'limiga yo'naltirilgani uchun, holatlar satrining chap tomonida «Show or hide the symbolic keyword toolbar» (ramziy belgilar panelini ko'rsatish yoki yopish) paydo bo'ladi. Yuqorida joylashgan menyudan tashqari o'xshash vazifalarni sizib chiquvchi menyu ham bajaradi. Ular Windowsning boshqa ilovalaridagi kabi hujjatning ixtiyoriy qismida sichqoncha o'ng tugmasini bosish orqali paydo bo'ladi. Bu holda berilgan menyu tarkibi u chaqirilgan joyga bog'liq bo'ladi. Shuning uchun ular kontekst menyu deb ataladi. MathCad kontekstga qarab, joriy vaqtda qanday amallar kerak bo'lishini anglab oladi va menyuga mos buyruqlarni joylashtiradi. Shuning uchun yuqoridagi menyuga nisbatan kontekst menyuni ishlatish qulaydir. Kontekst menyuda yuqorida joylashgan menyudagi singari qaysi buyruq qayerda joylashganini eslab qolish shart emas. Kontekst menyuda ham menyusti bor. MathCadning kontekst menyusi quyidagi ko'rinishga ega.



4.5-rasm. Kontekst menyu.

Asboblari paneli ko'p ishlatiladigan buyruqlarni tezda bajarish uchun ishlatiladi. Asboblari paneli orqali bajariladigan barcha harakatlarni yuqorida joylashgan menyu buyruqlari orqali bajarish mumkin. Menyu qatoridan pastda joylashgan uchta asosiy asboblari

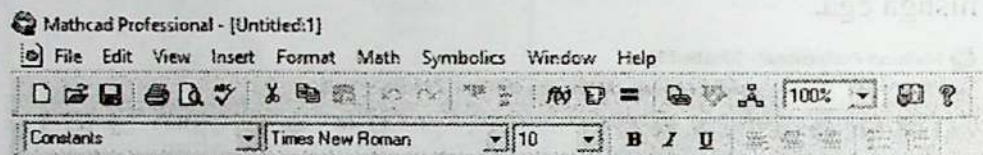
paneli aks etgan MathCad oynasi tasvirlangan. Panellardagi tugmalar bajariladigan vazifalari bo'yicha guruhlarga ajratilga:

- Standard (Standart) – fayllar bilan ishlash, muharrirlik, obyektlarni qo'yish va yordamchi tizimlarga kirish kabi bir qancha amallarni bajarish uchun xizmat qiladi;

- Formatting (Formatlash) – matn va formulalarni formatlash uchun ishlatiladi (shrift o'lehami va turini o'zgartirish, tekislash kabi);

- Math (Matematika) – hujjatga matematik belgilar va operatorlarni qo'yish uchun xizmat qiladi.

Uskunalar panelidagi tugmalar guruhlari ma'nolari bo'yicha vertikal chiziqlar ajratuvchilar bilan bo'lingan. Ixtiyoriy tugma sichqoncha orqali faollashtirilsa, tugma yonida tugma vazifasini ko'rsatuvchi kichik matn sizib chiqadi. Holatlar satrida esa bajariladigan amal haqida to'liqroq ma'lumot chiqadi.



4.6-rasm. Uskunalar paneli.

View (ko'rinish) bo'limining bo'limostilari:

Math (Matematika) paneli ekranga yana to'qqizta panelni chiqaradi (4.7-rasm). Ular yordamida hujjatlarga matematik amallarni qo'yish mumkin. MathCADning avvalgi versiyalarida matematik panellar palitra (palettes) yoki yig'ilgan panellar deb atalgan. Ulardan birortasini ko'rsatish uchun **Math** panelida mos tugma bosilishi kerak. Quyida matematik panel vazifalari keltirilgan:

- Calculator (Kalkulyator) – asosiy matematik amallarni qo'yish uchun ishlatiladi, uning nomi tugmalari oddiy kalkulyatorniki kabi bo'lganligidan kelib chiqqan;

- Graph (Grafik) – grafiklarni qo'yish uchun xizmat qiladi;

- Matrix (Matritsa) – matritsa va matritsa ustida bajariladigan amallarni qo'yish uchun ishlatiladi;

malar birikmasi orqali bajarish qulay hisoblanadi, lekin katta tajriba talab qiladi.

Uskunalar panelini sozlash:

MathCadda Windowsning boshqa dasturlari kabi foydalanuvchi uskunalar panelining tashqi ko'rinishini o'ziga optimal bo'lgan ko'rinishga sozlash mumkin.

Siz :

– panellarni ko'rsatishingiz yoki berkitishingiz; panellarni ekranining ixtiyoriy joyiga ko'chirishingiz; formasini o'zgartirishingiz; panel tugmalarini aniqlab sozlashingiz mumkin.

Panellarni ekranga chiqarish:

Ekranga ixtiyoriy panelni chiqarish yoki uni berkitish **View** (ko'rinish) / **Toolbars** (uskunalar paneli) buyruqlarini tanlash orqali amalga oshiriladi. Ochilgan menyusidan kerakli panel nomi tanlanadi. Ekrandan ixtiyoriy panelni olib tashlash kontekst menyu yordamida ham bajarilishi mumkin. Buning uchun kontekst menyu sichqoncha o'ng tugmasini panelning ixtiyoriy joyida bosish orqali chaqiriladi va **Hide** (yashirish) buyrug'i tanlanadi. Bundan tashqari, agar panel suzuvchi bo'lsa, ya'ni asosiy oynaga biriktirilmagan bo'lsa, uni yopish tugmasi orqali berkitish mumkin. Asosiy panellardan farqli ravishda matematik panellarni **Math** panelidagi mos tugmalarni bosib chaqirish yoki berkitish mumkin. Matematik panellarning bor yoki yo'qligi mos tugmalar orqali ko'rinib turadi.

Suzib yuruvchi panellarni yaratish:

Ixtiyoriy panelni MathCad oynasi chegaralaridan tortib olish uchun:

1) sichqoncha tugmasini panellarning birinchi yoki oxirgi ajratuvchisiga o'rnatish (birinchi ajratuvchi o'ziga xos ajralib turadigan ko'rinishga ega, oxirgisi esa oddiydir);

2) sichqonchani chap tugmasini bosib ushlab tursangiz, panel chiziqlarining o'ziga xos tasvirini ko'rasiz;

3) tugmani qo'yib yubormasdan panelni sudrang (panelni ekran-dagi ko'chiriladigan joyini aniqlagan holda sichqoncha ko'rsatkichini o'sha yerga ko'chiring);

4) sichqoncha tugmasini qo'yib yuboring, panel suzuvchi bo'lib qoladi va profili qayerda bo'lsa, o'sha yerga ko'chadi.

E'tibor bersangiz, suzuvchi asboblarni panellarida panel nomli sarlavha paydo bo'ladi. Panelni qayta oynaga biriktirish ichun, uni sarlavhasi yordamida oyna chegarasiga ko'chirish mumkin. Panel chegara tomon surilganda panel chegara tominidan tortilayotganligini ko'rish mumkin. Bu vaqtda sichqoncha tugmasini qo'yib yuborilsa panel suzuvchi bo'lmay qoladi. Panellarni faqatgina oynaning yuqori qismidagi menyu satriga emas, balki ixtiyoriy chegaraga biriktirish mumkin.

MathCad oynasi chegarasidan panelni ajratishning yanada oson yo'li bor. Buning uchun uning birinchi yoki oxirgi ajratuvchilarida sichqoncha tugmasi ikki marotaba bosiladi. Panelni oynaga biriktirish uchun esa, uning sarlavhasida sichqoncha tugmasini ikki marotaba bosish kifoya.

4.4. MathCad tizimida ishlatiladigan ma'lumotlarning turlari

Ma'lumotlarning turlariga sonli konstantalar, oddiy va tizimli o'zgaruvchilar, massivlar (vektorlar va matritsalar) va fayl ko'rinishidagi ma'lumotlar kiradi.

O'zgarishi mumkin bo'lmagan nomga ega obyektlar, saqlanayotgan qandaydir qiymatlar konstanta deyiladi. Dasturni bajarish davomida qandaydir qiymatga ega bo'lgan nomga ega obyektlar o'zgarishi mumkin bo'lganda ularni o'zgaruvchilar deyiladi. O'zgaruvchining turi uning qiymati bilan aniqlanadi; o'zgaruvchilar son qiymatli, qatorli, belgili va h.k. bo'lishi mumkin. Konstantalar, o'zgaruvchilar va boshqa obyektlarning nomi identifikatorlar deb yuritiladi. MathCadda identifikatorlar lotincha yoki yunoncha harf va sonlarning to'plamidan iboratdir.

MathCadda uncha katta bo'lmagan maxsus obyektlar guruhi mavjudki, ularni konstantalar va o'zgaruvchilar klasslariga qo'shib bo'lmaydi. Ularning qiymatlari dastur ishga tushirilgan paytda aniqlanadi. Ularni tizim tomonidan ilgaridan boshlang'ich qiymatlari belgilangan tizimli o'zgaruvchilar deb atash to'g'riroq bo'ladi. Tizimli o'zgaru-

vhilar qiymatlarini o'zgartirish MathCadOptionsning Matematika -> Opsiylar buyrug'i bo'yicha bajariladi, bunda uning o'zgaruvchilar dialogli darchasining qo'yilmasidan foydalaniladi.

Oddiy o'zgaruvchilar tizimlaridan shunisi bilan farqlanadiki, ular foydalanuvchi tomonidan dastlab aniqlangan bo'lishlari, ya'ni kam deganda bir marta qiymat berilgan bo'lishlari lozim. Bu holda «:=» belgisi qo'llaniladi.

Agar o'zgaruvchi := operatori yordamida boshlang'ich qiymat berilsa, klaviaturadagi: klavishi bosilib u chiqarilsa, bunday belgilash lokal deyiladi. Bungacha o'zgaruvchi belgisi aniqlanmagan va uni qo'llash mumkin emas. Ammo, $\equiv$ belgisi yordamida (klaviatura ~ belgisi) global belgilashni ta'minlash mumkin. MathCad hujjatni ikki marta chapdan o'ngga va yuqoridan pastga to'liq o'qib chiqadi. Birinchi o'tishda ($\equiv$) lokal belgilash operatori tomonidan aniqlangan barcha harakatlar bajariladi, ikkinchi o'tishda esa ($:=$) lokal belgilash operatori belgilanib bergan harakatlar amalga oshiriladi va hisoblashlarning barcha zaruriy natijalari ($=$) qayd etiladi.

Bundan tashqari, quyuq tenglik = belgisi (Ctrl+=) ham mavjud bo'lib, u tenglamalar tizimini yechishda taxminiy tenglik operatori sifatida qo'llanadi. Yana simvolli tenglik belgisi $\rightarrow$ (Ctrl+) ham mavjud.

Lokal va global o'zgaruvchilar boshqa dasturlash tillarida bo'lganidek, MathCadda ham lokal va global o'zgaruvchilar farqlanadi. «:=» ramzi bilan MathCadda lokal o'zgaruvchilar belgilanadi. Buning uchun «:=» ramzi kiritilsa kifoya.

Global o'zgaruvchi «o'zgaruvchi $\equiv$ ifoda». Global o'zgaruvchilarning lokal o'zgaruvchilardan farqi ularning hujjatining istalgan joyida qo'llana olishidir.

4.5. MathCad tizimida funksiyalarning ishlatilishi

Matematik hisoblashlarda foydalaniladigan funksiyalar muhim instrument hisoblanadi. Birgina formula orqali, ammo turli boshlang'ich ma'lumotlar bilan ko'pkarrali hisoblashlarni amalga oshirishda ularni qo'llash maqsadga muvofiqdir.

Xususiy funksiyadan foydalanish uchun:

1) funksiyani yozish;

2) bajarish uchun yozilgan funksiyani chaqirish kerak.

Funksiyani aniqlash uchun identifikatorlar qo'llanadi. Funksiya nomi va funksiya parametrlarining formal ismlari. Formal parametr – aniq qiymati uni funksiyaga qaratilganda aniq parametriga mos qiymatga almashtirganda aniqlashtiriladigan identifikatordir.

Funksiyani aniqlash formati:

Foydalaniladigan funksiyani chiqarish istalgan standart funksiyani chiqarish kabidir.

Natijani alohida o'zgaruvchiga joylash mumkin:

O'zgaruvchi_nomi_natija:=funksiya_nomi (formal parametrlar ro'yxati).

Yoki bosmalash:

Funksiya_nomi (formal parametrlar ro'yxati)=

Namuna: Koordinata boshidan berilgan nuqttagacha masofani qaytaradigan disfunksiyasi aniqlansin. Hisoblash uchun A (1,96; 3,8) va B (6; 42,5) masofalar belgilansin.

Chiziqli algebra kursidan ma'lumki, koordinata boshidan qandaydir A (x, y) nuqttagacha bo'lgan masofa $d = \sqrt{x^2 + y^2}$ formulasi orqali aniqlanadi. Bu yerda (x, y) – berilgan nuqtaning koordinatalari. Mana shu formula **Dist** funksiyasining asosini tashkil qiladi. Funksiyani yozishda ikkita formal parametrni – nuqtaning koordinatalarini ko'zda tutish kerak bo'ladi. Mana shu parametrlar o'rniga berilgan nuqtalarning koordinatalari kiritilishi kerak. **Dist** funksiyasi quyidagi ko'rinishda yozilishi mumkin:

$$\text{Dist}(x, y) := \sqrt{x^2 + y^2}.$$

Berilgan nuqtalardan masofani hisoblash uchun funksiya quyidagicha ifodalanadi:

$$\text{Dist}(2.96-3.8)=2.277;$$

$$P:=\text{Dist}(5.42.8);$$

$$P=32.932.$$

Ikkinchi holatda natija yordamchi o'zgaruvchiga joylashadi.

Berilgan oraliqdagi qiymatlarni qabul qiluvchi o'zgaruvchilarni aniqlash:

MathCad tizimida berilgan oraliqdagi qiymatlarni qabul qiluvchi o'zgaruvchilarni aniqlash imkoniyati taqdim etilgan, shu bilan bir qatorda qo'shni qiymatlar bir-biridan teng masofalarda uzoqlashgan. Bu holda boshlang'ich, keyingi va oxirgi qiymatlar berilgan.

Mana shu xildagi o'zgaruvchilarda faqat indeksiz identifikatorlarni qo'llash mumkin:

O'zgaruvchi nomi:=boshlang'ich qiymat, boshlang'ich qiymat+qadam... oxirgi qiymat

Berilgan qadam qiymatida oxirgi qiymatga aniq erishilmasa, o'zgaruvchining berilgan oraliqdagi oxirgi qiymatdan katta bo'lgan eng katta qiymati qabul qilinadi.

Bundan tashqari, MathCad agar qadam 1 yoki -1 qiymatlarga mos kelgan taqdirda keyingi qiymatni bermaslik imkoniyatiga ega.

Bu holda o'zgaruvchini aniqlash formati quyidagi ko'rinishda taqdim etiladi:

O'zgaruvchi nomi:=boshlang'ich qiymat... oxirgi qiymat

4.6. MathCad tizimida masalalarni yechish usuli va vositalari

Konstantalarning qo'llanish turlari:

MathCad tizimida ma'lumotlar turining quyidagilari ko'zda tutilgan:

- 1) butun (2, -54, +43);
- 2) kasr (1.3, -2.23);
- 3) kompleks ($2.5+7i$). Bu holda shuni ko'zda tutish lozimki, ko'rinishidagi «yolg'on bir»ni yozishda paneldagi «**Calculus**» maxsus tugmachasidan foydalanish kerak;
- 4) qatorli. Odatda bu «yig'indini hisoblash» ko'rinishidagi sharhidir;
- 5) tizimli. Bu turdagi konstantaga misol tariqasida ε yoki π qiymatlarini ko'rsatish mumkin.

Sodda hisoblashlar uchun, masalan, sinus funksiyasini biror qiymatda hisbolash uchun $\sin(1/4) =$ ifodani klaviaturadan kiritish kifoya qiladi. Tenglik belgisi bosilgandan so'ng ifodaning o'ng tomonida quyidagi ko'rinishda natija paydo bo'ladi:

$$\sin\left(\frac{1}{4}\right) = 0.247.$$

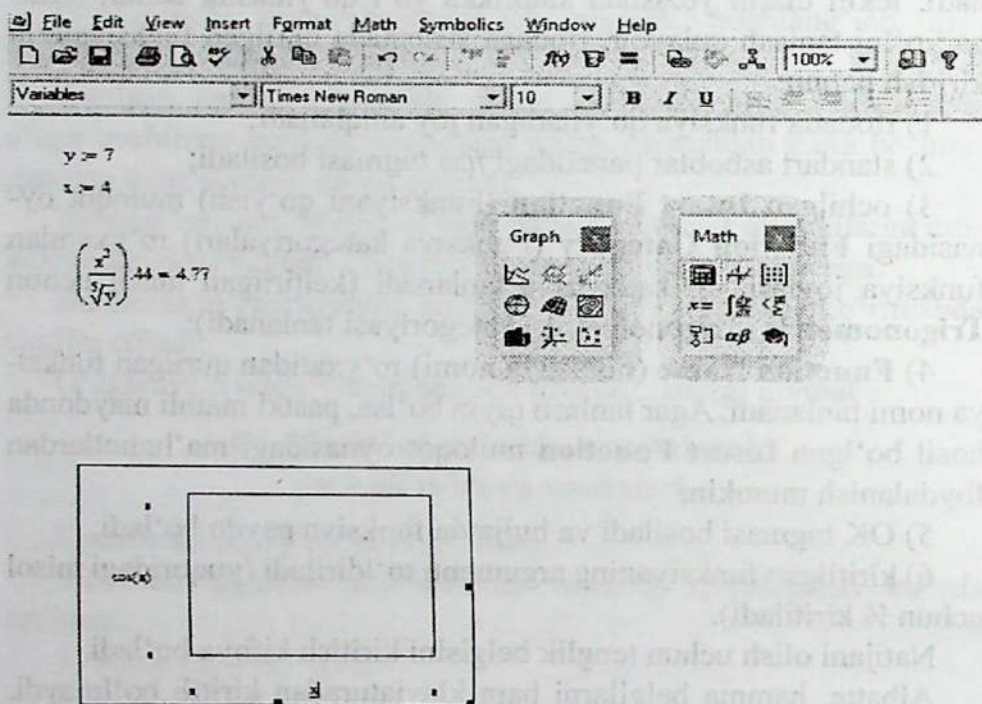
Xuddi shunday qilib, murakkab va katta hisoblashlarni ham MathCadning o'zida qurilgan maxsus funksiyalar orqali bajarish mumkin. Funksiya nomlarini klaviaturadan kiritish birmuncha oson hisoblanadi, lekin ularni yozishda xatolikka yo'l qo'ymaslik uchun boshqa yo'lni tanlash qulaydir. Ifodaga sistemada qurilgan funksiyalarni kiritish uchun:

- 1) ifodada funksiya qo'yiladigan joy aniqlanadi;
- 2) standart asboblarning panelidagi $f(x)$ tugmasi bosiladi;
- 3) ochilgan **Insert Function** (Funksiyani qo'yish) muloqot oynasidagi **Function Category** (Funksiya kategoriyalari) ro'yxatidan funksiya joylashgan kategoriya tanlanadi (keltirilgan misol uchun **Trigonometric** (Trigonometrik) kategoriyasi tanlanadi);
- 4) **Function Name** (Funksiya nomi) ro'yxatidan qurilgan funksiya nomi tanlanadi. Agar tanlash qiyin bo'lsa, pastki matnli maydonda hosil bo'lgan **Insert Function** muloqot oynasidagi ma'lumotlardan foydalanish mumkin;
- 5) OK tugmasi bosiladi va hujjatda funksiya paydo bo'ladi;
- 6) kiritilgan funksiyaning argumenti to'ldiriladi (yuqoridagi misol uchun $\frac{1}{4}$ kiritiladi).

Natijani olish uchun tenglik belgisini kiritish kifoya bo'ladi.

Albatta, hamma belgilarni ham klaviaturadan kiritib bo'lmaydi. Masalan, integral yoki differensial belgilarini. Ularni kiritish uchun MathCADda Microsoft Word formula muharriri vositalariga o'xshash maxsus uskunalar paneli bor. Ulardan biri **Math** (Matematika) panelidir. U hujjatga qo'yish mumkin bo'lgan tipik matematik obyektlar (operator, grafika, dastur elementlari) asboblari tashkil topgan. Bu panel 3.7-rasmda tahrirlanayotgan hujjat fonida katta planda ko'rsatilgan.

Panelda to'qqizta tugma bo'lib, har birini belgilash ekranda yana bir asboblardan panelning paydo bo'lishiga olib keladi. Bu to'qqizta panel yordamida MathCad hujjatlariga turli obyektlarni qo'yish mumkin. **Math** panelida yuqori chap tomondagi birinchi ikkita tugmaning bosilgan holatini ko'rish mumkin (chap tomondagisi ustida sichqoncha ko'rsatkichi joylashtirilgan). Shuning uchun ekranda yana ikkita – **Calculator** (Kalkulyator) va **Graph** (Grafik) panellari ko'rinib turibdi. Panellardagi tugmalarga qarab ularning nima uchun ishlatilishini osongina anglasak bo'ladi.

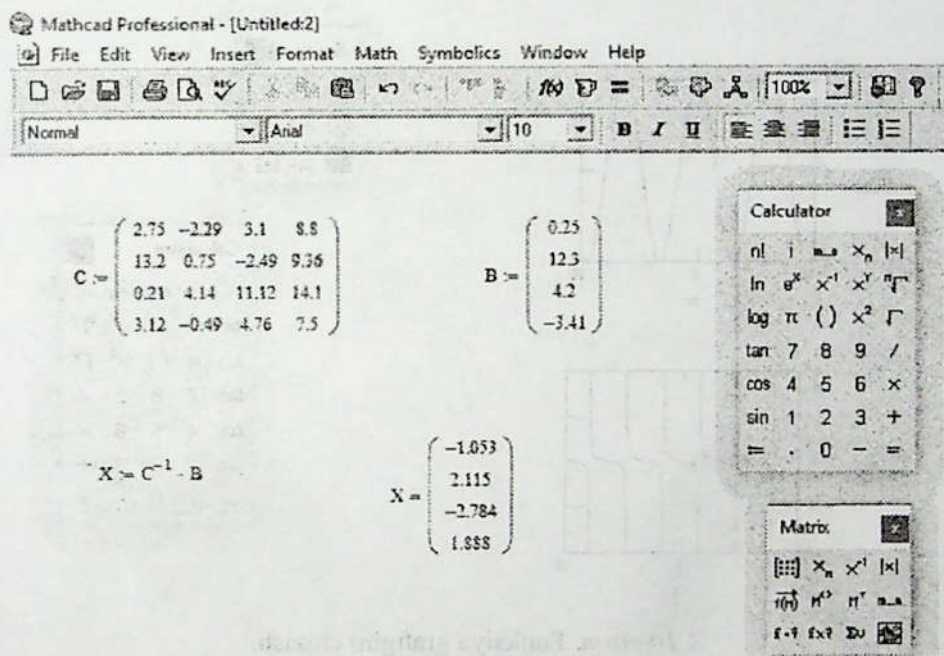


4.8-rasm. Math va Graph uskunalardan panelni ishlatish.

Masalan, **Calculator** paneli yordamida ifoda kiritilib, uni hisoblash kerak. Buning uchun avval sin tugmasini tanlaymiz (yuqoridan birinchi). So'ngra qavslar ichidagi qora to'rtburchak ko'rinishidagi belgini to'ldirish joyiga $\frac{1}{4}$ ifodasini **Calculator** paneli orqali kiritamiz va natijani olish uchun tenglik belgisini bosamiz.

Hujjatga matematik belgilarni Windowsning ko'pgina ilovalarida-gi kabi turli yo'l bilan qo'yish mumkin. MathCadda ishlash tajribasi va kompyuterda ishlash odatlarini hisobga olgan holda, foydalanuv-chi ulardan istalganini tanlashi mumkin.

Yuqorida aytib o'tilgan harakatlar MathCadni kengaytirilgan funksiyalardan iborat oddiy kalkulyator sifatida ishlatilishini namo-yon qildi. Matematiklar uchun foydalanuvchi funksiyalarida o'zga-ruvchi va amallarning berilish imkoniyati qiziqish uyg'otadi. Math-Cad uchun bu ishlar juda oddiy va «matematikada qanday bo'lsa, shunday kiritiladi» prinsipiga asoslangandir. O'zlashtirish hisoblash amalidan farq qilishi uchun tenglik belgisi bilan emas, «:=» belgisi bi-lan bajariladi. Bu belgining oddiy tenglikdan farqi qiymatni hisoblash chapdan o'ngga emas, balki o'ngdan chapga yo'naltirilganligidadir.

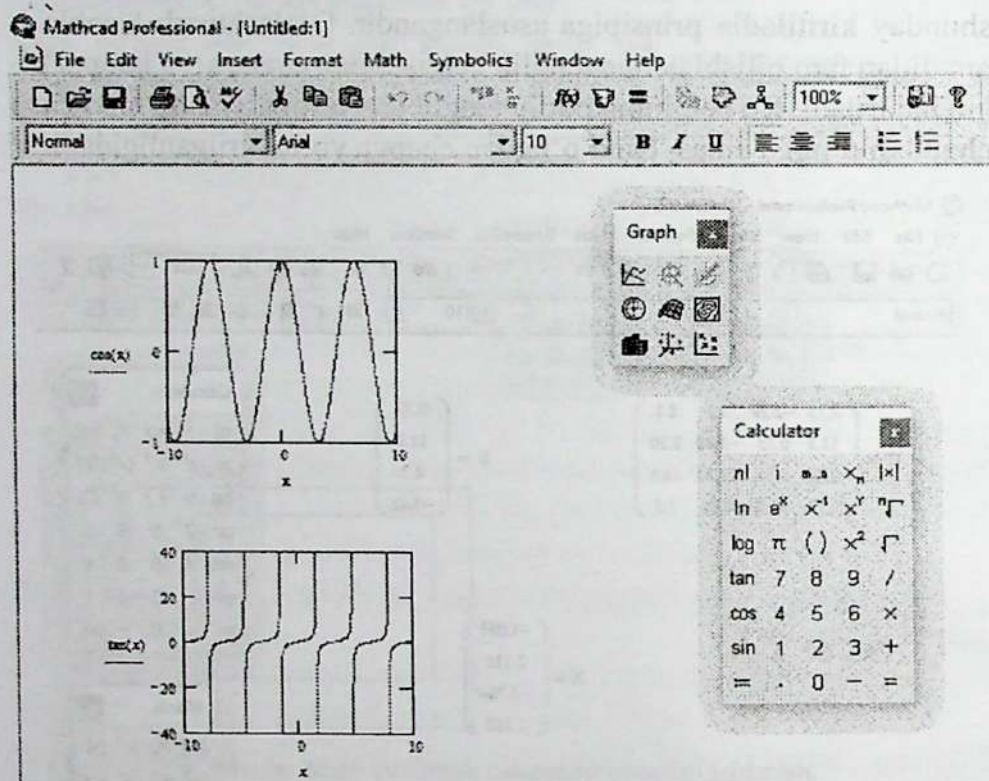


4.9-rasm. Matritsa usulida tenglamani yechish.

Funksiya grafigini chizish uchun, masalan, $F(x)$ ni, funksiyaning barcha parametrlarini va ularning qiymatlarini kiritish, so'ngra $F(x)$

funksiyasini kiritish va **Graph** panelida kerakli grafika turi tugmasini tanlash lozim va paydo bo'lgan tayyor grafikdan o'qlarda joylashuvchi qiymatlar aniqlanadi. MathCad dasturida ikki o'lchamli, uch o'lchamli grafik tasvirlarni ham hosil qilish mumkin. Ikki o'lchamli grafik tasvirni hosil qilish uchun quyidagi ketma-ketlik bajariladi:

Insert -> Graph -> X-YPlot->Graph panelidagi  tugmasi orqali. Dekort grafikaning shabloni tanlanadi.



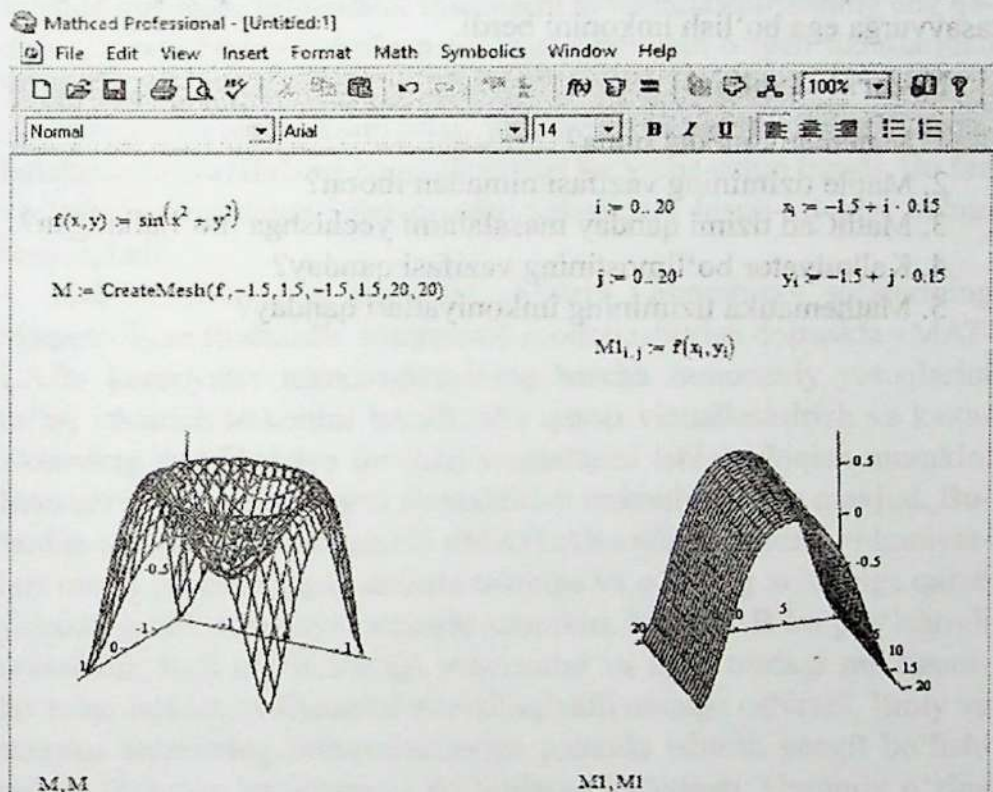
4.10-rasm. Funksiya grafigini chizish.

Uch o'lchamli yoki 3D grafika ikki o'zgaruvchan funktsiyani ko'rsatadi: $Z(X, Y)$. Uch o'lchamli grafik tasvirni yaratish uchun quyidagi amallar ketma-ketligi bajariladi:

MathCAD tizimida uch o'lchamli grafikani tuzishda yuzani matematik aniqlash kerak edi. Hozirda esa MathCADda *CreateMesh* funksiyasi qo'llaniladi.

• **CreateMesh** (*F* (yoki *G*, yoki *f1, f2, f3*), *x0, x1, y0, y1, xgrid, ygrid, fmap*) – ma'lum funksiyaning yuzasida to'r hosil qiladi, *F*, *x0, x1, y0, y1* – o'zgaruvchilar diapazonining o'zgarishi, *xgrid, ygrid* – o'zgaruvchilar to'ri hajmi, *fmap* – ko'rsatish funksiyasi. **CreateMesh** funksiyasi o'zgaruvchilar bilan yuzada to'r hosil qiladi – 5 dan 5 gacha va 20 va 20×20 nuqtali to'r.

CreateMesh funksiyasidan foydalanib 3D grafika tuzish (4.11-rasm) 1 ta usulda ko'rsatilgan. 4.11-rasmda bitta yuza har xil usul bilan tuzilgan, turli formatlash, yuza tegi va yuzaning o'zi ham kontur grafikada berilgan. Bu tuzilish rasimga katta taassurot beradi.



4.11-rasm. 3 o'lchovli grafik tasvir yaratish.

MathCadning o'ziga xos imkoniyatlaridan yana biri ko'p masalalarni analitik yechish imkonini beruvchi belgili hisoblashdir. Mualliflar fikricha, MathCAD matematikani yomon bo'lmagan olim darajasida biladi. Belgili protsessor intellektidan ustalik bilan foydalanish sizni ko'p miqdordagi integral va hosila kabi eskirgan hisoblashlardan ozod qiladi. Ifodalarning yozilish shakliga e'tibor bersangiz, birgina o'ziga xos xususiyati belgili hisoblashlarda tenglik belgisi o'rniga $\rightarrow$ belgisining ishlatilishidir. Uni MathCad muharriridagi **Evaluation** (Ifoda) yoki **Symbolic** (Ramziy belgi) panellaridan kiritish mumkin, integrallash va differensiallash belgilari esa **Calculus** (Hisoblash) panelidan kiritiladi.

Bu bo'limda MathCad tizimining o'ziga xos hisoblash imkoniyatlari ko'rib chiqildi. Keltirilgan misollar tizim vazifasi haqida yaxshigina tasavvurga ega bo'lish imkonini berdi.

Nazorat savollari

1. Matematik model nima?
2. Maple tizimining vazifasi nimadan iborat?
3. MathCad tizimi qanday masalalarni yechishga mo'ljallangan?
4. Kalkulyator bo'limostining vazifasi qanday?
5. Matematika tizimining imkoniyatlari qanday?

5-BOB. MATLAB DASTURLASHTIRISH TIZIMI

5.1. MATLAB tizimida statik va dinamik modellarni amalga oshirish

Murakkab matematik masalalarni bir necha tizimlar yordamida yechish eng yaxshi va mos vositalarni tanlash imkoniyatini beradi va olinadigan natijalarning ishonchliligini orttiradi. Shunday dasturlardan biri MATLAB tizimi hisoblanadi.

MATLAB – matritsali operatsiyalarni ilgari surish va qo‘llash asosida qurilgan, matematik hisoblarni avtomatlashtirishning eng qadimiy, puxta ishlab chiqilgan va vaqt sinovidan o‘tgan tizimlaridan biridir.

MATLABning imkoniyatlari juda keng va tizim ko‘pincha raqobatchilardan vazifalarni bajarish tezligi bo‘yicha ustun turadi. Bu fan va texnikaning deyarli har qanday sohasidagi hisob-kitoblar uchun amal qiladi.

«MATLAB» so‘zi inglizcha «Matrix Laboratory» so‘zlarining qisqartirilgan ifodasidir. Matematik modellashtirish doirasida «MATLAB» kompyuter texnologiyasining barcha zamonaviy yutuqlarini to‘liq ishlatish imkonini beradi, shu qatori vizuallashtirish va kattaliklarning audifikatsiya (ovozli) vositalarini ishlab chiqish mumkin. Internet orqali kattaliklarni almashtirish imkoniyati ham mavjud. Bulardan tashqari, foydalanuvchi «MATLAB» vositalarining imkoniyatlari orqali yechilayotgan masala talabiga va o‘zining xohishiga qarab shaxsiy grafik interfeys yaratishi mumkin. MATLAB ko‘p o‘lchovli massivlar, turli ko‘rinishdagi matritsalar va ko‘p turdagi ma’lumotlar bilan ishlash vositalarini muvaffaqiyatli amalga oshiradi. Ilmiy va texnika sohasining mutaxassislariga paketda ishlash yengil bo‘lishi uchun funksiya kutubxonasi bo‘limlarga bo‘lingan. Umumiy o‘ziga xos xarakterli model va funksiyalar «MATLAB» paketi asosiy yadro-

si tarkibiga kiradi. Aniq sohafa xos model va funksiyalar kengaytirilgan paket (Toolboxes) tarkibiga kiritilgan.

Shunday qilib, «MATLAB» birinchi navbatda matematik model-lashtirish vositasidir. U ilm va texnikaning barcha ko'zga ko'ringan sohalarida analiz va sintez nuqtayi nazaridan tadqiqot ishlarini olib borishni ta'minlaydi. Bunda paket tuzilishi model yaratishdagi ikki-ta asosiy bo'lgan analitik va imitatsion yondashuvlarni samarali qo'shish imkonini beradi.

MATLAB asosan quyidagi vazifalarni bajarish uchun ishlatiladi:

- algoritmlarni yaratish;
- modellashtirish;
- matematik hisoblashlar;
- ma'lumotlarni tahlil qilish, tadqiq qilish va vizuallashtirish;
- ilmiy va muhandislik grafikasi;
- ilovalarni ishlab chiqish va boshqalar.

MATLAB dasturlash tili yoki MATLAB tili – ma'lumotlarning matritsa ko'rinishida berilishi, hisoblash imkoniyatlari va grafik vositalarining kengligi nuqtayi nazaridan olganda, yuqori darajali algoritmik til hisoblanadi. Shu o'rinda, MATLAB tili faqat MATLAB muhitida dasturlar yaratish va ishlatish uchun xizmat qiladi.

MATLAB tizimida dasturlashning quyidagi turlari mavjud:

- protseduraviy;
- operatorli;
- funksional;
- mantiqiy;
- tarkibiy (modulli);
- obyektga yo'naltirilgan;
- vizual yo'naltirilgan.

Foydalanuvchilarni MATLABda yaratiladigan barcha dasturlari diskda saqlanadi va m-kengaytmaga ega, shu sababli ular m-fayllar deyiladi. m-fayllar ikki turga bo'linadi: function va scriptm-fayllar.

m-fayllar yaratishda MATLAB tilining quyidagi qoidalariga amal qilinishi lozim: o'zgaruvchilar e'lon qilinmaydi, metkalar ishlatil-

maydi, shartsiz o'tish operatori go to ishlatilmaydi, dastur tugallanganligi qayd qilinmaydi.

M-fayllar bilan ishlash quyidagilarni o'z ichiga oladi:

- Asosiy (script-fayl) va qism dastur (function-fayl)larni ishlab chiqish;

- MATLABda M-fayllarni yaratish, tahrirlash va saqlash;

- M-fayllarni ishga tushirish;

- M-fayllarni sozlash.

MATLAB tizimining tili matematik hisoblashlarni dasturlash sohasida har qanday mavjud yuqori darajadagi universal dasturlash tillaridan boyroqdir. U hozirgi vaqtda mavjud bo'lgan deyarli hamma dasturlash vositalarini amalga oshiradi, jumladan, obyektga mo'ljallangan va vizual dasturlashni ham. Umuman olganda, MATLAB tizimidan foydalanish tajribali dasturlovchilarga o'z fikrlari va g'oyalari amalga oshirish uchun cheksiz imkoniyatlar beradi.

5.2. MATLAB dasturining boshqa dasturiy tizimlar bilan integratsiyalashuvi

Murakkab matematik masalalarni bir necha tizimlar yordamida yechish eng yaxshi va mos vositalarni tanlash imkoniyatini beradi va olinadigan natijalarning ishonchliligini orttiradi.

Keyingi yillarda loyihachilar matematik tizimlarning integratsiyalashuviga va ulardan birgalikda foydalanishga katta e'tibor bermoqdalar.

MATLAB tizimi bilan keng tarqalgan matematik tizimlar (Mathcad, Maple V va Mathematic) integrallashuvi mumkin. Matematik tizimlarni zamonaviy matnli protsessorlar bilan birlashtirishga intilish ham mavjud. Masalan, MATLAB yangi versiyalarining vositasi – Notebook – Word 95/97/2000/XR 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'zgartirsa

bo'ladi. Masalan, boshlang'ich shartlarni o'zgartirib, masalani yechish natijalarining o'zgarishini kuzatish mumkin. MATLAB 6 da grafiklarni Microsoft PowerPoint slaydlariga eksport qilishning takomillashgan vositalari ham ko'zda tutilgan.

Matlabda 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-yo'naltirilgan dasturlash prinsiplariga asoslangan Simulink murakkab qurilmalarni yuqori aniqlikda modellash imkoniyatini beradi.

O'z navbatida, boshqa ko'plab matematik tizimlar, masalan, Mathcad va Maple MATLAB bilan obyektli va dinamik bog'lanishi mumkin. Natijada ular MATLABdagi matritsalar bilan ishlashning samarali vositalaridan foydalanishlari mumkin. Kompyuter matematik tizimlarining bunday integratsiyalashuv tendensiyasi shubhasiz keyinchalik ham davom etadi.

5.3. Matritsalar bilan ishlash imkoniyatlari

MATLAB tizimi vektorlar va matritsalar ustida murakkab amallarni bajaradi. Undan arifmetik va algebraik amallardan tashqari matritsalarini invertirlash, ularning xususiy qiymatlarini hisoblash, chiziqli tenglamalar sistemasini yechish, ikki va uch o'lchamli funksiyalarining grafiklarini olish va boshqa ko'plab amallarni bajaruvchi kuchli kalkulyator sifatida ham foydalanish mumkin. Oddiy son va o'zgaruvchilarga ham MATLABda 1×1 o'lchamli matritsa sifatida qaraladi. Shu sababli oddiy sonlar va massivlar ustida bajariladigan amallarning shakli va usullarida bixillikka erishilgan. Zarur hollarda vektor va matritsalar massivlarga aylantiriladi va ularning qiymatlari har bir element uchun hisoblanadi.

Ko'plab matematik tizimlar foydalanuvchi dasturlash bilan amalda shug'ullanmasdan o'z masalalarini yechishi uchun mo'ljallab yaratilgan. Lekin bunday yo'nalish boshlanishidanoq o'z kamchiliklariga

ega ekanligi, umuman olganda, xato ekanligi ravshan edi. Haqiqatan ham, ko'plab masalalar algoritmlarni yozishni soddalashtiruvchi va algoritmlarni yaratishning yangi usullarini beruvchi rivojlangan dasturlash vositalarini talab qiladi.

Bir tomondan, MATLAB ko'plab amaliy masalalarni yechish imkoniyatini beruvchi operatorlar va funksiyalarga ega. Ular yordamida ko'plab amaliy masalalarni yechish mumkin. Bunday masalalarni yechish uchun avvallari 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 ko'plab minglarni tashkil qiladi va uzluksiz ortib bormoqda.

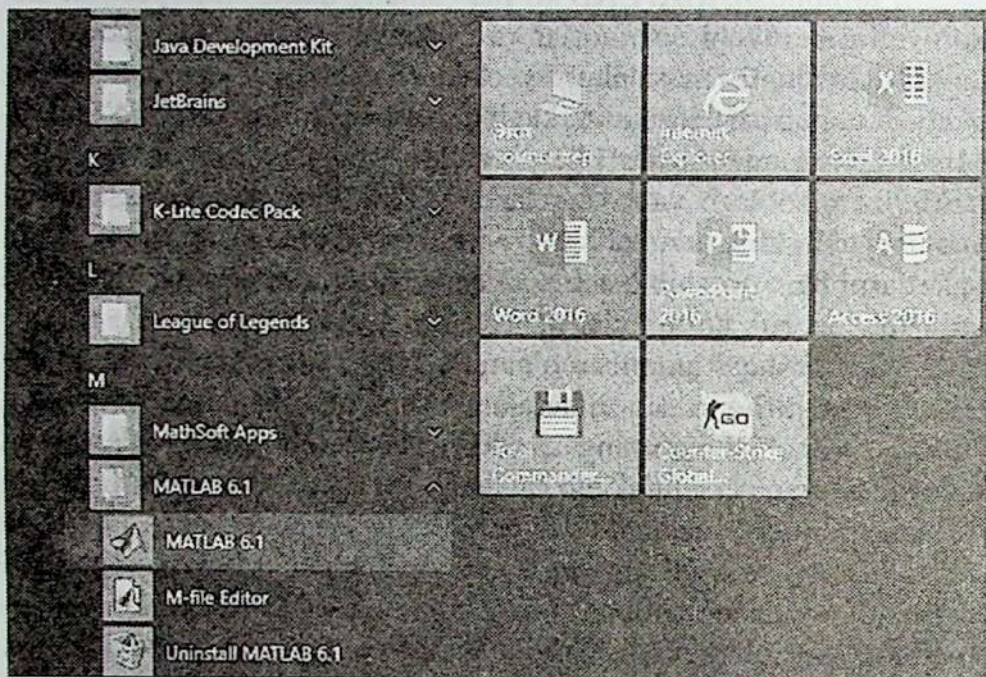
Lekin, boshqa tomondan olganda, MATLAB tizimi kuchli matematik-yo'naltirilgan yuqori darajali dasturlash tili sifatida yaratilgan. Bunday yo'nalish tizimning afzalliklaridan biri bo'lib hisoblanadi va uni yangi, yanada murakkab matematik masalalarni yechish uchun qo'llash mumkinligidan dalolat beradi.

MATLAB tizimi BASICga o'xshash (Fortran va Paskalning ayrim elementlari ham qo'shilgan) kirish tiliga ega. Dastur ko'plab kompyuterdan foydalanuvchilar uchun tanish bo'lgan an'anaviy usulda yoziladi. Bundan tashqari, tizim dasturlarni har qanday matn tahrirlagichi yordamida tahrirlash imkoniyatini beradi. MATLAB o'zining sozlagichli tahrirlagichiga ham ega.

MATLAB tizimining tili matematik hisoblashlarni dasturlash sohasida har qanday mavjud yuqori darajadagi universal dasturlash tilaridan boyroqdir. U hozirgi vaqtda mavjud bo'lgan deyarli hamma dasturlash vositalarini amalga oshiradi, jumladan, obyektga mo'ljallangan va vizual dasturlashni (Simulink vositalari yordamida) ham. Umuman olganda, MATLAB tizimidan foydalanish tajribali dasturlovchilar uchun o'z fikrlari va g'oyalarini amalga oshirish uchun cheksiz imkoniyatlar beradi.

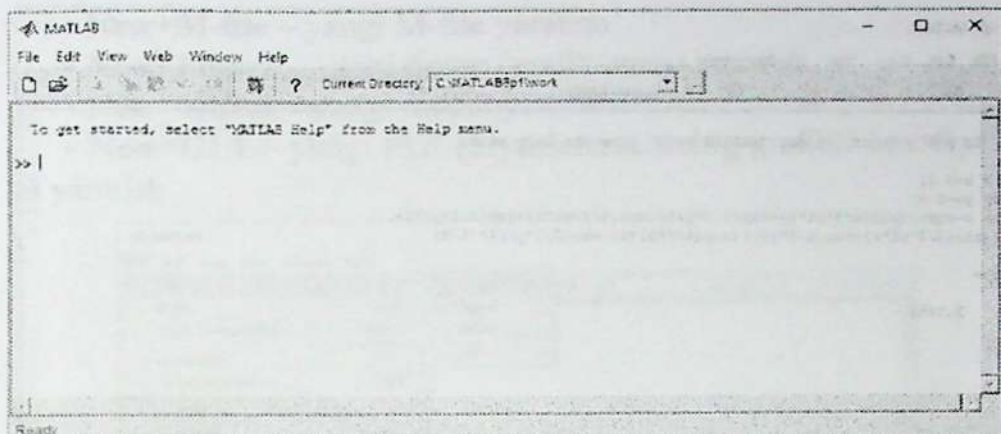
5.4. MATLAB tizimining dialog rejimida ishlash

Matlabni ishga tushirish uchun ishchi stolning pastki chap burchagida joylashgan Pusk (Start) tugmasi bosiladi va MATLABning o'rnatilgan versiyasi tanlanadi (5.1-rasm).



5.1-rasm. MATLABni ishga tushirish.

MATLAB ishga tushgandan keyin ekranda uning asosiy oynasi paydo bo'ladi (5.2-rasm) va u komandalar (buyruqlar) rejimida ishlashga tayyor holga keladi. Odatda bu oyna to'liq ochilmaydi va ekranning faqat bir qismini egallaydi. Ustki o'ng burchagida joylashgan uchta tugmadan o'rtadagisini bosish yo'li bilan oynani to'liq ochish mumkin. Chapdagi tugma bosilganda oyna yopiladi, o'ngdagi tugma bosilganda esa MATLABning ishlashi to'xtatiladi.



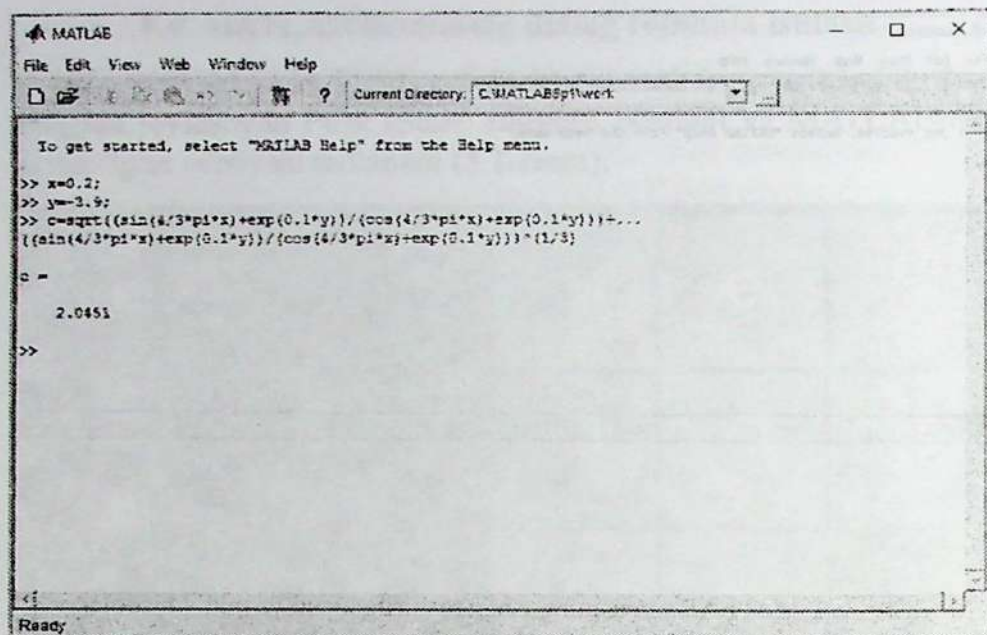
5.2-rasm. MATLAB oynasining ishga tushirilgandan va oddiy hisoblar bajarilgandan keyingi ko'rinishi.

MATLAB bilan ishlash seansini sessiya (session) deb atash qabul qilingan. Sessiya, mohiyati jihatidan, foydalanuvchining MATLAB tizimi bilan ishlashini aks ettiruvchi joriy hujjat bo'lib hisoblanadi. Unda kiritish, chiqarish satrlari va xatolar to'g'risida axborot bo'ladi. Agar buyruq yetarlicha uzun bo'lsa, u holda uch nuqta (...) qo'yilib, Enterni bir marta bosish orqali keyingi qatordan davom ettiriladi.

Masalan: quyidagi ifoda berilgan. Ifodaning natijasi topilsin.

$$C = \sqrt{\frac{\sin(\frac{4}{3}\pi x) + e^{0.1y}}{\cos(\frac{4}{3}\pi x) + e^{0.1y}}} + 3 \sqrt{\frac{\sin(\frac{4}{3}\pi x) + e^{0.1y}}{\cos(\frac{4}{3}\pi x) + e^{0.1y}}},$$

ifodaning $x = 0.2$ va $y = -3.9$ dagi qiymatini hisoblaymiz:



5.3-rasm. Ifodaning natijasini olish.

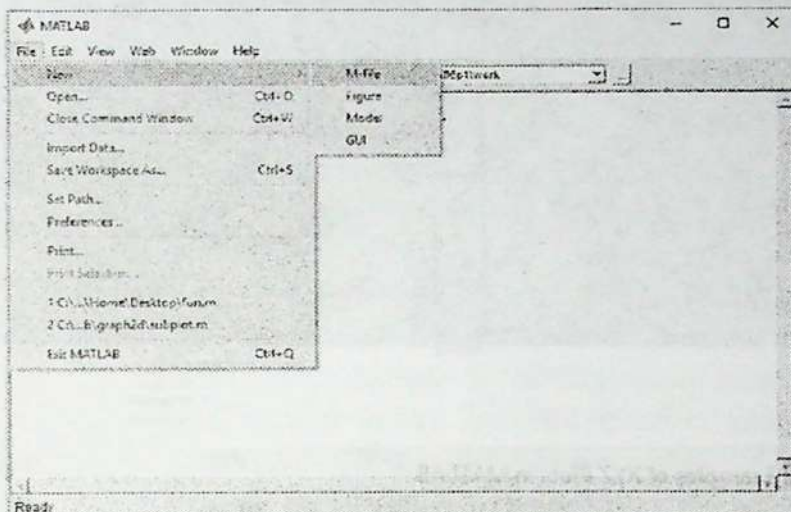
Xotiraning ishchi sohasida joylashgan sessiyaga kiruvchi o'zgaruvchilar va funksiyalarning tavsiflarini (sessiyani emas) .mat formatli fayl shaklida diskka Save (Saqlash) buyrug'i yordamida yozib olish mumkin. Load (Yuklash) buyrug'i yordamida ma'lumotlar diskdan ishchi sohaga yuklanadi. Sessiyaning fragmentlarini Diary (Kundalik) buyrug'i yordamida kundalik shaklida rasmiylashtirish mumkin.

Obyektga yo'naltirilgan dasturlash ham MATLAB tizimida keng qo'llaniladi va u grafika masalalarini dasturlashda dolzarb hisoblanadi.

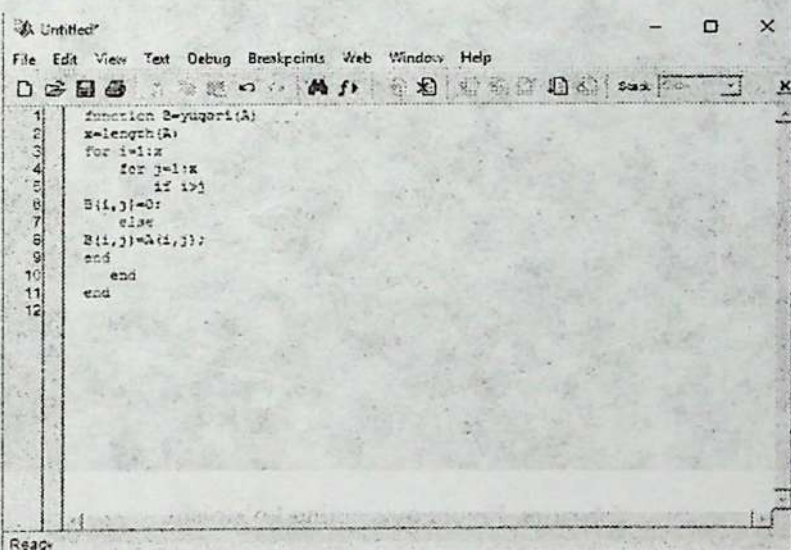
Vizual yo'naltirilgan dasturlash esa bloklar ko'rinishida berilgan qurilma va tizimlarni modellashga mo'ljallangan Simulink paketida o'z aksini topgan.

Fayllar bilan ishlaydigan standart buyruqlarni o'z ichiga olgan **File** menyusining 1-bandi **New** buyrug'i bo'lib, unda **M-file**, **Figure**, **Model**, **GUI** bandlari mavjud.

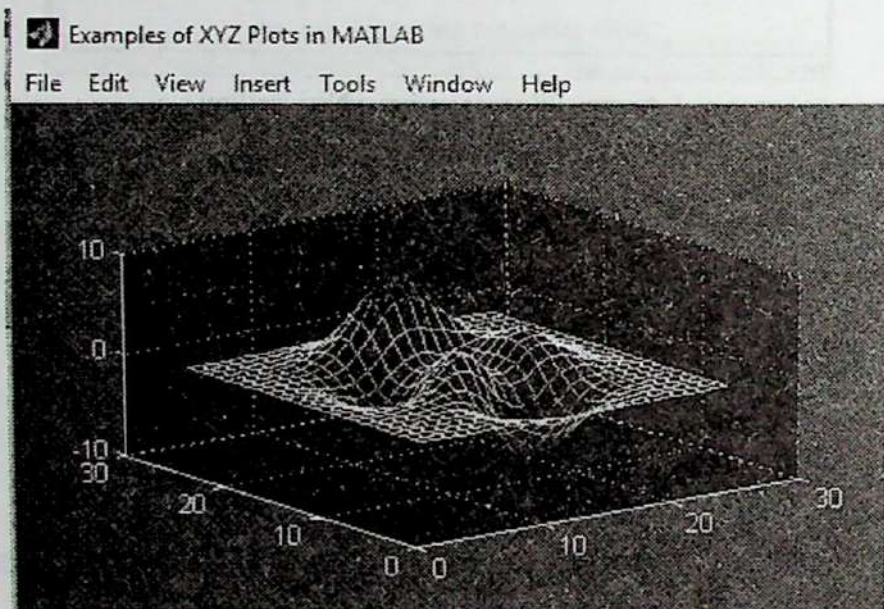
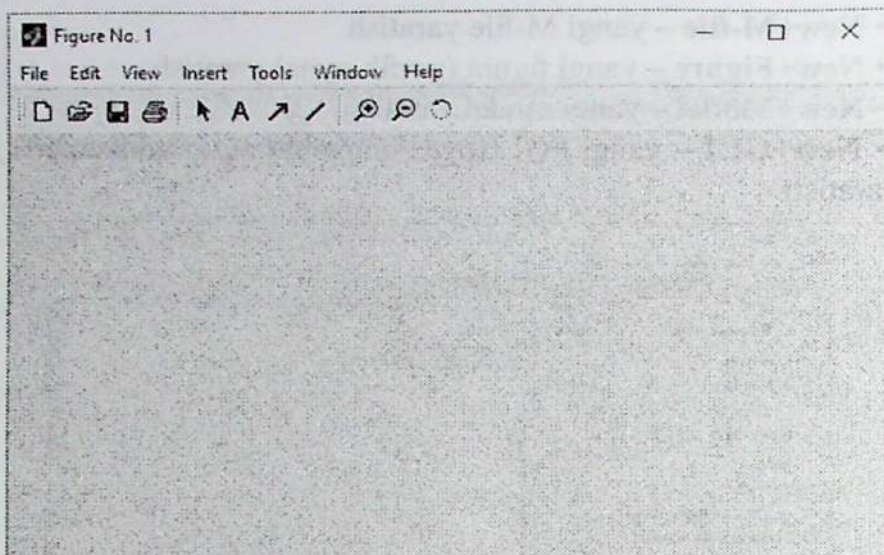
- **New+M-file** – yangi M-file yaratish
- **New+Figure** – yangi figura (grafik oyna) yaratish
- **New+Model** – yangi model yaratish
- **New+GUI** – yangi FGI (foydalanuvchining grafikli interfeysi) ni yaratish



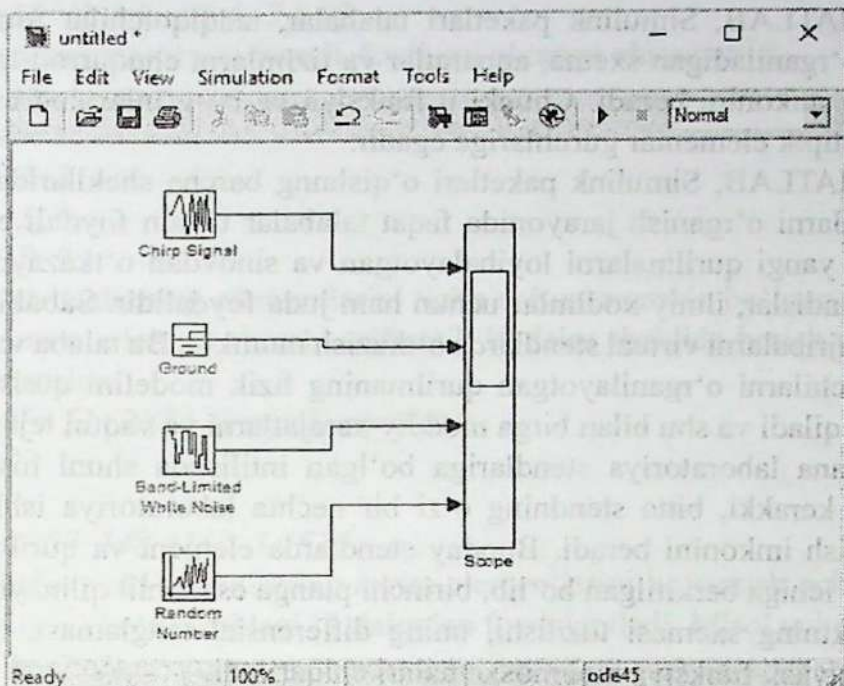
5.4-rasm. File menyusining New bo'limi.



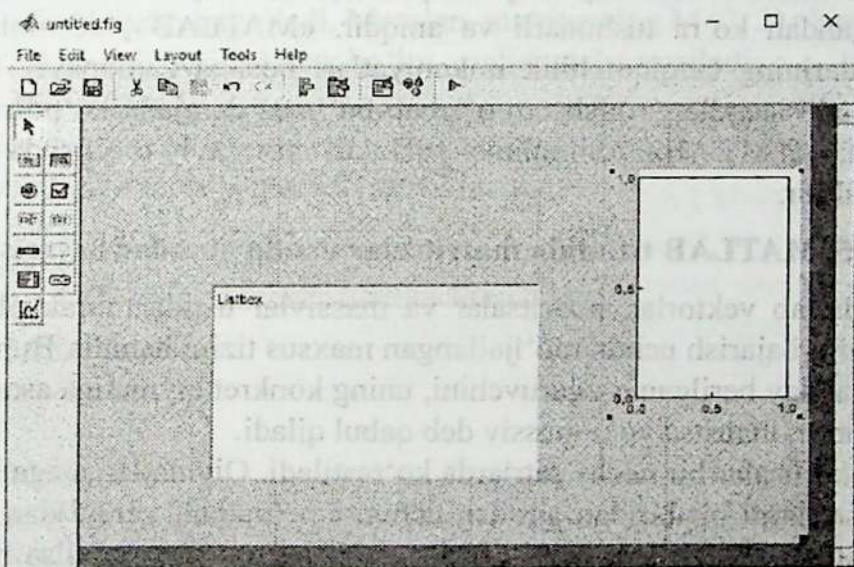
5.5-rasm. M-File oynasining ko'rinishi.



5.6-rasm. Figure oynasining ko'rinishi.



5.7-rasm. Model oynasining ko'rinishi.



5.8-rasm. GUI oynasining ko'rinishi.

MATLAB, Simulink paketlari talabalar, tadqiqotchilar tomonidan o'rganiladigan sxema, apparatlar va tizimlarni chuqurroq tadqiq qilish imkonini beradi. Chunki u funksiya va bo'g'inlarning asosiy keng tipik elementar guruhlariga egadir.

MATLAB, Simulink paketlari o'qishning barcha shakllarida turli fanlarni o'rganish jarayonida faqat talabalar uchun foydali emas, balki yangi qurilmalarni loyihalayotgan va sinovdan o'tkazayotgan muhandislar, ilmiy xodimlar uchun ham juda foydalidir. Sababi barcha tajribalarni virtual stendlarda o'tkazish mumkin. Bu talaba va tadqiqotchilarni o'rganilayotgan qurilmaning fizik modelini qurishdan ozod qiladi va shu bilan birga moddiy xarajatlarni va vaqtni tejaydi.

Yana laboratoriya stendlariga bo'lgan intilishda shuni hisobga olish kerakki, bitta stendning o'zi bir nechta laboratoriya ishlarini bajarish imkonini beradi. Bunday stendlarda element va qurilmalar stend ichiga berkitilgan bo'lib, birinchi planga esa tahlil qilinayotgan obyektning sxemasi tuzilishi, uning differensial tenglamasi, o'tish funksiyasi, funksiya mnemosxemalari chiqariladi.

Virtual laboratoriya ishlarining bajarilishi talabalar va tadqiqotchilar uchun zamonaviy ko'p maqsadli laboratoriya stendlarida bajarilgandan ko'ra tushunarli va aniqdir. «MATLAB», «Simulink» paketlarining tadqiqotchilik imkoniyatlari boshqa zamonaviy ko'p maqsadli stendlarga nisbatan o'lchab bo'lmas darajada ko'pdir. Nihoyat, «MATLAB», «Simulink» paketlari masofaviy o'qitishda juda foydalidir.

5.5. MATLAB tizimida matritsalar ustida amallar bajarish

Matlab vektorlar, matritsalar va massivlar ustida murakkab hisoblarni bajarish uchun mo'ljallangan maxsus tizim hamdir. Bunda u har qanday berilgan o'zgaruvchini, uning konkret qiymatiga asoslanib vektor, matritsa yoki massiv deb qabul qiladi.

Matritsalar bir necha satrlarda ko'rsatiladi. Qiymatlarga ega bo'lgan satrlarni bir-biridan ajratish uchun «;» (nuqtali vergul)dan foydalaniladi. Xuddi shu belgi kiritish satrining oxiriga qo'yilsa natija ekranga chiqmaydi. Misol uchun:

« $M = [1\ 2\ 3; 4\ 5\ 6; 7\ 8\ 9]$;

kvadrat matritsani beradi. Endi uni ekranga chiqaramiz:

« M

$M =$

1 2 3

4 5 6

7 8 9

Matritsalarining elementlarini tizim uchun mumkin bo'lgan funksiyalarni o'z ichiga oluvchi arifmetik ifodalar shaklida berish mumkin, masalan:

« $V = [2 + 2/(3+4) \exp(5) \sqrt{10}]$;

« V

$V =$

2.2857 148.4132 3.1623

Vektor yoki matritsaning ayrim elementlarini ko'rsatish uchun V (1) $M(i, y)$ ko'rinishidagi ifodalardan foydalaniladi. Misol uchun:

« $M(2, 2)$

ans = 5

Matritsaning $M(i, y)$ elementiga qiymat berish $M(i, y) = h$ ifodadan foydalanib bajariladi. Masalan, matritsaning $M(2, 2)$ elementiga 10 qiymatni berish kerak bo'lsa, quyidagicha yoziladi:

« $M(2, 2) = 10$

Bir indeksli $M(i)$ ifoda yordamida bitta ustunga yoyilgan matritsa elementlariga murojaat qilish mumkin:

« $M(2)$

ans =

4

« $M(8)$

ans =

6

« $M(9)$

ans =

9

« $M(5) = 100$;

«M

M=

1 2 3

4 100 6

7 8 9

Elementlari kompleks sonlar bo'lgan vektor va matritsalarini ham olish mumkin, masalan:

«i=sqrt (-1):

« SM = [1 2; 3 4] + i*[5 6; 7 8]

yoki

« SM = [1+5*i 2+6*i; 3+7*i 4+8*i]

Quyidagi matritsani hosil qiladi:

CM=

1.0000 + 5.0000i 2.0000 + 6.0000i

3.0000 + 7.0000i 4.0000 + 8.0000i

Matritsa va vektorlarning ayrim elementlari bilan bir qatorda ularning hamma elementlari (massivlar) ustida ham amallar bajarish mumkin. Buning uchun amal belgisining oldiga nuqta qo'yiladi. Masalan, * operatori vektorlar yoki matritsalarini ko'paytirishni anglatadi.

*operator esa massivning hamma elementlarini elementlararo ko'paytirishni bildiradi. Matritsa skalyarga ko'paytiriladigan M*2 va M.*2 ifodalar teng kuchli. Quyidagi misollarni ko'raylik:

>> M1=[1 2 3];

>> M2=[4 5 6];

>> M=M1*M2

??? Error using ==> *

Inner matrix dimensions must agree.

>> M=M1.*M2

M=

4 10 18

>> M*2

ans =

8 20 36

>> M.*2

```
ans =  
8 20 36  
>>
```

Bu yerda vektorlarni ko'paytirish $M=M1*M2$ ifodasida ko'paytirish belgisidan oldin nuqta qo'yilmaganligi uchun tizim xato to'g'risidagi *Inner matrix dimensions must agree* (Matritsalarining o'lchamlari o'zaro moslashishi kerak) degan axborotni berdi. Xato tuzatilgandan keyin to'g'ri natija olinadi.

5.6. Matritsalar ustida bajariladigan buyruqlar xususiyatlari

Matlabda matritsa va vektorlar ustida amallar bajarish bir vaqtning o'zida barcha arifmetik amallarni bajarish imkonini beradi. Buning uchun amal belgisi oldidan nuqta qo'yiladi. MATLABda vektor va matritsalarini berish uchun maxsus funksiyalar mavjud. Bu funksiyalar bir o'lchovli va ko'p o'lchovli massivlar yaratish uchun xizmat qiladi. *ones* funksiyasi massivning birlik elementini tuzadi.

```
>>a= ones (3,3) ;
```

```
a =
```

```
1 1 1
```

```
1 1 1
```

```
1 1 1
```

zeros funksiya nol elementli massivni yaratadi.

```
>>b = zeros (3, 3);
```

```
b =
```

```
0 0 0
```

```
0 0 0
```

```
0 0 0
```

Matritsani berish bir nechta satr va bir nechta ustunlarni ko'rsatishni talab etadi. Satr chegaralari nuqtali vergul bilan ajratiladi. Masalan:

```
>> M=[1 2 3; 4 5 6; 7 8 9];
```

```
>> M
```

```
M =
```

```
1 2 3
```


4 5 6

7 8 9

Matritsa va vektorlarning elementlarini arifmetik ifoda ko'rinishida ham kiritish mumkin. Masalan:

```
>> V=[2+2/(3+4),exp(5),sqrt(10)];
```

```
>> V
```

```
V =
```

```
2.2857 148.4132 3.1623
```

Vektor yoki matritsalarining alohida elementlarini ko'rsatish uchun V(i) yoki M(i,j) ko'rinishidagi ifodadan foydalaniladi. Masalan:

```
>> M(2,2);
```

```
ans =
```

```
5
```

```
>> M(3,3);
```

```
ans =
```

```
9
```

```
>> M(3,2);
```

```
ans =
```

```
8
```

Matritsalarini shakllantirish va matritsa ustida amallar bajarish uchun matritsaning alohida satr va ustunlarini o'chirish zarur bo'lishi mumkin. Buning uchun bo'sh kvadrat qavs, ya'ni [] dan foydalaniladi. Masalan, M matritsa bilan shu bajarib ko'raylik:

```
>> M=[1 2 3;4 5 6; 7 8 9];
```

```
>> M
```

```
M =
```

```
1 2 3
```

```
4 5 6
```

```
7 8 9
```

```
>> M(:,2)=[ ]
```

```
M =
```

```
1 3
```

```
4 6
```

```
7 9
```

Bunda ikkinchi ustun o'chirildi.

Chiziqli algebra masalalarini yechish sohasida MATLAB keng imkoniyatlarga ega. Vektor va matritsalar ustida bir qator amallarni MATLABda bajarishni keltirib o'taylik:

```
>> % matritsa elementlariga murojaat
```

```
>> A(1,3)+M(2,2)*A(1,2)-M(1,1)^2
```

```
ans =
```

```
19
```

```
>> % vektorlarni elementlari bo'yicha qo'shish
```

```
>> v1=[2 5 -1];
```

```
>> v2=[1 -1 3];
```

```
>> v1+v2
```

```
ans =
```

```
3 4 2
```

```
>> % vektor elementlarini songa qo'shish
```

```
>> v1+2
```

```
ans =
```

```
4 7 1
```

```
>> % vektor elementlari bo'yicha arifmetik amallar
```

```
>> 2*v2-v1/4
```

```
ans =
```

```
1.5000 -3.2500 6.2500
```

```
>> % vektor elementlari bo'yicha ko'paytirish
```

```
>> v1.*v2
```

```
ans =
```

```
2 -5 -3
```

```
>> % element bo'yicha darajaga ko'tarish
```

```
>> v1.^2
```

```
ans =
```

```
4 25 1
```

```
>> % element bo'yicha darajaga ko'tarish
```

```
>> v1.^v2
```

```
ans =
```

```
2.0000 0.2000 -1.0000
```



```
>> % elementlari bo'yicha songa ko'paytirish
>> v1./3
ans =
0.6667 1.6667 -0.3333
>> % elementlari bo'yicha vektorni vektorga bo'lish
>> v2./v1
ans =
0.5000 -0.2000 -3.0000
```

5.7. MATLAB tizimida matritsa elementlarini transponirlash

Matritsalarini transponirlash, ya'ni ustunlarini satrlari bilan almashtirish uchun ' operatoridan foydalaniladi. Ustunlaridagi elementlarning yig'indisi sum operatori yordamida hisoblanadi. Satrlaridagi elementlarning yig'indisini hisoblash uchun matritsa avval transponirlanadi va transponirlangan matritsaning ustunlaridagi elementlarning yig'indisi aniqlanadi. Buni quyidagi misollardan ham ko'rish mumkin:

```
>> B=[1 2 3;4 5 6;7 8 9]
B =
1 2 3
4 5 6
7 8 9
>> B.'
ans =
1 4 7
2 5 8
3 6 9
>> sum(B)
ans =
12 15 18
>> sum(B.')
```

6 15 24

Matritsalarining ustunlari va satrlarini o'chirish quyidagicha amalga oshiriladi.

Quyidagi misolni ko'raylik:

```
>> M=[1 2 3; 4 5 6; 7 8 9]
```

M =

1 2 3

4 5 6

7 8 9

```
>> M(:,2)
```

ans =

2

5

8

```
>> M(:,2)=[]
```

M =

1 3

4 6

7 9

Endi matritsaning ikkinchi satrini ikki nuqta operator (:) dan foydalanib o'chiramiz:

```
>> M(2,:)=[]
```

M =

1 3

7 9

5.8. MATLAB dasturlashtirish tizimida grafiklarni yaratish va formatlash. Ikki o'lchovli grafika

MATLABda ma'lumotlarni vizuallashtirish uchun yuqori darajali grafik imkoniyatlar mavjud. Barcha grafiklar, menyulari va uskunalar paneliga ega alohida grafik oynada tasvirlanadi. Bunda grafiklarni oynadagi menyular va uskunalar paneli elementlari orqali sayqallashtirish, kerakli aniqlikni oshirish hamda foydalanuvchi talabiga ko'ra o'zgartirish mumkin.

Ma'lumki, dekart koordinatalar sistemasida grafik chizish (x, y) juftligini qiymatlarini aniqlab, hosil bo'lgan nuqtalarni kesmalar bilan tutashtirish orqali hosil qilinadi. Demak (x, y) juftliklar soni qanchalik ko'p bo'lsa, grafik ham shunchalik silliq va aniqroq bo'ladi. Juftliklar avvaldan berilgan bo'lishi, ma'lum funksiyaning argumenti va qiymatlaridan hisoblab hosil qilinishi yoki tajriba o'tkazish natijasida olingan bo'lishi mumkin. Masalan:

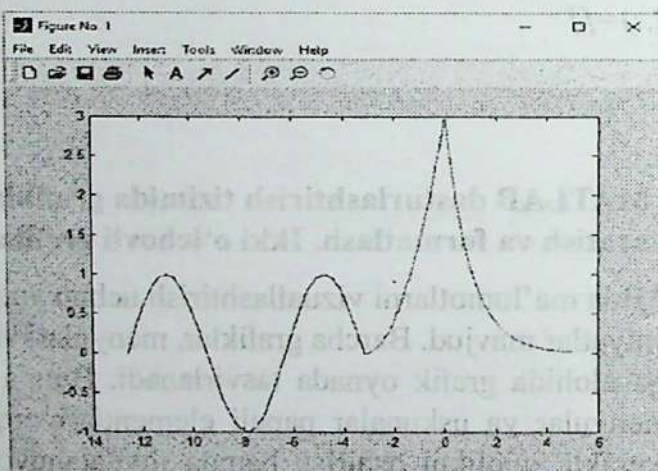
1-misol.

$$y(x) = \begin{cases} \sin x & -4\pi \leq x \leq -\pi \\ 3(x/\pi + 1)^2 & -\pi < x \leq 0 \\ 3e^{-x} & 0 < x \leq 5 \end{cases}$$

funksiyaning grafigini

chizish kerak bo'lsa, quyidagi MATLAB komandalari ketma-ketligi yetarli bo'ladi.

```
>> 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)
```

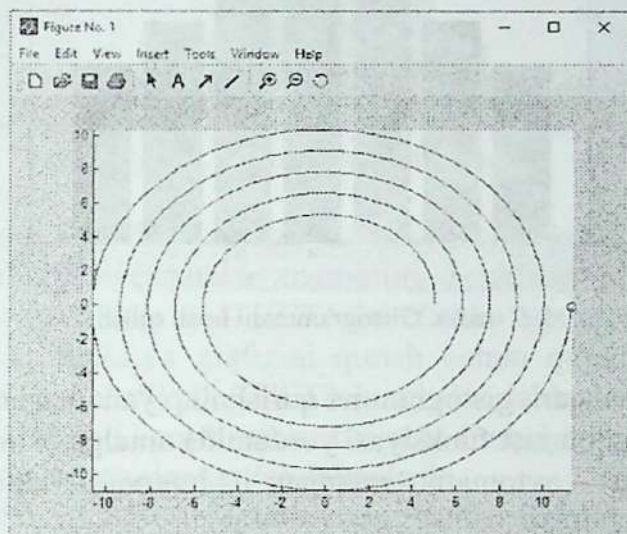


5.9-rasm. Funksiya grafigi.

`plot(x,y)` komandasi grafik oynani ochadi va unda kerakli funksiya grafigini chizib beradi. Yangi komandani e'lon qilish uchun kursorni komandalar oynasiga o'tkazish kerak.

2-misol. MATLAB muhitida quyidagi $x=\cos(5*t)*(t+5)$, $y=\sin(5*t)*(t+5)$, $x \in [0, 2\pi]$ funksiyaning grafigini yasang.

```
>> t=0:0.005:2*pi;  
>> x=cos(5*t).*(t+5);  
>> y=sin(5*t).*(t+5);  
>> comet(x,y)
```



5.10-rasm. Comet buyrug'i orqali olingan funksiya grafigi.

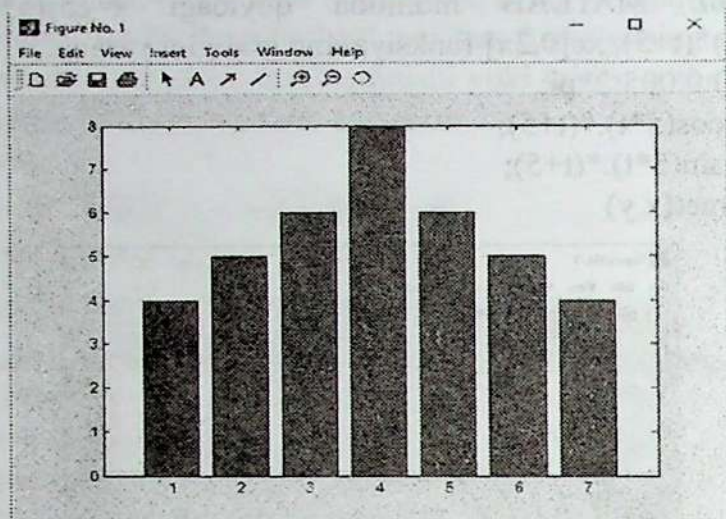
5.9. Gistogrammalar. Polyar koordinatalarda grafika

Amaliy hisoblarda biror vektor tarkibini tasvirlaydigan ustunli diagrammalar deb ataluvchi gistogrammalar ko'p uchraydi. Bunda vektorning har bir elementi balandligi uning qiymatiga mos bo'lgan ustun shaklida ko'rsatiladi. Ustunlar tartib raqamlariga va eng baland ustunning maksimal qiymatiga nisbatan ma'lum masshtabga ega bo'ladi. Bunday grafiklar, masalan, iqtisodiy o'zgarish va boshqa jarayonlarni ifodalashi mumkin. Ular `bar(a)` komandasi yordamida quriladi, masalan:


```
>> c=[4 5 6 8 6 5 4];
```

```
>> bar(c)
```

komandalar yordamida quyidagi gistogrammani olish mumkin:



5.11-rasm. Gistogrammani hosil qilish.

Bundan tashqari, gistogramma qurishning yana boshqa usuli ham mavjud bo'lib, bu hist funksiyasi yordamida amalga oshiriladi:

- $N=\text{hist}(u)$ – avtomatik tanlangan 10 intervalli vektor qiymatini qaytaradi;
- $N=\text{hist}(u,m)$ – xuddi yuqoridagi kabi, faqat M (M -skalyar) intervalda qaytaradi;

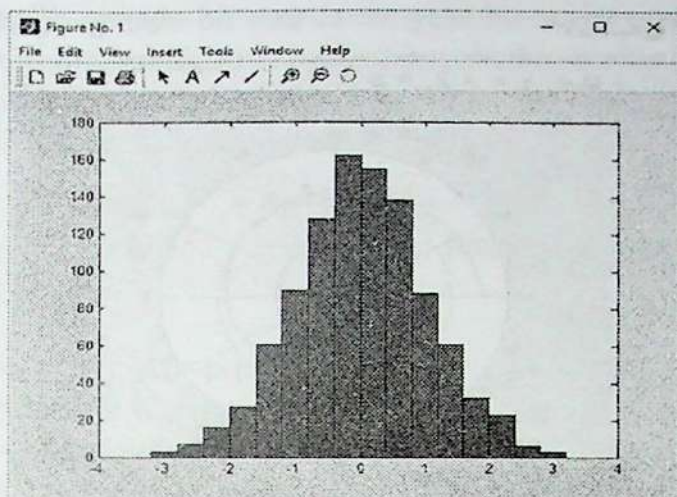
Quyidagi misolni ko'ramiz:

```
>> x=-3:0.4:3; y=randn(1000,1);
```

```
>> hist(y,x); h=hist(y,x)
```

h =

```
3 7 16 27 61 90 128 162 155 138 88 61 32 23 6 3
```



5.12-rasm. Gistogrammani hosil qilish.

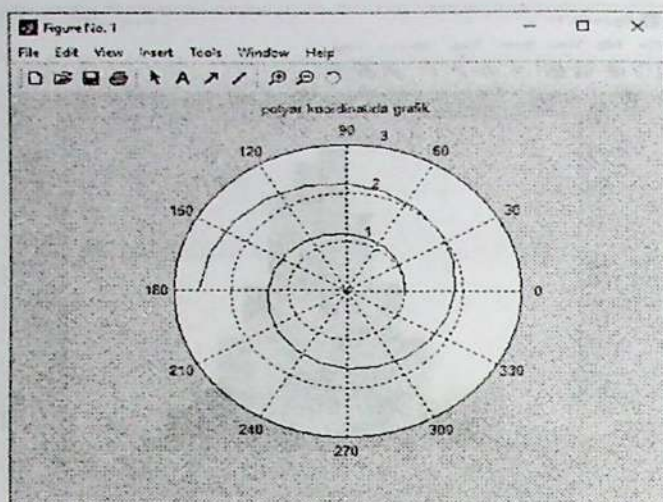
Qutbli koordinatalar tizimida ixtiyoriy nuqta xuddi radius vektor oxiri kabi, koordinatalar tizimining boshlang'ich nuqtasidan chiqib, RHO uzunlikka va THETA burchakka egaligini ko'rsatadi. RHO(THETA) funksiya grafigini qurish uchun quyida keltirilgan buyruqlardan foydalaniladi. THETA burchak odatda 0 dan $2 \cdot \pi$ gacha o'zgaradi. Qutbli koordinatalar tizimida funksiya grafigini qurish uchun quyidagi buyruqlardan foydalaniladi:

- `polar(THETA,RHO)` – qutbli koordinatalar tizimida radius-vektor oxirining o'z holatidagi RHO uzunlik bilan va THETA burchakni ko'rsatuvchi grafikni quradi;

- `polar(THETA,RHO, S)` – analogli avvalgi buyruqda ishtirok etgan, lekin S qatorli konstanta yordamida qurish uslubini analogli plot buyrug'i asosida ruxsat beradi.

Quyidagi misolni ko'ramiz:

```
>> angle=0:.1*pi:3*pi;
>> r=exp(angle/10);
>> polar(angle,r),...
>> polar(angle,r);
>> title('polyar koordinatida grafik');
>> grid on
```

5.13-rasm. Polar koordinatada grafik.

MATLABda ikki o'lovli 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.;

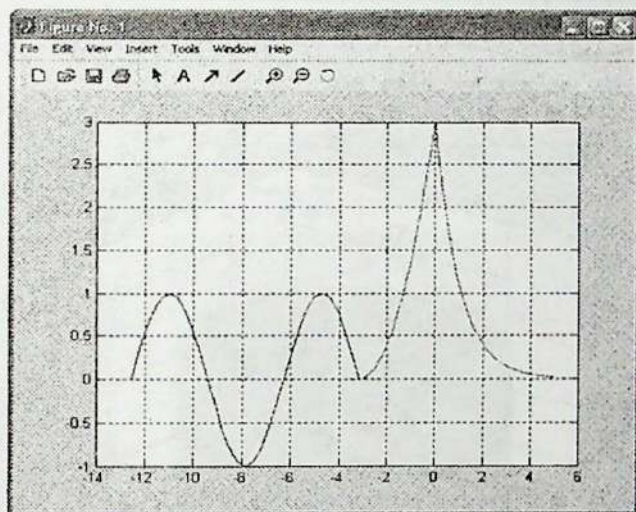
Odatda bir o'zgaruvchili funksiya grafiklarini chizishda **plot** buyrug'i va uning turli xil ko'rinishlaridan foydalaniladi.

Bunda **plot(x,y)** buyrug'iga uchinchi parametрни kiritish, ya'ni **plot(x,y,s)** buyrug'ini bajarish grafikdagi chiziqlar rangi, tipi, stili va foydalanuvchi talabiga ko'ra boshqa xususiyatlarini ko'rsatish imkonini beradi. Bu xususiyatlarni o'zgartirish grafik oyna menyu buyruqlari, uskunalar paneli elementlari orqali va grafik maydonda sichqonchani o'ng tugmasini bosish orqali ham amalga oshirsa bo'ladi. Agar bitta koordinata sistemasida bir necha grafikni bir vaqtda (bitta oynada) chizish talab qilinsa, u holda **plot** buyrug'ida avval 1-oraliq va 1-funksiya, keyin 2-oraliq va 2-funksiya va h.k. kiritiladi.

Bitta grafik oynada bir vaqtda bir yoki bir necha grafik maydon qurish yoki bitta grafik maydonda bir yoki bir nechta funksiyalar grafiklarini bir vaqtda yasash mumkin. Bir nechta funksiyalar bo'laklaridan tuzilgan funksiyalarni chizish mumkin:

$$y(x) = \begin{cases} \sin x & -4\pi \leq x \leq -\pi \\ 3(x/\pi + 1)^2 & -\pi < x \leq 0 \\ 3e^{-x} & 0 < x \leq 5 \end{cases} \text{ funksiya grafigini chizamiz:}$$

```
>> 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)
```



5.14-rasm. Funksiya grafigi.

5.10. MATLAB dasturlashtirish tizimida uch o'lchamli grafiklarni qurish va formatlash

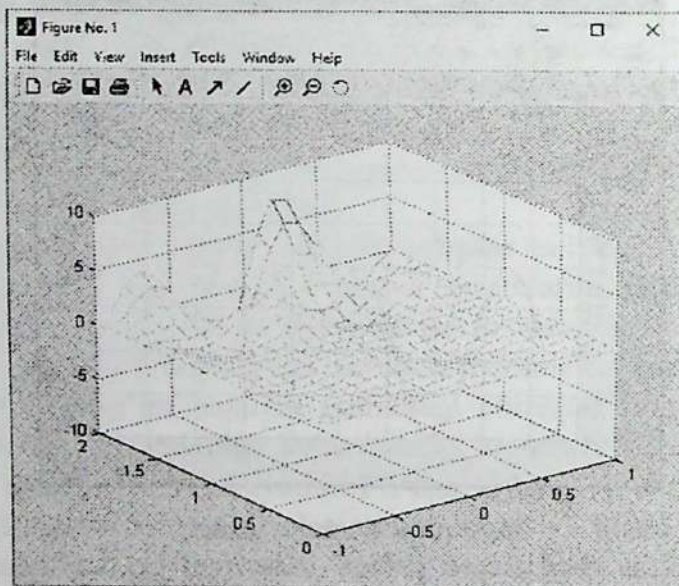
MATLABda ba'zi funksiyalar argumentlari ikki va undan ortiq bo'lgan hollarda ham uning grafiklarini qurish mumkin. $Z=f(x,y)$ ikkita o'zgaruvchili funksiya ko'rinishida tasvirlangan murakkab funksiyalarning grafiklarini qurish ancha oson. Bunday grafiklar uch o'lchovli yoki 3D-grafika deb yuritiladi.

MATLABda uch o'lchovli grafiklarni chizishda asosan quyidagi buyruqlardan foydalaniladi:

- **bar3**, **plot3**, **mesh**, **surf**, **sphere**, **cylinder**, **bar3h**, **contour**, **meshgrid**, **fill3**, **ellipsoid**, **logo** va h.k.;

Odatda ko'p o'zgaruvchili funksiya grafiklarini chizishda **mesh** buyrug'i va uning turli xil ko'rinishlaridan foydalaniladi. Masalan:

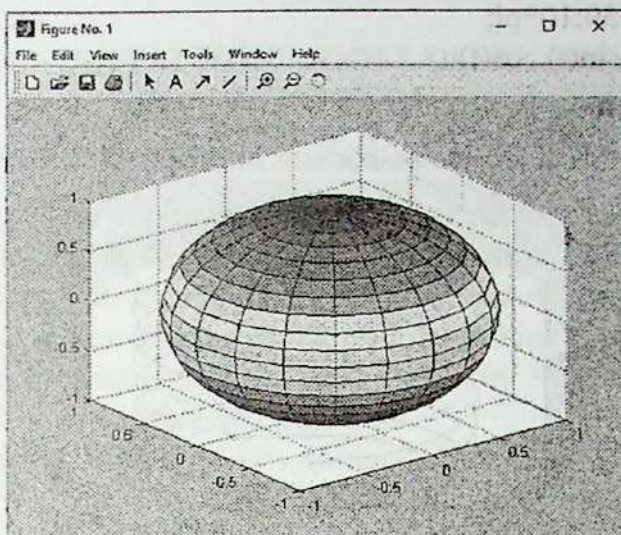
```
>> [X,Y]=meshgrid(-1:0.1:1, 0:0.1:2);  
>> Z=4*sin(2*pi*X).*cos(1.5*pi*Y).*(1-X.^2).*Y.*(1-Y);  
>> mesh(X,Y,Z)
```



5.15-rasm. Funksiya grafigi.

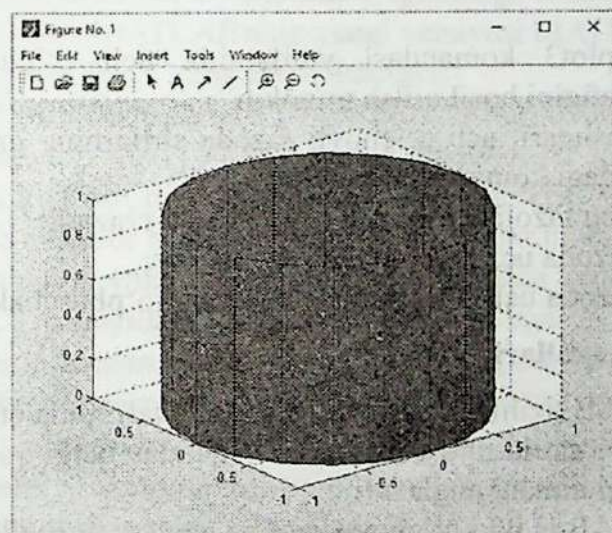
Fazoda geometrik jismlarni tasvirlashda **sphere**, **cylinder**, **bar3h**, **ellipsoid** kabi bir qator buyruqlardan foydalaniladi. Masalan:

>> sphere



5.16-rasm. Fazoda uch o'lchovli grafik tasvir.

>> cylinder

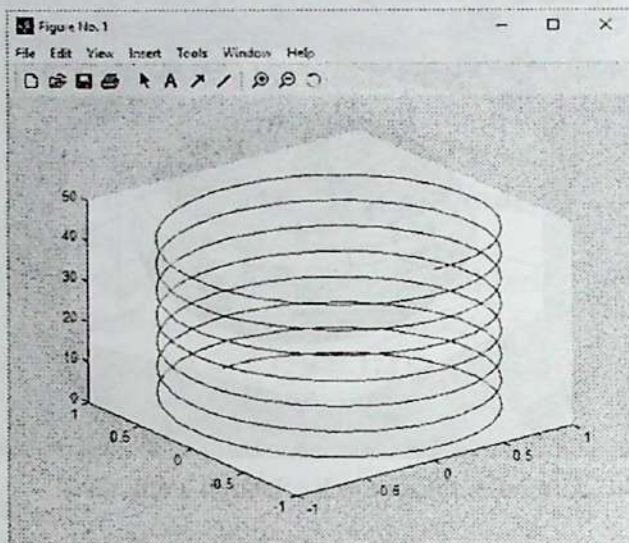


5.17-rasm. Fazoda geometrik jismlarni tasvirlash.

Uch o'lovli fazoda grafik chizish uchun `plot3 (x,y,z)` komandasi-dan foydalaniladi. Bunda x,y,z -vektorlar bir xil sondagi koordinatalar-ga ega bo'lishi kerak, aks holda sistema xatolikni beradi. Masalan:

```
>>t=0:pi/50:10*pi;
```

```
>> plot3(sin(t),cos(t),t)
```



5.18-rasm. Uch o'lovli grafika.

Demak, `plot3` komandasi yordamida uch o'lovli fazoda chiziqlarning grafigini hosil qilish mumkin.

Bundan tashqari, uch o'lovli fazoda sirtlarning grafigini hosil qiluvchi quyidagi komandalar mavjud:

- **mesh** – bu fazoda uch o'lovli «to'r»ni chizadi;
- **surf** – fazoda uch o'lovli sirtini chizadi;
- **fill3** – fazoda uch o'lovli to'ldirilgan ko'pburchakni chizadi.

Nazorat savollari

1. MATLAB tizimida natija qaysi so'z orqali ifoda etiladi?
2. Ustun va satrlarni o'chirish uchun qanday belgi qo'yiladi?
3. `>>` belgi nimani anglatadi?
4. MATLABda ikki o'lovli grafika qanday quriladi?
5. Fazoda geometrik jismlar qanday tasvirlanadi?

6-BOB. IMITATSION MODELLASHTIRISH VA ULARNING TURLARI

6.1. Simulink tizimida imitatsion modellar yaratish

Simulink tizimi – bu murakkab texnik tizimlarga dinamik sistemalarni modellashtirish, imitatsiya va tahlil qilish uchun interaktiv vositasi hisoblanadi. U grafik blok-diagrammalarni qurish, dinamik tizimlarni imitatsiya qilish, tizimlarning ishlashini tekshirish va loyihalarni mukammallashtirish imkoniyatlarini beradi. Simulink MATLAB bilan to'la integrallashgan.

Uning asosiy interfeysi diagrammalarni qurish uchun grafikli uskunalar vositasi va moslashtirilgan blokli kutubxonalar to'plamidan iboratdir.

Simulink tizim darajasida modellashtirish va simulatsiya qilish uchun mo'ljallangan bo'lib, bu bizga bitta dizayn muhitida ishlab chiqilayotgan tizimni har tomonlama o'rganishga imkon beradi.

Hozirgi vaqtda MATLABning yangi versiyasi MATLAB 6.5 (Release 13) va Simulink 5 keng ishlatilmoqda.

MATLAB 6.5 dasturlarni tez bajarishni ta'minlovchi kompilatorga ega. Shu sababli MATLAB 6.5 texnik hisoblashlar sohasida S dasturlash tilida kodlash bilan raqobatlashishi mumkin. Yangi versiya-ning diqqatga sazovor tomonlaridan biri m-fayllar mahsuldorligining (effektivligining) avvalgi versiyalardagiga nisbatan yuqoriligidir.

Simulink 5 ta quyidagi yangi xususiyatlarga ega:

1) o'rni belgilangan (fiksatsiya qilingan) nuqta bilan hisoblashlarni amalga oshirish mumkin. Suzuvchi nuqta bilan hisoblashlardan fiksatsiya qilingan nuqta bilan hisoblashlarga yoki teskarisiga o'tish yo'li bilan modelni mukammallashtirish mumkin (bu holda Fixed-Point Blocksetni o'rnatish zarur);

2) Look-Up Table Editor uskunasi jadval bloklaridagi ma'lumotlarni qulay holda ko'rib chiqish va tahrirlash imkoniyatini beradi. Tahrirlagichni chaqirish model oynasidagi Tools menyusidan amalga oshiriladi;

3) model Discretizer asbobi uzluksiz bloklarni diskret bloklarga tanlab almashtirish imkoniyatini beradi (Control System Toolbox, 5.2-versiyani o'rnatish talab qilinadi). Diskretizator model oynasidagi Tools menyusidan chaqiriladi;

4) mukammallashtirilgan Diagnostic Viewer xatoliklarni diagnostika qilish vositasi xatolar to'g'risidagi axborotlarni konfiguratsiya qilish va ularga giperhavolalarni qo'shish (kiritish) imkoniyatini beradi;

5) maskalar tahrirlagichi Mask Editor dinamik dialog oynasini yaratish vositasiga ega. Maskalar tahrirlagichining Parameters bo'limidagi Callback paneli blok (osttizim) parametrlarining o'zgarishini qayta ishlaydigan funksiyalarni kiritish imkoniyatini beradi. S-function Builder bloki yangi Data Properties bo'limiga ega. Uning yordamida portlardagi ma'lumotlarning turlarini, kirish va chiqish signallarining ko'rinishini (haqiqiy yoki kompleks) berish, portlarning metka (belgi)larini aniqlash, signallarning birliklarini kiritish mumkin.

Yangi Model Verification Library kutubxonasi qo'shilgan. Kutubxona hisoblash jarayonida modelni tekshiruvchi bloklarga ega.

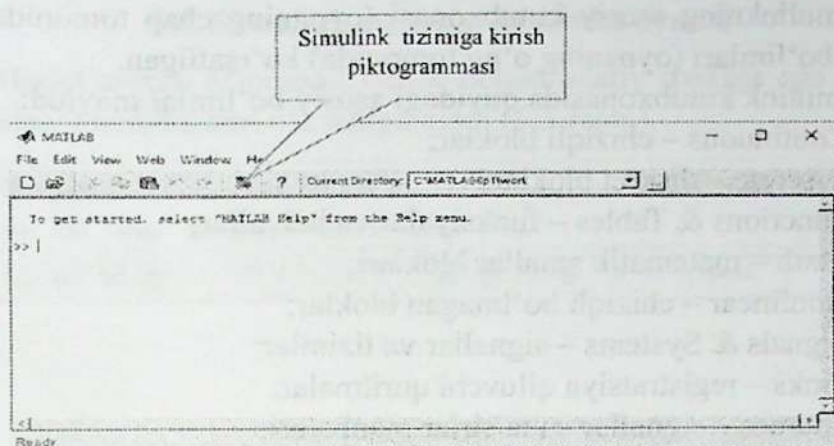
Hisobot yaratish uskunasi Print details model va osttizimlarning sxemalari hamda bloklarning parametrlari va ularning qiymatlarini o'z ichiga olgan HTML-hujjatlarni shakllantiradi. Print details buyrug'i **File** menyusiga kiritilgan.

Simulink paketini ishga tushirish uchun quyidagi ketma-ketlik amalga oshiriladi:

MATLAB dasturining asosiy oynasi ochilgandan keyingi holat 5.1-rasmda ko'rsatilgan, simulink dasturi quyidagi uchta usulning biri yordamida ishga tushiriladi.

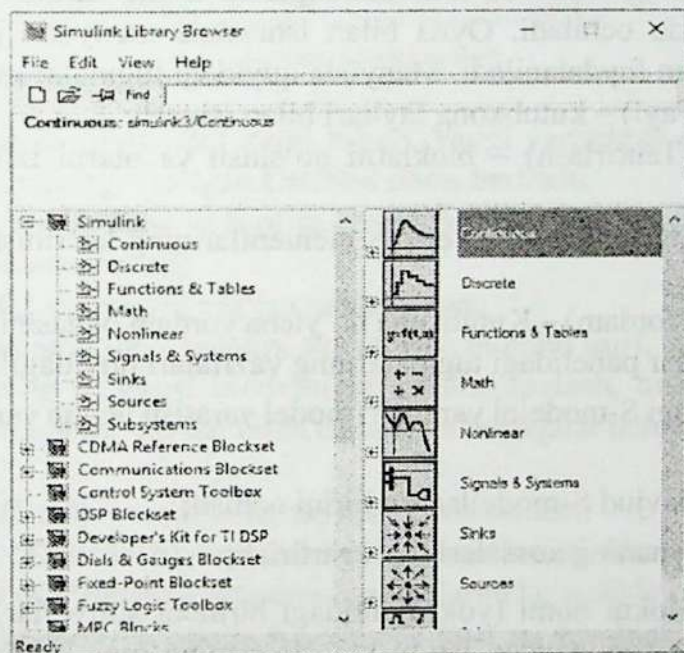
- Simulink piktogrammasini bosish;
- MATLABning bosh oynasidagi buyruq satrida Simulink so'zini terib klaviaturadagi <Enter> klavishini bosish;

- file menyusida Open... buyrug'ini bajarish va modelning faylini (mdl-fayl) ochish.



6.1-rasm. MATLAB dasturining asosiy oynasi.

6.2. MATLAB tizimidagi Simulink kutubxonasining bloklari



6.2-rasm. Simulink kutubxonasining bo'limlari oynasi.

Birinchi va ikkinchi usullardan foydalanilganda Simulink kutubxonasi bo'limlarining Library Browser oynasi ochiladi.

Simulinkning asosiy kutubxonasi (oynaning chap tomonida) va uning bo'limlari (oynaning o'ng tomonida) ko'rsatilgan.

Simulink kutubxonasida quyidagi asosiy bo'limlar mavjud:

- Continuous – chiziqli bloklar;
- Discrete – diskret bloklar;
- Functions & Tables – funksiyalar va jadvallar;
- Math – matematik amallar bloklari;
- Nonlinear – chiziqli bo'lmagan bloklar;
- Signals & Systems – signallar va tizimlar;
- Sinks – registratsiya qiluvchi qurilmalar;
- Sources – signallar va ta'sirlar manbalari;
- Subsystems – osttizimlar bloklari.

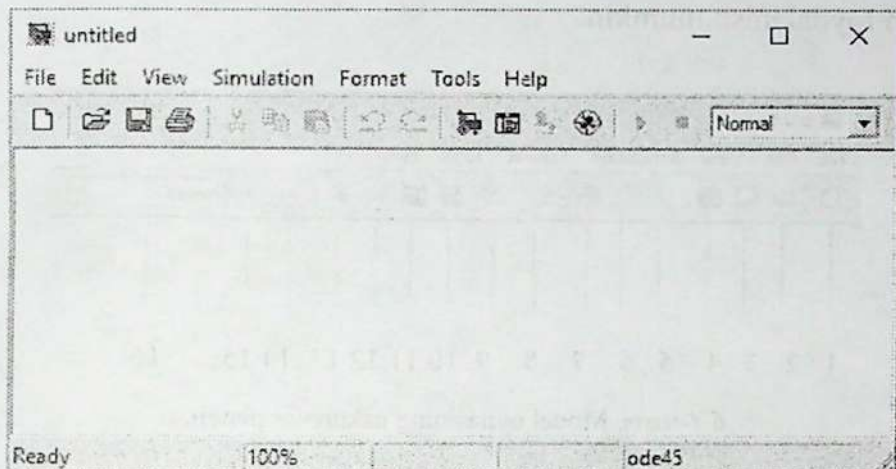
Simulink kutubxonasi bo'limlarining ro'yxati daraxtsimon shaklga ega bo'lib, bunday ro'yxatlar bilan ishlash qoidalari odatdagidek. Kutubxonaning zarur bo'limi tanlanganda uning tarkibi oynaning o'ng qismida ochiladi. Oyna bilan ishlashda menyuda jamlangan buyruqlardan foydalaniladi. Menyuda quyidagi tugmalar mavjud:

- *File* (Fayl) – kutubxona fayllari bilan ishlash;
- *Edit* (Tahrirlash) – bloklarni qo'shish va ularni izlash (nomi bo'yicha);
- *View* (Ko'rinish) – interfeys elementlarining ko'rinishini boshqarish;
- *Help* (Yordam) – Kutubxona bo'yicha yordam oynasini chiqarish. Uskunalar panelidagi tugmalarning vazifalari quyidagilar:
-  Yangi S-modelni yaratish (model yaratish uchun yangi oynani ochish);
-  Mavjud S-modellardan birini ochish;
-  Oynaning xossalarini o'zgartirish;
-  Blokni nomi (yoki nomidagi birinchi simvollar) bo'yicha izlash. Blok topilgandan keyin kutubxonaning mos bo'limi ochiladi va topilgan bo'lim ajratib ko'rsatiladi. Agar blok topilmasa, izoh

oynasida *Not found* < blok nomi>(blok topilmadi) yozuvlari paydo bo'ladi.

6.3. Simulink paketining model oynasi

Model oynasi Microsoft Office uchun odatiy shaklga ega bo'lib, quyidagi elementlarni o'z ichiga oladi (6.3-rasm).



6.3-rasm. Model oynasi.

- Sarlavha (oynaning nomi bilan). Yangi yaratilgan oynaga mos tartib raqamga ega bo'lgan Untitled nomi beriladi;

- **File, Edit, View** va boshqa buyruqlarga ega bo'lgan menyu;
- Uskunalar paneli;
- **Model** sxemasini yig'ish uchun oyna;
- Modelning joriy holatini aks ettiruvchi holat satri.

Oynaning menyusi modelni tahrirlash, sozlash, hisoblash jarayonini boshqarish, fayllar bilan ishlash va boshqalar uchun buyruqlarga ega:

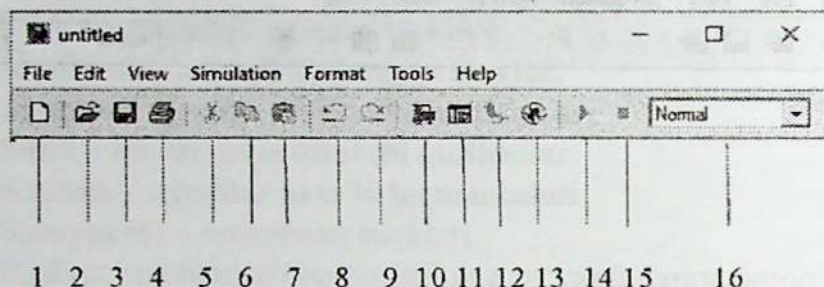
- **File** (Fayl) – modelning fayllari bilan ishlash;
- **Edit** (Tahrirlash) – modelni o'zgartirish va bloklarni izlash;
- **View** (Ko'rinish) – interfeys elementlarini ko'rsatishni boshqarish;
- **Simulation** (Modellash) – modellash va hisoblash jarayonini boshqarish sozlanmalari (parametrlari)ni berish.

• **Format** (Formatlash) – bloklar va modelning tashqi ko‘rinishini o‘zgartirish;

• **Tools** (Uskunalar vositalari) – model bilan ishlash uchun maxsus vositalarni qo‘llash (sozlagich, chiziqli tahlil va boshqalar);

• **Help** (Yordam) – Yordam tizimining oynasini chaqirish;

• **Model** bilan ishlash uchun uskunalar panelidagi tugmalardan ham foydalanish mumkin.



6.4-rasm. Model oynasining uskunalar paneli.

Uskunalar paneli tugmalarining vazifalari:

- 1) New Model – Modelning yangi (bo‘sh) oynasini;
- 2) Open Model – Mavjud mdl-faylni ochish;
- 3) Save Model – Diskda mdl-faylni saqlash;
- 4) Print Model – Modelning blok-diagrammalarini bosmaga chiqarish;
- 5) Cut – Modelning belgilangan qismini qirqib oraliq saqlash buferiga olish;
- 6) Copy – Modelning belgilangan qismining nusxasini oraliq saqlash buferiga olish;
- 7) Paste – Oraliq saqlash buferida saqlangan axborotni model oynasiga qo‘yish;
- 8) Undo – Oldingi tahrirlash amalini bekor qilish;
- 9) Redo – Bekor qilingan tahrirlash amalining natijasini tiklash;
- 10) Library Browser – Kutubxonalar oynasini ochish;
- 11) Toggle Model Browser – Model oynasini ochish;

12) Go to parent system – Ostitizimdan iyerarxiya bo'yicha yuqori pog'onadagi tizimga o'tish. Buyruq faqat ostitizim ochilgan bo'lsagina ishlaydi;

13) Debug – Model sozlagichini ishga tushirish;

14) Start/Pause/Continue Simulation – modelni bajarilish uchun ishga tushirish (**Start**); model ishga tushgandan keyin tugmaning tas-

virida  simvol hosil bo'ladi va unga endi Pause (modellashni to'xtatish) buyrug'i mos keladi; modellashni davom ettirish uchun xuddi shu tugmaning o'zi qaytadan bosiladi, chunki bu tugmaga pauza rejimida Continue (Davom ettirish) buyrug'i mos keladi;

15) Stop Simulation – modellashni to'xtatish;

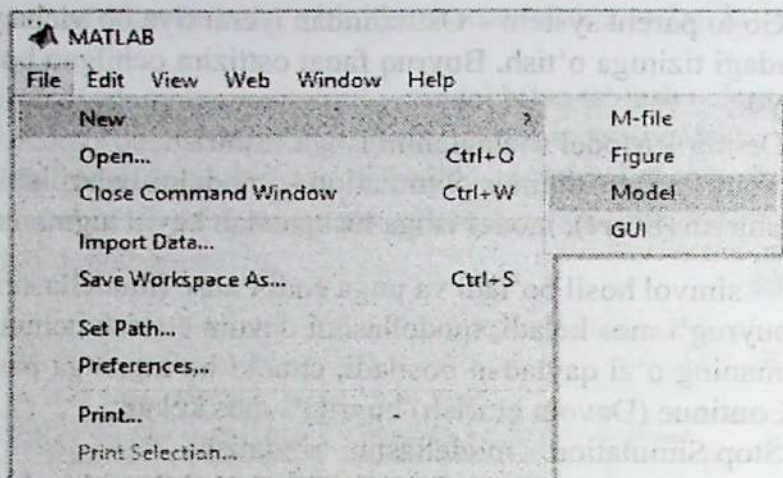
16) Normal/Accelerator – Odatdagi/Tezlashtirilgan hisoblash rejimi. Ushbu rejimdan Simulink **Performance Tool** ilovasi o'rnatilgan bo'lsagina foydalanish mumkin.

Model oynasining pastki qismida holat satri joylashgan. Unda, sichqonchaning tugmasi interfeys mos elementining ustiga olib kelinganda, asboblari paneli tugmalari va menyu punktlariga qisqa sharhlar hosil bo'ladi. Xuddi shu matn maydoni Simulink holatini ko'rsatish uchun ham xizmat qiladi: **Ready** (Tayyor) yoki **Running** (Bajarilish).

6.4. MATLAB tizimining Simulink paketida model yaratish

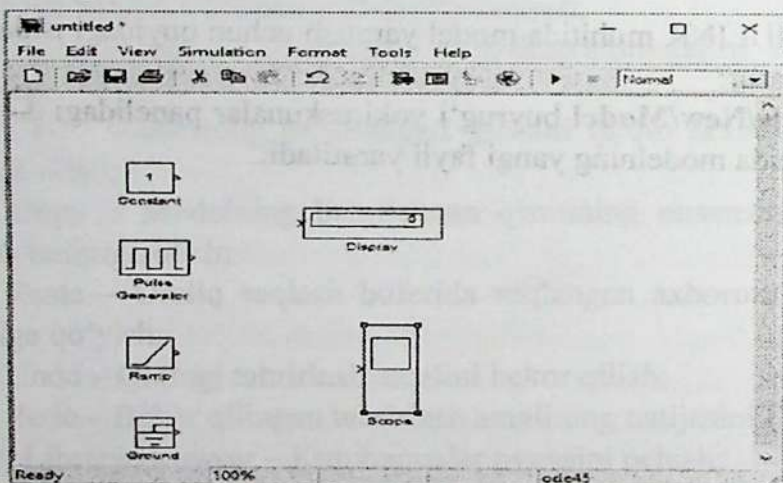
SIMULINK muhitida model yaratish uchun quyidagi ishlarni bajarish zarur:

- **File/New/Model** buyrug'i yoki uskunalar panelidagi  tugma yordamida modelning yangi fayli yaratiladi;



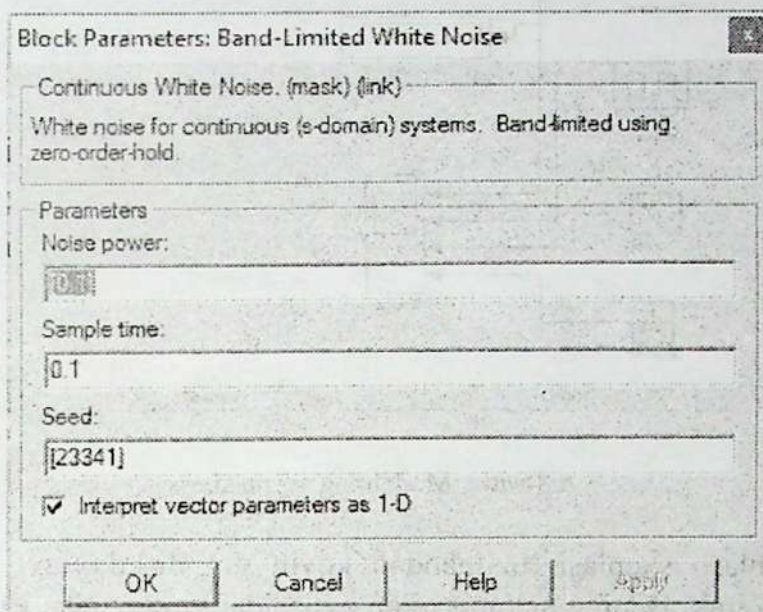
6.5-rasm. Yangi fayl yaratish.

• Model oynasida bloklar joylashtiriladi. Buning uchun kutubxonaning kerakli bo‘limi ochiladi (masalan, **Sources** – manbalar). So‘ngra kerakli blokni kursor bilan ko‘rsatiladi va sichqonchani chap tugmasini bosib yaratilgan oynaga suriladi. Bloklarga ega bo‘lgan model oynasi 6.6-ramda ko‘rsatilgan. Agar blokni o‘chirish zarur bo‘lsa, uning ustida sichqonchani chap tugmasi bosiladi, keyin esa klaviaturadagi **Delete** klavishi bosiladi.



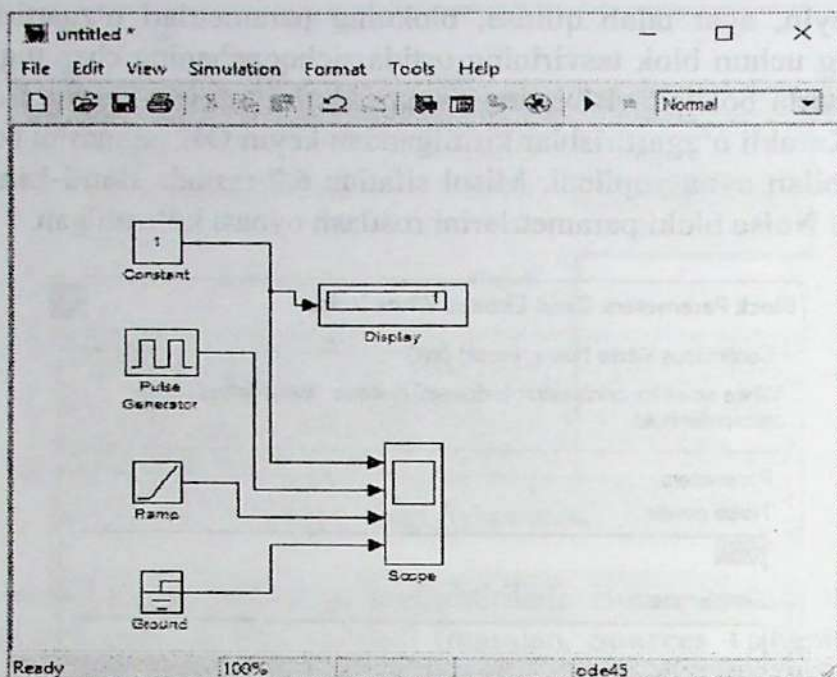
6.6-rasm. Bloklarga ega bo‘lgan model oynasi.

Keyin, agar talab qilinsa, blokning parametrlari o'zgartiriladi. Buning uchun blok tasvirining ustida sichqonchani chap tugmasi ikki marta bosiladi. Blokning parametrlarini tahrirlash oynasi ochiladi. Kerakli o'zgartirishlar kiritilgandan keyin **OK** tugmasini bosish yo'li bilan oyna yopiladi. Misol sifatida 6.7-rasmda **Band-Limited White Noise** bloki parametrlarini rostdash oynasi ko'rsatilgan.



6.7-rasm. Band-Limited White Noise bloki parametrlarini rostdash oynasi.

Barcha kerakli bloklar sxemaga joylashtirilgandan keyin sxema elementlari o'zaro ulanadi. Bloklarni o'zaro bir-biriga ulash uchun blokning chiqishiga kursor olib boriladi va sichqonchani chap tugmasi bosilgan holda boshqa blokning kirishigacha liniya chiziladi. Bog'lanish liniyasida tarqalish nuqtasini hosil qilish uchun tugun joylashishi zarur bo'lgan nuqtada sichqonchani o'ng tugmasi bosilib kerakli liniya chiziladi. Chizilgan liniyani yo'qotish uchun liniya tanlanadi va klaviaturadagi **Delete** klavishi bosiladi. Bloklari bir-biri bilan ulangan modelning sxemasi 6.8-rasmda keltirilgan.



6.8-rasm. Modelning sxemasi.

Hisoblash sxemasi tuzilgandan keyin uni diskda fayl sifatida saqlash kerak. Buning uchun sxema oynasidagi menyudan **File/Save** punkti tanlanib, papka va fayl nomi ko'rsatiladi.

Bloklar bilan amallar. Bir oynadagi bloklardan ikkinchi oynaga qo'yish uchun nusxa olish quyidagicha amalga oshiriladi: kerakli kutubxona yoki model-prototipning oynasi ochiladi va kerakli blok sichqoncha yordamida yaratilayotgan (tahrir qilinayotgan) modelning oynasiga suriladi.

Bloklardan menyu buyruqlari yordamida ham nusxa olish mumkin. Bunda bajariladigan amallar ketma-ketliga quyidagicha bo'ladi:

- model yoki kutubxona oynasida nusxasi olinishi kerak bo'lgan blok yoki bloklar belgilanadi;
- faol oynaning **Edit** (To'g'rilash) menyusida **Copy** (Nusxa olish) buyrug'i tanlanadi;

- blokning nusxasi qo'yiladigan oyna faollashtiriladi va undagi **Edit** menyusidan **Paste** buyrug'i tanlanadi.

Har bir blokning nusxasiga Simulink nomi beradi. Blokning birinchi nusxasining nomi uning kutubxonadagi nomi bilan bir xil bo'ladi. Blokning keyingi nusxalarining nomiga tartib raqami qo'shiladi. Foydalanuvchi blokning nomini o'zgartirishi mumkin. Blok nusxalari sozlanuvchi parametrlarining qiymatlari original (nusxasi olingan) blokni bilan bir xil bo'ladi.

Model bloklarining o'rinlarini almashtirish. Model ichidagi bloklarning o'rnini sichqoncha yordamida ularni surish yo'li bilan almashtiriladi. Bunda Simulink bloklarni o'zaro bog'lovchi liniyalarni qaytadan chizadi. Bir necha blokni birgalikda surish uchun ular ajratiladi va ajratilgan bloklardan biri yangi o'ringa suriladi. Natijada qolgan ajratilgan bloklar ham ular orasidagi nisbiy masofalar va bog'lovchi liniyalar o'zgarmagan holda suriladi.

Model ichida bloklardan nusxa olish quyidagi ikkita usuldan biri yordamida amalga oshirilishi mumkin:

- **<Ctrl>** tumasini bosgan holda blokni kerakli joyga surish;
- sichqonchaning o'ng tugmasini bosgan holda kerakli joyga surish, bunda blokka navbatdagi tartib raqami beriladi.

Blokni olib tashlash. Blok sxemadagi keraksiz bloklarni olib tashlash uchun ularni ajratib **** yoki **<Backspace>** klavishlaridan birini bosish yetarli. Bundan tashqari, blok-sxema oynasining **Edit** menyusidagi **Clear** (Tozalash) yoki **Cut** (Qirqish) buyruqlaridan ham foydalanish mumkin. Agar **Cut** buyrug'idan foydalanilgan bo'lsa, keyinchalik olib tashlangan blokning nusxasini **Paste** buyrug'i yordamida modelga joylashtirish mumkin.

Blokni uzib qo'yish. Blokni bog'lovchi liniyalardan uzib qo'yish uchun **<Shift>** klavishi bosilgan holda uni boshqa joyga suriladi.

Blokni burish. Boshlang'ich holatda blok orqali signal chapdan o'ngga o'tadi, ya'ni chap tomonda blokning kirishlari, o'ng tomonda esa chiqishlari joylashadi. Blokni burish uchun quyidagi amallarni bajarish kerak:

- burish kerak bo'lgan blok ajratiladi;

- blok sxema oynasining **Format** (Format) menyusidagi quyidagi buyruqlardan biri tanlanadi: **Flip Block** (Blokni 180 gradusga burish) yoki **Rotate Block** (Blokni soat mili yo'nalishida 90 gradusga burish).

Blokning o'lchamlarini o'zgartirish. Blok ajratiladi va sichqonchaning ko'rsatkichi blok burchak belgilaridan birining ustiga olib kelinadi. Ko'rsatkichning shakli ikki tomonga yo'nalgan strelka ko'rinishiga o'zgargan vaqtda sichqonchaning chap tugmasi bosilib kerakli tomonga suriladi.

Blokning nomini o'zgartirish va surish. Blokning nomi yagona va kamida bitta simvoldan iborat bo'lishi kerak. Blokning nomini o'zgartirish uchun uning ustida sichqonchaning chap tugmasi bosiladi (bosib qo'yib yuboriladi) va odatdagi usullar yordamida kerakli o'zgartirishlar kiritiladi.

Shriftni o'zgartirish uchun model oynasidagi **Format** (Format) menyusidan **Font** (Shrift) buyrug'i chaqiriladi va ochilgan dialog oynasidan shrift tanlanadi. Agar blokdan o'tadigan signalning yo'nalishi chapdan o'ngga bo'lsa, blokning nomi uning pastida, signalning yo'nalishi o'ngdan chapga bo'lsa, yuqorisida va pastdan yuqoriga yoki yuqoridan pastga bo'lsa, blokning o'ng tomonida bo'ladi.

Ajratilgan blok nomining o'rnini ikki xil usul bilan o'zgartirish mumkin:

- sichqoncha yordamida blokning qarama-qarshi tomoniga surish;
- model oynasining **Format** menyusidagi **Flip Name** buyrug'idan foydalanish – bu usul ham blok nomini qarama-qarshi tomonga o'tkazish imkonini beradi.

Blok nomini berkitish uchun model oynasining **Format** menyusidagi **Hide Name** (Nomni berkitish) buyrug'idan foydalaniladi. Blokning berkitilgan nomini tiklash uchun **Show Name** (Nomni ko'rsatish) buyrug'i xizmat qiladi.

Signallarning belgilari va kommentariylarni (izohlarni) joylashtirish. Blok sxemalar tushunarli va qulay bo'lishi uchun liniyalardan o'tuvchi signallarni ko'rsatuvchi belgilar qo'yish mumkin.

Belgilar gorizontal liniyalarning ostiga yoki ustiga, vertikal liniyalarning o'ng yoki chap tomoniga joylashtiriladi. Belgini liniyaning boshlanishi, oxiri yoki o'rtasiga qo'yish mumkin.

Signal belgisini hosil qilish uchun liniyaning ustida sichqonchaning chap tugmasi ikki marta bosiladi va belgining matni kiritiladi. Sichqonchaning chap tugmasi liniyaning ustida bosilishiga e'tibor berish kerak. Aks holda model uchun izoh hosil bo'ladi.

Belgi sichqoncha yordamida siljtiladi. Agar belgini siljitish vaqtida <Ctrl> klavishi bosib turilsa, yangi joyda belgining nusxasi hosil bo'ladi. Belgining nusxasini liniyaning boshqa sigmentida sichqonchaning chap tugmasini ikki marta bosish yo'li bilan ham hosil qilish mumkin.

Belgini tahrir qilish uchun uning ustida sichqonchaning chap tugmasi bosiladi va matnga kerakli o'zgartirishlar kiritiladi.

Belgini olib tashlash uchun u ajratiladi va <Shift>klavishi bosib turilgan holda yoki <Backspace> klavishi bosiladi. Bu holda liniyadagi hamma belgilar olib tashlanadi.

Izohlarni hosil qilish va o'zgartirish. Izohni blok sxemadagi har qanday bo'sh yerga joylashtirish mumkin. Buning uchun sichqonchaning chap tugmasi ikki marta bosiladi va hosil bo'lgan to'rtburchak ramkaning ichiga izohning matni kiritiladi.

Izoh sichqoncha yordamida siljtiladi. Agar izoh siljtilayotgan vaqtda <Ctrl> klavishi bosib turilsa, yangi joyda izohning nusxasi hosil bo'ladi.

Hosil qilingan izohni tahrir qilish mumkin. Buning uchun uning ustida sichqonchaning chap tugmasi bosiladi va kerakli o'zgartirishlar kiritiladi. Shriftni o'zgartirish uchun izohning matni ajratiladi va blok sxema oynasidagi **Format** (Format) menyusidan **Font** (Shrift) buyrug'i tanlanadi. Kerakli shrift, uning o'lchami va atributlari tanlangandan **OK** tugmasi bosiladi.

Izohni olib tashlash uchun <Shift> klavishi bosilgan holda yoki <Backspace> klavishi bosiladi.

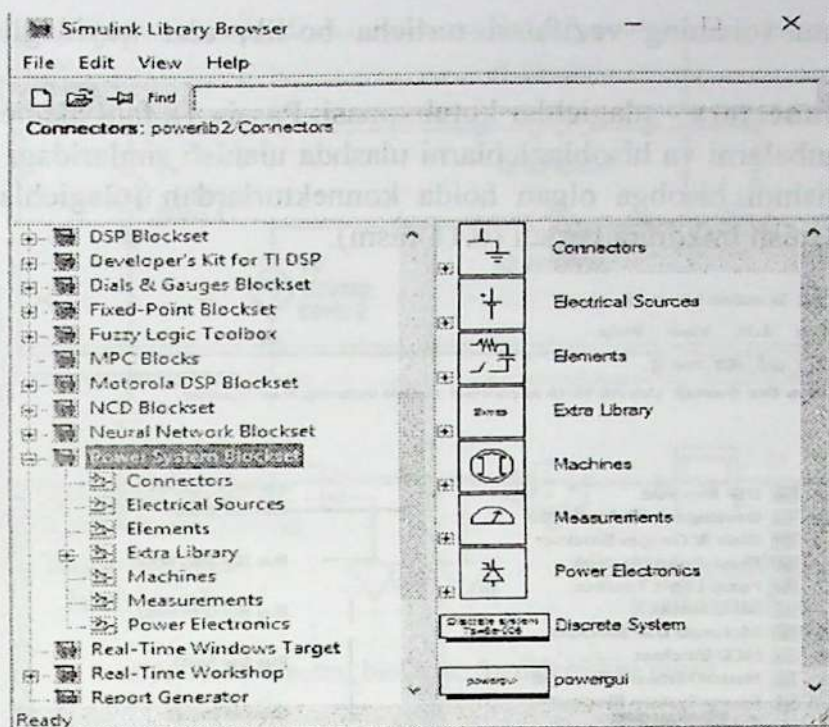
6.5. Power System Blockset paketida model yaratish

Power System Blockset paketi elektr qurilmalar va elektron qurilmalarning virtual modellarini yaratish imkonini beruvchi tizim hisoblanadi. Power System Blocksetning afzalligi shundaki, simulatsiya va strukturaviy modellashtirish usullarini birlashtirib, murakkab elektr tizimlarini modellashtirish mumkin. Masalan, elektr energiyasi-ning yarimo'tkazgichli konvertorining quvvat qismini Power System Blockset simulatsiya bloklari va boshqaruv tizimi – oddiy Simulink bloklari yordamida bajarish mumkin, bu uning elektr zanjirini emas, balki uning ishlash algoritmini aks ettiradi.

Ushbu yondashuv, elektron simulatsiya paketlaridan farqli o'laroq, butun modelni sezilarli darajada soddalashtirishga imkon beradi va bu uning barqarorligi va ishlash tezligini oshiradi. Bundan tashqari, Power System Blockset paketidan foydalanish jarayonida modelda bloklardan va boshqa Simulink kutubxonalaridan, shuningdek MATLABning funksiyalaridan foydalanish mumkin, bu elektr tizimlarini modellashtirish uchun deyarli cheksiz imkoniyatlarni beradi.

Shunday qilib, Power System Blockset Simulinkning bir qismi sifatida hozirgi vaqtda elektr qurilmalar va tizimlarni modellashtirish uchun eng yaxshi paketlardan biri hisoblanadi.

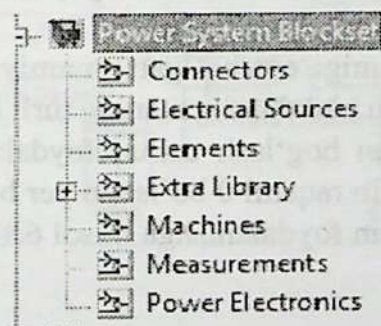
Power System Blockset paketi tarkibida kuchlanish elementlari (katta quvvatli elementlar) bo'lgan sistemalarni imitatsion modellash uchun mo'ljallangan. U yetti bo'limdan iborat (6.9-rasm).



6.9-rasm. Power System Blockset paketi oynasining umumiy ko'rinishi.

6.6. Power System Blockset paketi kutubxonasining bloklari

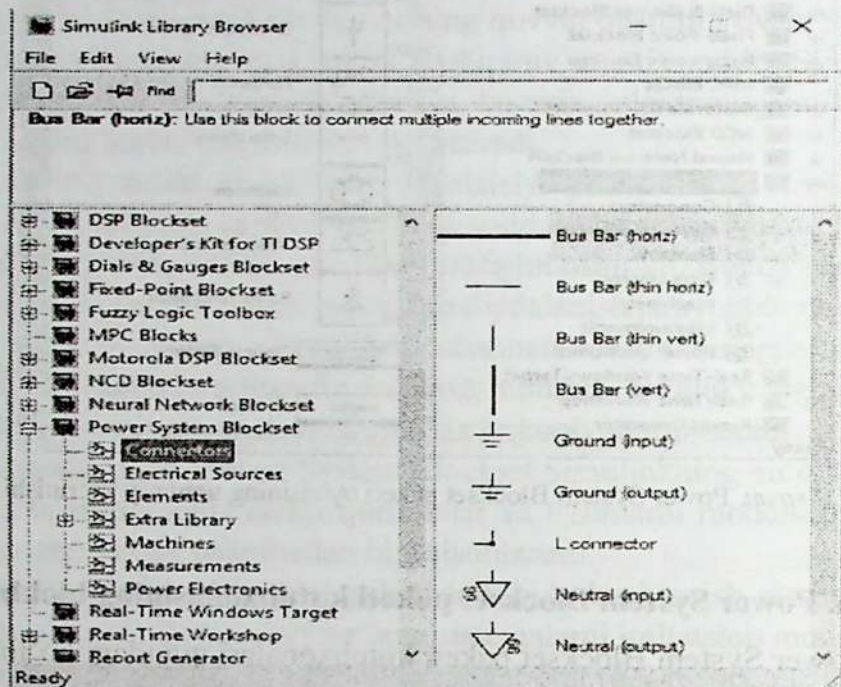
Power System Blockset paketi kutubxonalari quyidagi bo'limlardan tashkil topgan:



6.10-rasm. Power System Blockset paketining 7 ta kutubxonasi nomlari.

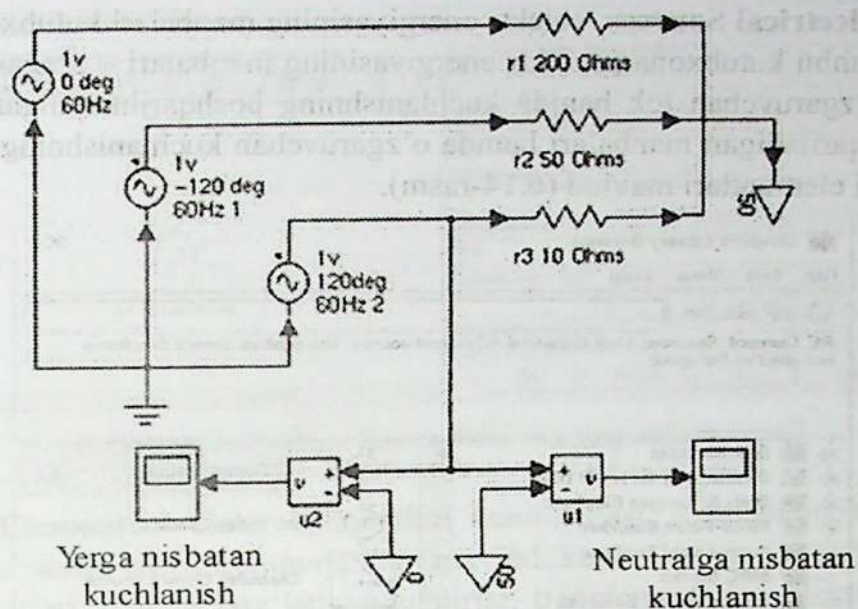
Kutubxonaning vazifalari turlicha bo'lib, ular quyidagilardan iborat:

Connectors – ulagichlar kutubxonasi. Passiv va faol elementlarni, manbalarni va hisoblagichlarni ulashda ulanish simlaridagi oqim yo'nalishini hisobga olgan holda konnektorlardan (ulagichlardan) foydalanish imkonini beradi (6.11-rasm).



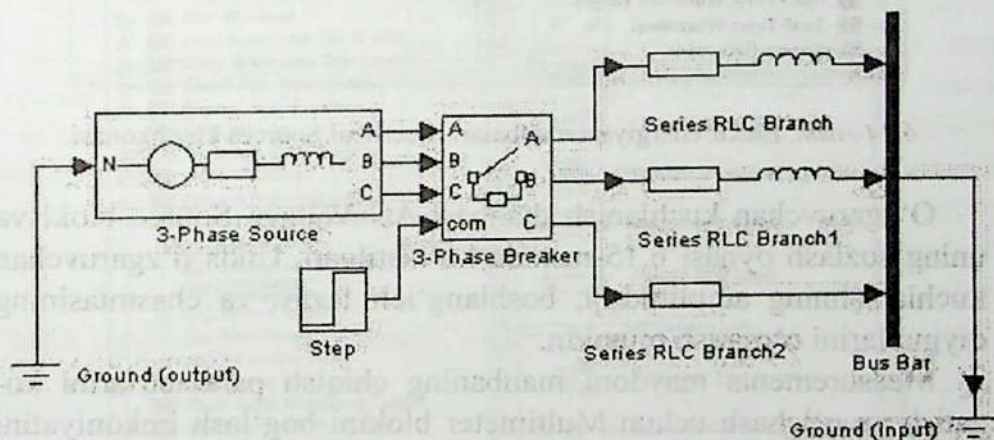
6.11-rasm. Connectors kutubxonasining oynasi.

Neutral bloki tartib raqamiga ega bo'lgan umumiy nuqta hosil qilish uchun ishlatiladi. Ushbu blokdan sxemaning turli joylaridagi ikki nuqtani liniyani chizmasdan bog'lash uchun foydalanish mumkin. Agar Neutral blokining tartib raqami 0 bo'lsa, u yer bilan bog'lanish hosil qiladi. Neutral blokidan foydalanishga misol 6.12-rasmda ko'rsatilgan.



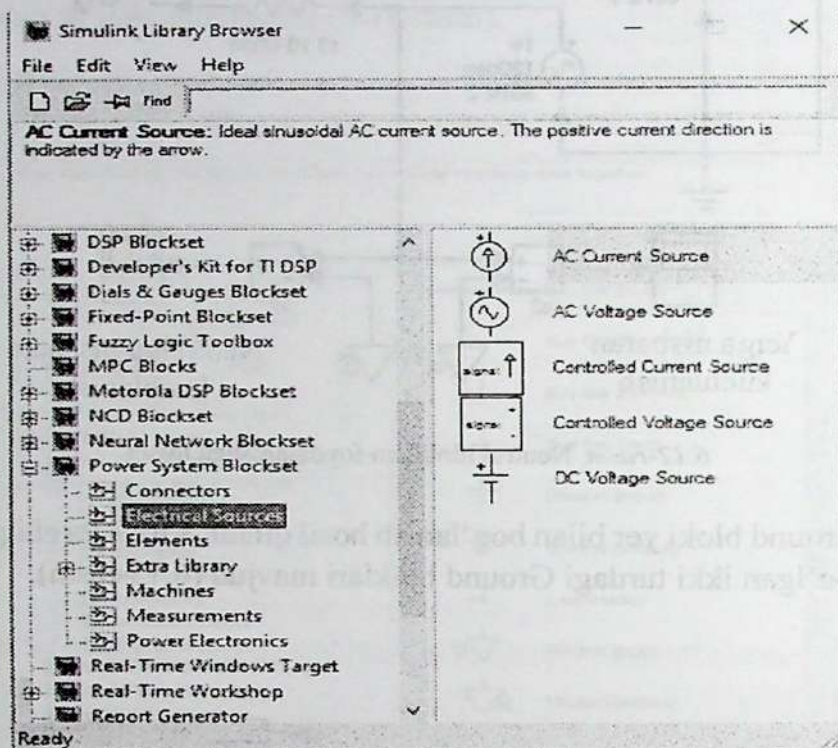
6.12-rasm. Neutral blokidan foydalanishga misol.

Ground bloki yer bilan bog'lanish hosil qiladi. Kirish va chiqishga ega bo'lgan ikki turdagi Ground bloklari mavjud (6.13-rasm).



6.13-rasm. Ground blokidan foydalanishga misol.

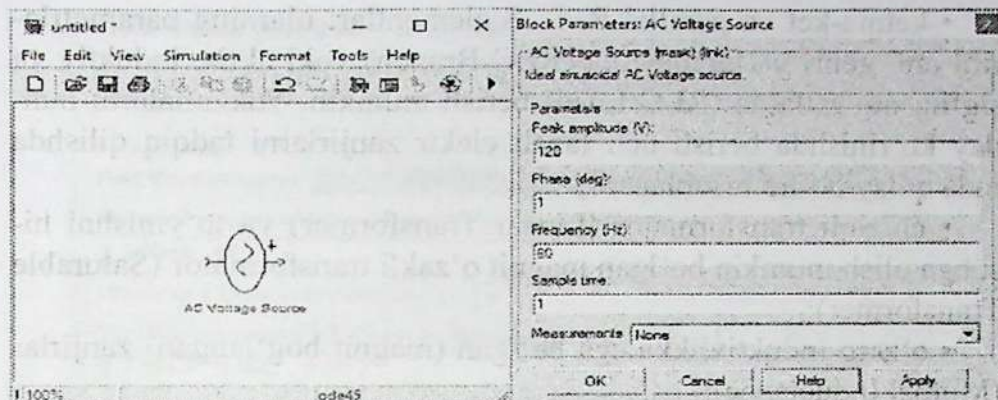
Electrical Sources – elektr energiyasining manbalari kutubxonasi. Ushbu kutubxonada elektr energiyasining manbalari – o‘zgarmas va o‘zgaruvchan tok hamda kuchlanishning boshqarilmaydigan va boshqariladigan manbalari hamda o‘zgaruvchan kuchlanishning uch fazali elementlari mavjud (6.14-rasm).



6.14-rasm. Elektr energiyasi manbalari Electrical Sources kutubxonasi.

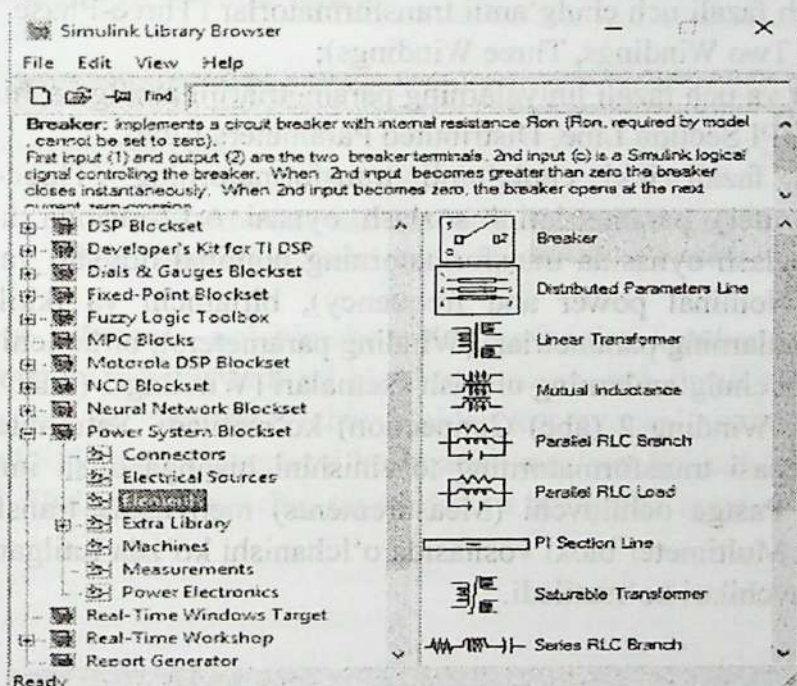
O‘zgaruvchan kuchlanish manbasi AC Voltage Source bloki va uning sozlash oynasi 6.15-rasmda ko‘rsatilgan. Unda o‘zgaruvchan kuchlanishning amplitudasi, boshlang‘ich fazasi va chastotasining qiymatlarini o‘rnatish mumkin.

Measurements maydoni manbaning chiqish parametrlarini kuzatish va o‘lchash uchun Multimeter blokini bog‘lash imkoniyatini beradi.



6.15-rasm. AC Voltage Source bloki va uning sozlash oynasi.

Elements – elektr elementlari kutubxonasi. Kutubxonada juda ko‘p passiv elektr komponentlari mavjud: ketma-ket va parallel RLC zanjirlari, magnit bog‘langan zanjirlar, transformatorlar, elektr uza-tish liniyalari, kommutatsiya moslamalari va boshqalar (6.16-rasm).



6.16-rasm. Passiv elementlar kutubxonasi Elements oynasi.

- ketma-ket va parallel R, L, S elementlar: ularning parametrlarini om, genri va faradlarda (RLC Branch) yoki aktiv, induktiv va sig'im quvvatlarda (RLC Load) berish mumkin. Yuklamalarni bunday ko'rinishda berish uch fazali elektr zanjirlarni tadqiq qilishda juda qulay bo'lib hisoblanadi;

- chiziqli transformator (Linear Transformer) va to'yinishni hisobga olish mumkin bo'lgan magnit o'zakli transformator (Saturable Transformer);

- o'zaro induktivlikka ega bo'lgan (magnit bog'langan) zanjirlar (Mutual Inductance);

- kirish va chiqish signallari orasida talab qilingan nochiziqli bog'lanishni shakllantirish imkoniyatini beruvchi nochiziqli element (Surge Arrester);

- kalit (Breaker), uning ochiq holatdagi parametrlari (qarshiligi, induktivligi) va kirish signali nolga teng bo'lgandagi holati (ochiq yoki yopiq) sozlash maydonlarida beriladi;

- uch fazali uch chulg'amli transformatorlar (Three-Phase Transformer, Two Windings, Three Windings);

- bir va uch fazali liniyalarning parametrlarini amalga oshiruvchi bloklar (PI Section Line, Distributed Parameters Line).

Uch fazali ikki chulg'amli transformatorning (Three-Phase Transformer) parametrlarini sozlash oynasi 6.17-rasmda keltirilgan. Sotlash oynasida transformatorning nominal quvvati va chas-totasi (Nominal power and frequency), birlamchi va ikkilamchi chulg'amlarning parametrlari (Winding parameters), birlamchi va ikkilamchi chulg'amlarning ulanish sxemalari (Winding 1 (ABC) Connection, Winding 2 (abc) Connection) ko'rsatiladi. Saturable Core bayroqchasi transformatorning to'yinishini hisobga olish imkonini beradi. Pastga ochiluvchi (Measurements) menyusida transformatorning Multimeter bloki vositasida o'lchanishi ko'zda tutilgan holat o'zgaruvchilari ko'rsatiladi.

Three-Phase Transformer (Three Windings) (mask) \$link\$

This block implements a three-phase transformer by using three single-phase transformers. Set the winding connection to 'Yn' when you want to access the neutral point of the Wye (for winding 1 and 3 only).

Parameters

Port Configuration

Nominal power and frequency [Pn(VA) , fn(Hz)]

[250e6 , 60]

Winding 1 (ABC) connection:

Winding parameters [V1 Ph-Ph(Vrms) , R1(pu) , L1(pu)]

[735e3 , 0.002 , 0.08]

Winding 2 (abc-2) connection:

Winding parameters [V2 Ph-Ph(Vrms) , R2(pu) , L2(pu)]

[315e3 , 0.002 , 0.08]

Winding 3 (abc-3) connection:

Winding parameters [V3 Ph-Ph(Vrms) , R3(pu) , L3(pu)]

[315e3 , 0.002 , 0.08]

☐ Saturable core

Magnetization resistance Rm(pu):

500

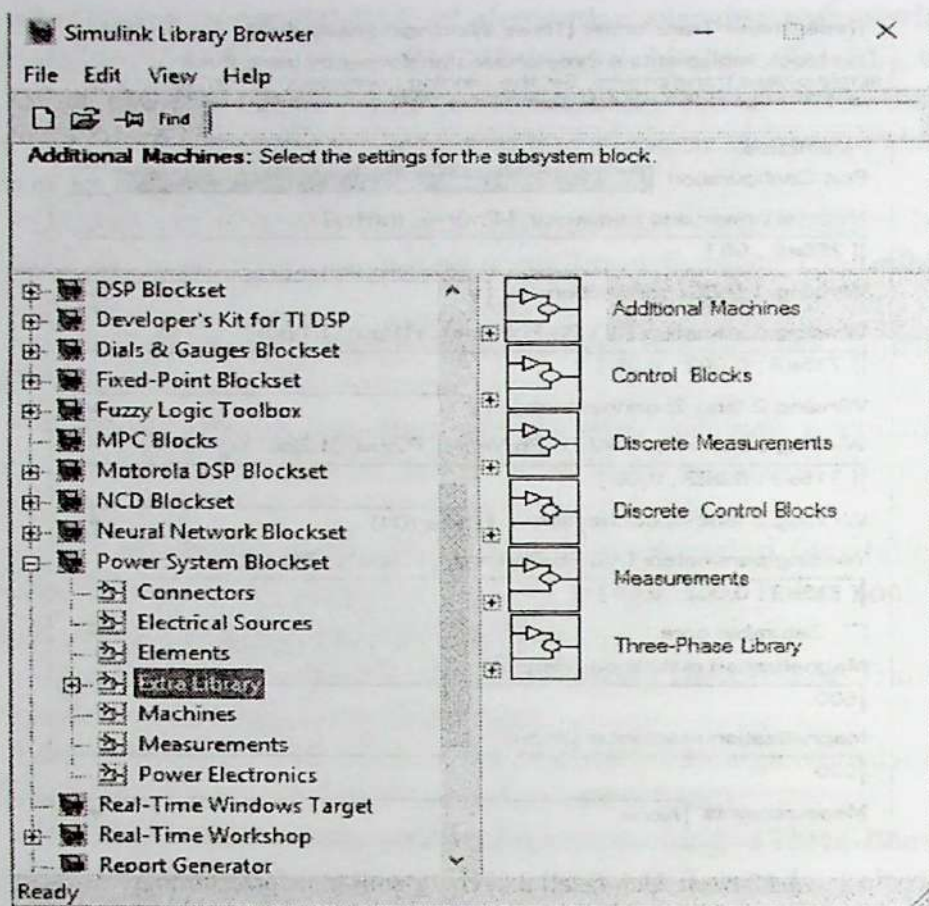
Magnetization reactance Lm(pu)

500

Measurements

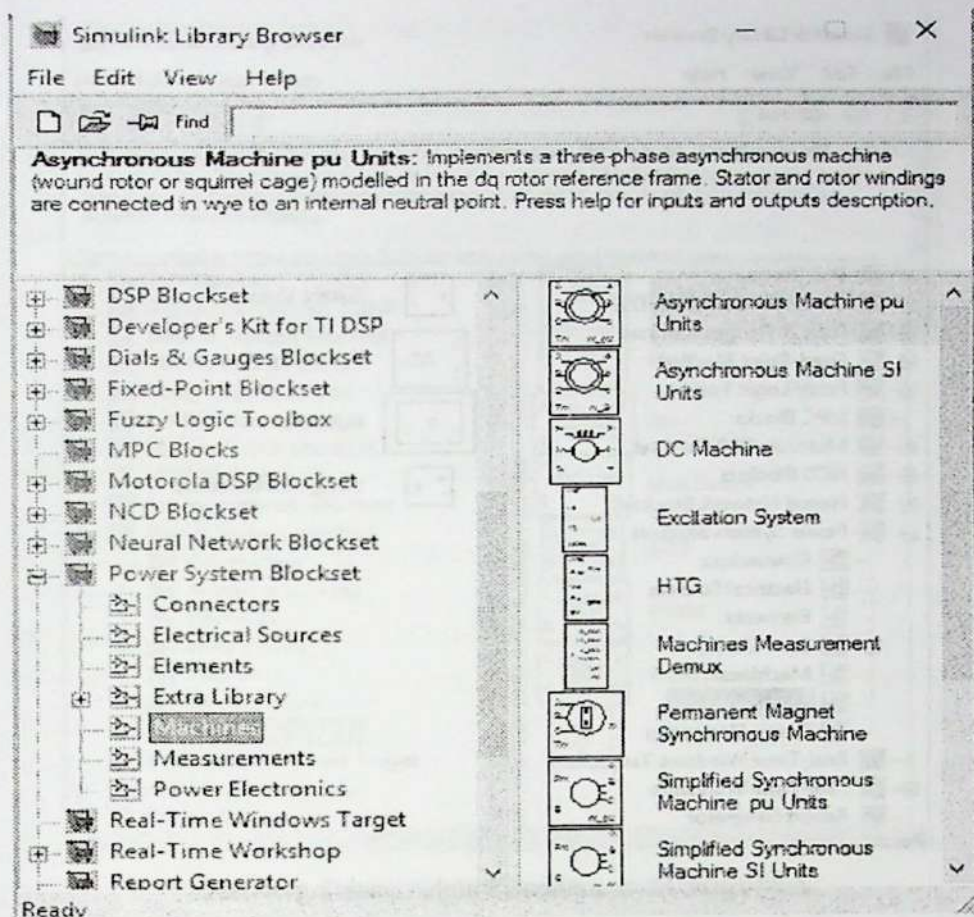
6.17-rasm. Uch fazali ikki chulg'amli transformatorning
(Three-Phase Transformer) parametrlarini sozlash oynasi.

Extra Library – qo'shimcha elektr qurilmalari kutubxonasi. Bu yerda juda ko'p sonli turli xil bloklar, xususan, uch fazali modellar kutubxonasi, qo'shimcha o'lchov bloklari kutubxonasi, yarimo'tkazgichli rektifikator va inverterni boshqarish moslamalari kutubxonasi, turli xil filtrlar, generatorlar, taymerlar va boshqalar mavjud (6.18-rasm).



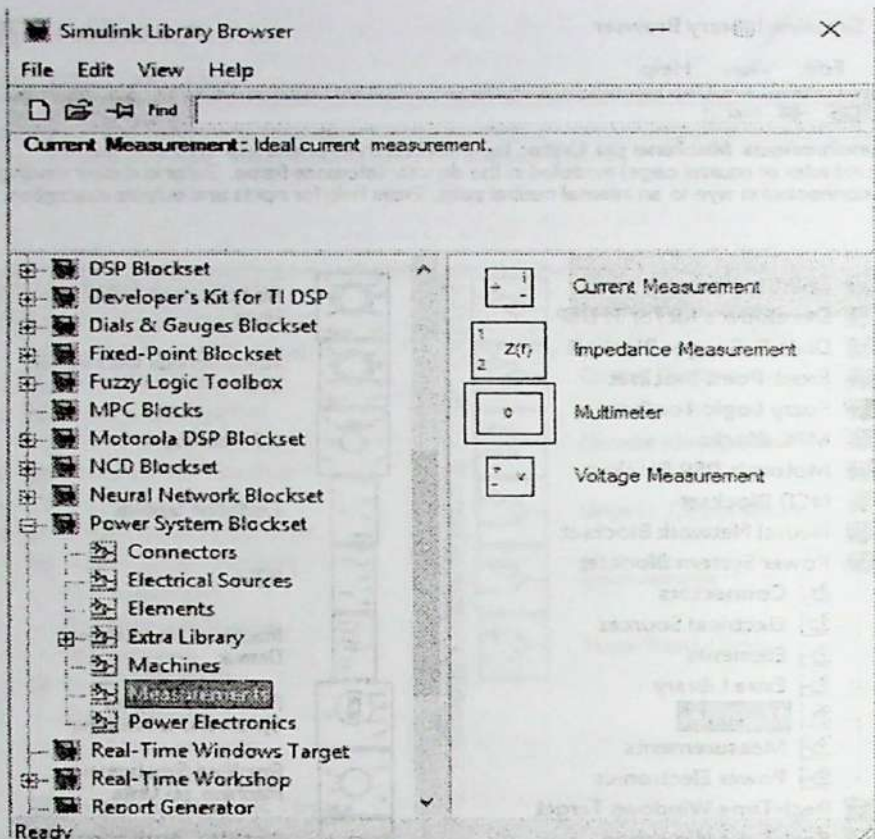
6.18-rasm. Extra Library kutubxonasi.

Machines – elektr mashinalar kutubxonasi. Elektr mashinalari kutubxonasida sinxron va asinxron mashinalarning turli xil modeliari, doimiy tok mashinalari, generatorlar uchun qo‘zg‘alish moslamalari (boshqaruv tizimiga ega gidravlik va bug‘ turbinalari), shuningdek o‘lchovli o‘zgaruvchilarni (oqimlar, oqim bog‘lanishlari, tezlik va h.k.) olish uchun mo‘ljallangan Machines Measurement Demux maxsus bloki mavjud. Bu kutubxona avtomat, apparatlar, stanoklar uchun sxemalar tuzish imkonini beradi (6.19-rasm).



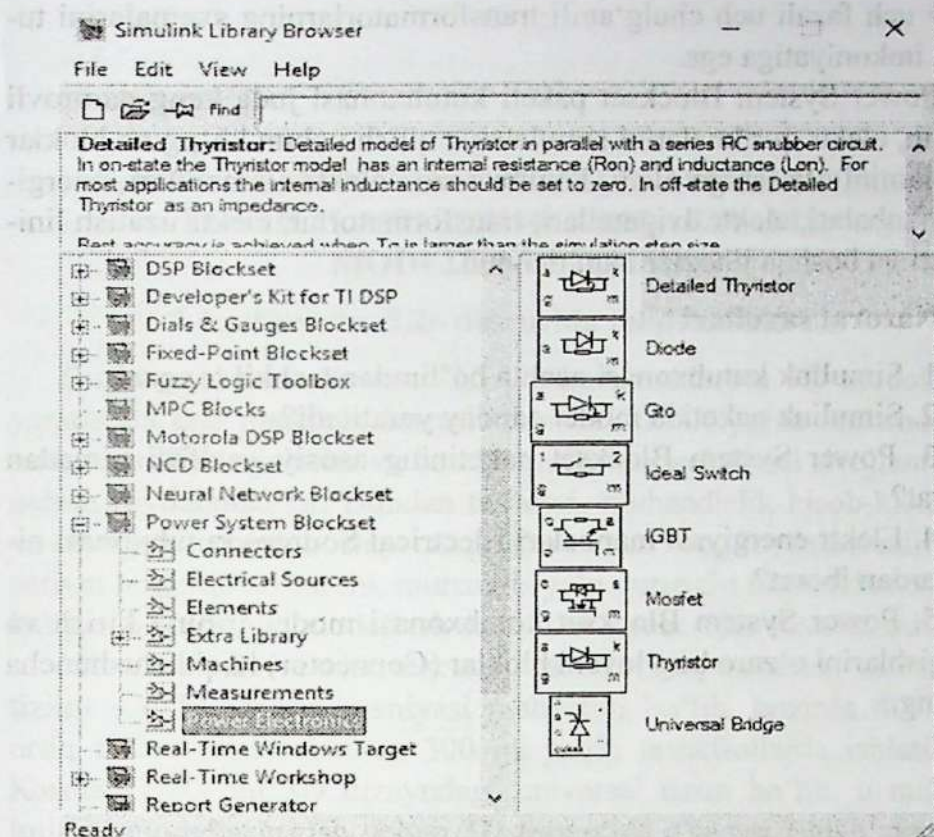
6.19-rasm. Machines kutubxonasi oynasi.

Measurements — o'lchash moslamalari kutubxonasi. O'lchov uskunalari kutubxonasi oqim va kuchlanish qiymatlarini o'lchash uchun bloklar, uch fazali tizimdagi oqim va kuchlanishni o'lchash uchun blok, elektr zanjiri bo'limi uchun o'lchagich va o'lchov moslamalarini to'g'ridan to'g'ri ulamasdan bloklarning o'zgaruvchilarini o'lchashga imkon beradigan Multimeter bloki mavjud (6.20-rasm).



6.20-rasm. Measurements kutubxonasi oynasi.

Power electronics – elektron qurilmalar uchun sxema yarata-di. Ushbu kutubxonada yarimo'tkazgichli qurilmalar (diod, tiristor, qulflanuvchi tiristor, IGBT tranzistorlar) modellari, shuningdek bir, ikki va uch fazali rektifikator yoki inverter davrlarini simulatsiya qilishga imkon beradigan universal ko'prik mavjud. Bundan tashqari, kutubxonada uch darajali ko'prik modeli va ideal kalit modeli mavjud. Shuningdek, kutubxona qo'shimcha elektr qurilmalar bo'limi, ular orasida juda ko'p sonli turli xil bloklarni o'z ichiga oladi. Uch fazali modellar kutubxonasi, qo'shimcha o'lchov bloklari kutubxonasi, boshqarish moslamalari kutubxonasi va boshqa elementlar ham mavjud (6.21-rasm).



6.21-rasm. Power electronics kutubxonasining oynasi.

Power System Blockset paketi quyidagilarni o'z ichiga oladi:

- uch fazali elektr zanjirlarni tadqiq qilishda bundan foydalanish juda qulay bo'lib hisoblanadi;
- kirish va chiqish signallari orasida talab qilingan noxiziqli bog'lanishni shakllantirish elementlari mavjud;
- kalit, uning ochiq holatdagi parametrlari (qarshiligi, induktivligi) va kirish signali nolga teng bo'lgandagi holatini (ochiq yoki yopiq) sozlash;
- bir va uch fazali liniyalarning parametrlarini amalga oshiruvchi bloklari mavjud;

- uch fazali uch chulg'amli transformatorlarning sxemalarini tuzish imkoniyatiga ega.

Power System Blockset paketi kutubxonasi juda keng qamrovli bo'lib, elektr qurilmalarini simulatsiya qilish uchun ko'pgina bloklar to'plamini o'z ichiga oladi. Kutubxonada elektr elementlari, energiya manbalari, elektr dvigatellari, transformatorlar, elektr uzatish liniyalari va boshqa jihozlar ham mavjud.

Nazorat savollari

1. Simulink kutubxonasi nechta bo'limdan tashkil topgan?
2. Simulink paketida model qanday yaratiladi?
3. Power System Blockset paketining asosiy vazifasi nimadan iborat?
4. Elektr energiyasi manbalari Electrical Sources kutubxonasi nimalardan iborat?
5. Power System Blockset kutubxonasi modellarining kirish va chiqishlarini o'zaro bog'lovchi bloklar (Connector) haqida tushuncha bering.

7-BOB. ZAMONAVIY AVTOMATLASHTIRILGAN LOYIHALASH TIZIMLARIDA GRAFIK MODELLASHTIRISH

7.1. «Компас-3Д» dasturida modellashtirish

Bugungi kunda «Компас-3Д», 2D chizmalar va 3D modellarni yaratishda eng mashhur dasturlardan biridir. Ko'pgina muhandislar undan qurilish rejalarini va barcha qurilish obyektlarini rivojlantirish uchun foydalanadilar. Bundan tashqari, muhandislik hisob-kitoblari va shunga o'xshash boshqa maqsadlar uchun keng foydalaniladi. Aksariyat hollarda dasturchi, muhandis yoki quruvchi tomonidan o'qitiladigan birinchi 3D modellashtirish dasturi «Компас-3Д» hisoblanadi.

«Компас-3Д» (комплексная автоматизированная система) tizimi «АСКОН» kompaniyasi mahsuloti bo'lib, hozirda 400 dan ortiq ta'lim muassalari va 300 ga yaqin tashkilotlarda ishlatiladi. Компас-3Д – bu 3D dizayndagi universal tizim bo'lib, u mahsulotlarni modellashtirish uchun 3D formatida g'oyani ishlab chiqishdan tortib, hujjatlarning to'liq to'plamini tayyorlashga qadar turli xil vositalarni o'z ichiga oladi. Unda mahsulot, bino, tuzilish modelini yaratish va shu bilan birga qabul qilingan standartlarga mos keladigan qismlarning batafsil chizmalarini, texnik hujjatlarni olish mumkin.

«Компас-3Д» – bu to'g'ridan to'g'ri modellashtirishning yuqori funktsionalligi va foydalanish qulayligining muvaffaqiyatli kombinatsiyasi tufayli minglab korxonalar uchun standartga aylangan 3D dizayn tizimi hisoblanadi.

«Компас-3Д»ning asosiy tarkibiy qismlari 3D dizayn tizimi, Compass Graphic avtomatlashtirilgan 2D dizayn tizimidan iborat. «Компас-3Д» tizimi kelajakdagi mahsulotning to'laqonli elektron uch o'lchovli modelini yaratishga va quyidagi natijalarga erishishga imkon beradi:

- fazoviy loyihalashni amalga oshirish;
 - dizaynning dastlabki bosqichlarida asosiy xatolarni qilmaslik;
 - mavjud bo'lmagan mahsulotni tasavvur qilish va uning to'plamini tekshirish;
 - kerakli hisob-kitoblarni va dizaynlarni optimallashtirishni amalga oshirish;
 - loyihani iloji boricha tezroq modellashtirish va o'zgartirish;
 - CAM-tizimlar bilan birgalikda «Компас-3Д»dan foydalangan holda mahsulotni ishlab chiqarishga tayyorlash vaqtini qisqartirish;
 - mahsulot hujjatlarini tezda tayyorlash;
 - 3D modelini ishlatgan holda marketing materiallarini tayyorlash.
- «Компас-3Д» quyidagilarni o'z ichiga oladi:
- jamoaviy ishlash vositalari;
 - 3D qattiq, sirt va to'g'ridan to'g'ri modellashtirish uchun zamonaviy vositalar;
 - versiyalar va konfiguratsiyalar bilan ishlash vositalari, shu jumladan oynali qismlar va yig'ish qismlari;
 - o'n minglab ehtiyot qismlar va standart mahsulotlarni o'z ichiga olgan loyihalarda ishlash vositalari;
 - varaq materialidan qismlarni modellashtirish uchun vositalar, tekis chizmani avtomatik chizish;
 - quyma qoliplarini qurishni osonlashtiradigan maxsus xususiyatlar: quyma qiyaliklar, ajratish chiziqlari, qism shaklidagi bo'shliqlar;
 - standart elementlarning maxsus kutubxonalarini yaratish vositalari;
 - muvofiq ravishda loyihalash va texnologik hujjatlarni yaratish imkoniyati.
- GOST, ISO, DIN yoki korporativ standartlar bilan: chizmalar, oddiy va guruh texnik xususiyatlari, hisobotlar, diagrammalar, jadvallar, matnli hujjatlar;
- uch o'lchovli modellarda o'lchamlar va belgilashlarni o'rnatish qobiliyati;
 - turli xil CAD / CAM / CAE tizimlariga ma'lumotlarni uzatish uchun vositalar.

Dastur turli formatdagi fayllarni import-eksport qilishni qo‘l-
lab-quvvatlaydi.

«Компас-3Д» tizimi boshqa avtomatlashtirilgan loyihalash tizim-
laridan o‘zining parametri holda loyihalashi va modellarni Solid
yadrosida bajarishi tufayli keng qo‘llanilmoqda. Solid yadrosida ba-
jarilgan modellar hozirgi mashinasozlikning yuqori ALT hisoblangan
Pro/ENGINEER, CATIA, Unigraphics, Solid Works tizimlari bilan
to‘g‘ridan to‘g‘ri aloqani ta‘minlaydi. Bu ALTlarning hozirda bir-bi-
ri bilan o‘zaro CAD/CAE/CAM/PDM (CAD-kompyuter yordamida
konstruksiyalash, CAE-kompyuter yordamida muhandislik tahlilla-
rini bajarish, CAM-texnologik jarayonlarni kompyuterda bajarish,
PDM- loyiha ma‘lumotlarini boshqarish tizimi) tizimidagi integ-
ratsiyasi, modulligi e‘tiborga loyiqdir. Bu esa ularning ta‘sir doirasini
kengaytirish, ALT bozorida yuqori o‘rinni egallashi imkonini beradi.
Shu bilan birgalikda ularning ta‘lim muassasalari bilan aloqasi kela-
jakdagi rivojini belgilaydi.

«Компас-3Д» tizimini yuklash uchun quyidagi ketma-ketlik ba-
jariladi:

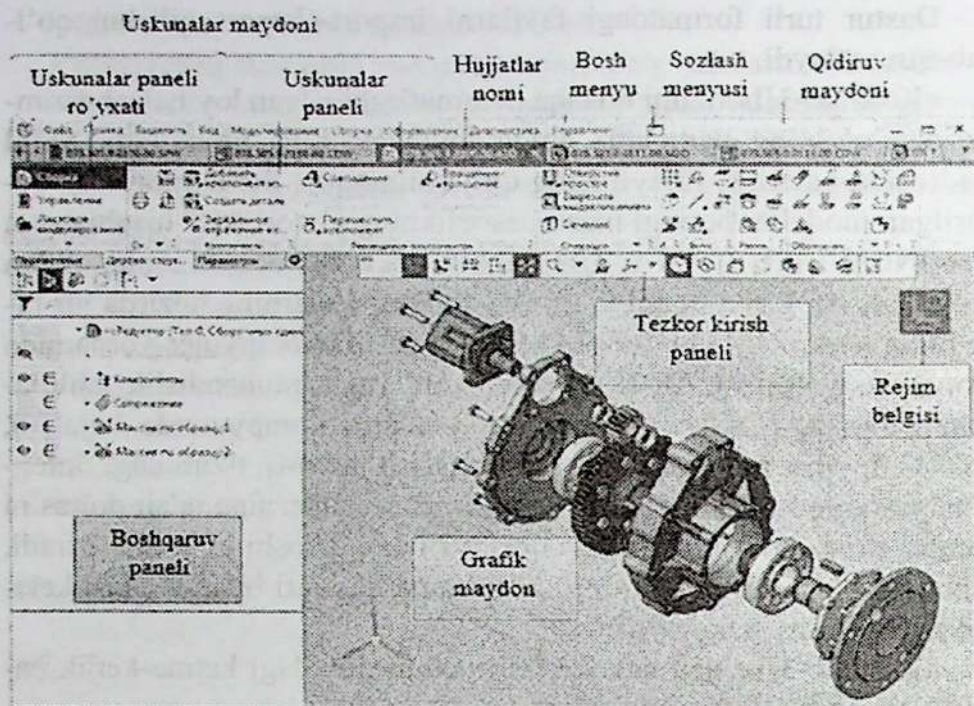
Push → КОМПАС-3Дv17

yoki



ishchi stolda joylashgan dastur yorlig‘i orqali.

Dastur yuklangandan keyin, uning interfeysi (foydalanuvchi bilan
muloqot oynasi) paydo bo‘ladi. Uning asosiy elementlari quyidagi-
lardan iborat:



7.1-rasm. «Компас-3Д»ning elementlari.

Uskunalar maydoni – hujjat obyektlarini tahrirlash, ko‘rish, chop etish va yaratish buyruqlarini tanlash uchun xizmat qiladi.

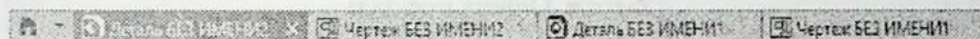
U quyidagilarni o‘z ichiga oladi:

- uskunalar paneli to‘plamlari ro‘yxati – uskunalar maydonining chap tomonida joylashgan bo‘lib, to‘plamlar o‘rtasiga o‘tishga xizmat qiladi;
- joriy to‘plam panellari – uskunalar maydonining bir qismini egallaydi, qolgan qismi chaqiruv buyruqlari tugmachalarini o‘z ichiga oladi.



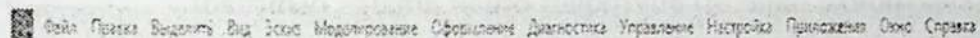
7.2-rasm. Uskunalar maydoni.

Hujjatlar nomi menyusi – joriy hujjatlar o‘rtasida oynalarni almashtirish uchun xizmat qiladi. Hujjatlar nomi menyusining kontekst menyusi hujjatlar va oynalarning boshqaruv buyruqlarini o‘z ichiga oladi.



7.3-rasm. Hujjatlar nomi menyusi.

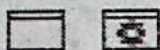
Bosh menyu – tizim buyruqlarini chaqirishga xizmat qiladi. Menyuning sarlavhalarini bo‘limlarini o‘z ichiga oladi. Bosh menyuning tarkibi joriy hujjat va tizimning ish tartibi turiga bog‘liq bo‘ladi.



7.4-rasm. Bosh menyu.

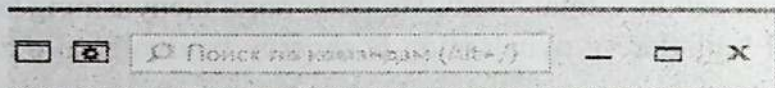
Sozlash menyusi tugmasi – quyidagi buyruqlarni bajarish uchun harakatlar menyusi ochiladi:

- ish maydoni – tanlash, saqlash, ish maydonining nomini o‘zgartirish yoki o‘chirish (ya’ni «Компас-3Д» interfeysi sozlamalari);
- fon – interfeys elementlarining ranglarini tanlash (tugmalar, maydonlar, ro‘yxatlar va boshqalar);
- Mavzu – «Компас-3Д»ning rangli gamma oynasiga mavjud mavzusini tanlash. Yorqin va qorong‘i rangli mavzular;
- belgilar – ikonkalarini ko‘rsatish variantini tanlash. Instrumental maydon; monoxrom va rangli tasvirlar;
- klaviatura – «Компас-3Д» buyruqlarining kombinatsiyasini sozlash tugmalari.



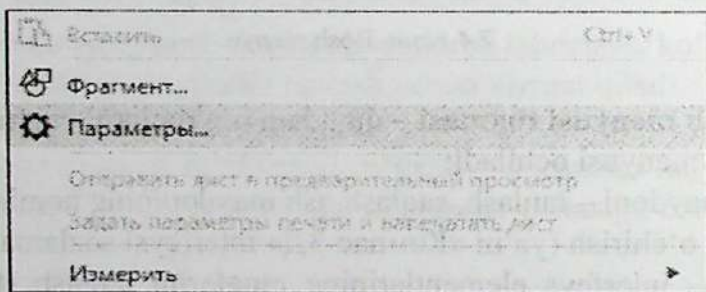
7.5-rasm. Sozlash menyusi tugmasi.

Qidiruv maydoni – nom bo‘yicha buyruqlarni qidiradi. Matn kiritilganda quyida bu maydon topilgan buyruqlar ro‘yxatini ko‘rsatadi. Ro‘yxatdan buyruqlarni tanlab uni ishga tushiradi.



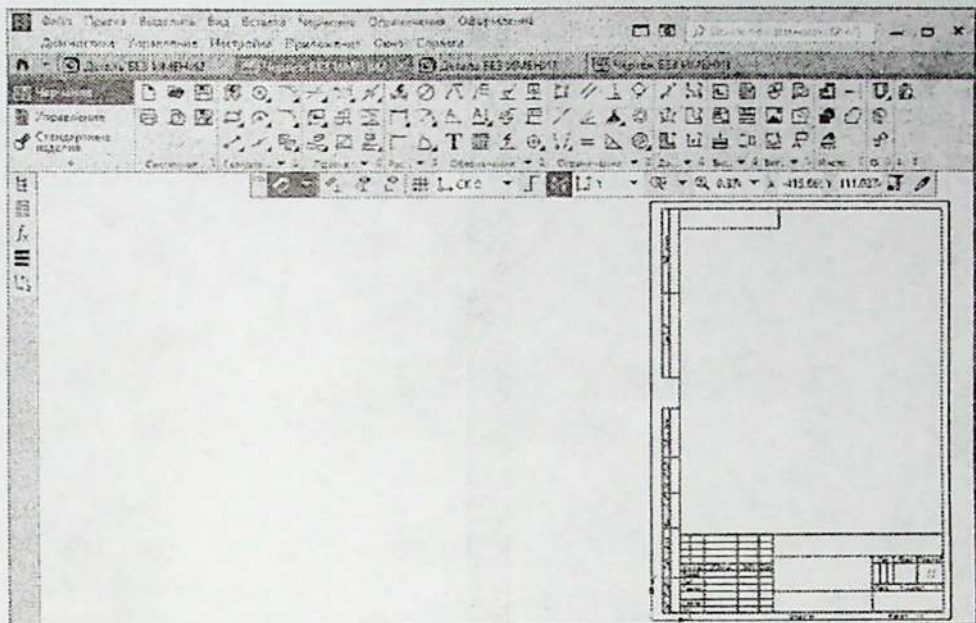
7.6-rasm. Qidiruv maydoni.

Boshqaruv panellar maydoni – bir-biriga bog‘langan bir nechta boshqaruv elementlar panelini o‘z ichiga oladi. Jumladan parametr-lar paneli va obyekt daraxtini qurish. Panellarni bo‘lib, grafik may-doni sohasiga o‘ngga yoki chapga joylashtirish mumkin. Boshqaruv panellarini yoqish yoki o‘chirish menyuning **Sozlamalar paneli** (Настройка панели) buyruqlari orqali amalga oshiriladi.



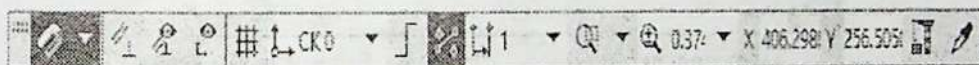
7.7-rasm. Boshqaruv paneli.

Grafik maydon – «Компас-3Д» oynasining ko‘p qismini egal-laydi. Joriy hujjatning mazmunini ko‘rsatish va u bilan ishlash, unda chizish obyektlarini yaratish, tahrirlash yoki modellashtirish, matnli hujjatlarning matnini, spetsifikatsiyalarni kiritish va tahrirlash uchun xizmat qiladi.



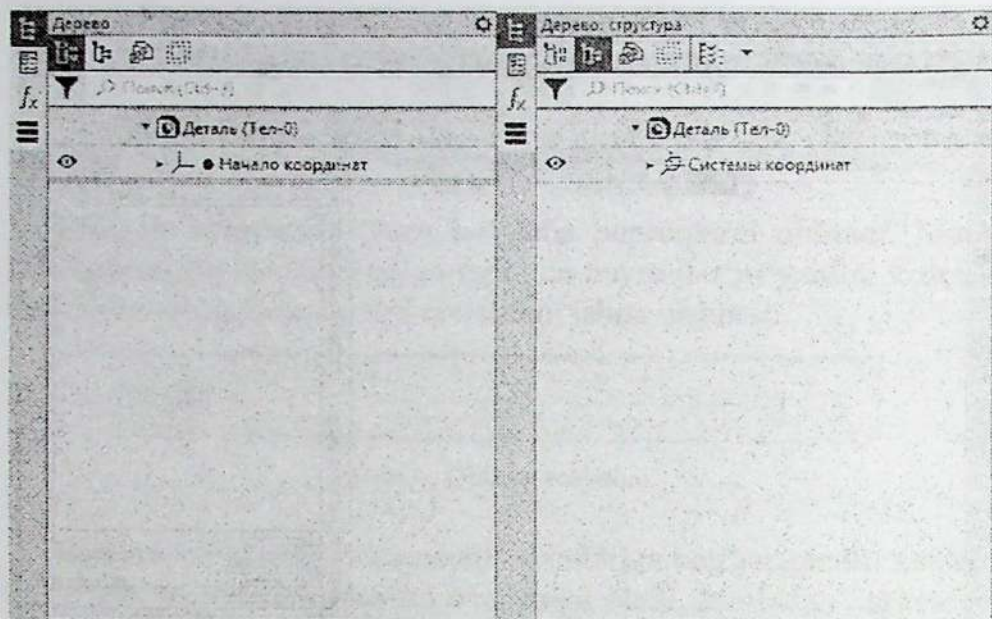
7.8-rasm. Grafik maydon.

Tezkor kirish paneli – tasvir va boshqalarning boshqaruv buyruqlari tartibini tanlashni o‘z ichiga oladi. Panelning tarkibi kontekst menyusiga bog‘liq bo‘ladi. Bu panelni Grafika maydonining yuqori qismiga gorizontal joylashtirish mumkin.



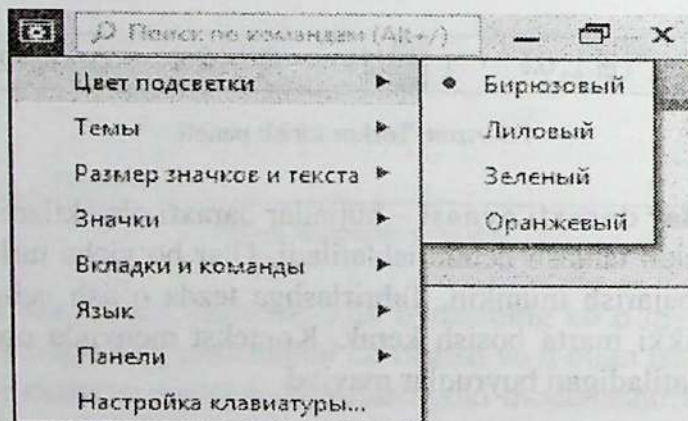
7.9-rasm. Tezkor kirish paneli.

Hujjatlar daraxti oynasi – hujjatlar daraxti obyektlarni bittadan yoki guruhlab tanlash uchun ishlatiladi. Ular bo‘yicha turli xil operatsiyalar bajarish mumkin. Tahrirlashga tezda o‘tish uchun daraxt obyektini ikki marta bosish kerak. Kontekst menyuda obyektlarda tez-tez ishlatiladigan buyruqlar mavjud.



7.10-*расм.* Нуйжатлар дарaxти oynasi.

Интерфейсни sozlash – «Компас-3Д» интерфейсида ishlatiladigan ranglarni Sozlash (Настройка) tugmasi yordamida tanlash mumkin (7.11-*расм.*).



7.11-*расм.* Sozlash paneli.

Rangni sozlash uchun ushbu menyuda quyidagi buyruqlardan foydalaniladi:

Orqa fon rangi – interfeys elementlarini ajratib ko'rsatish uchun ular bilan ishlashda (tugmalar, maydonlar, ro'yxatlar va boshqalar) rang tanlash imkonini beradi. Mavjud bo'lgan variantlar:

- och yashil;
- binafsha;
- yashil;
- to'q sariq.

Mavzular – «Компас-3Д»ning umumiy oynasining mavjud bo'lgan ranglarini tanlash imkonini beradi:

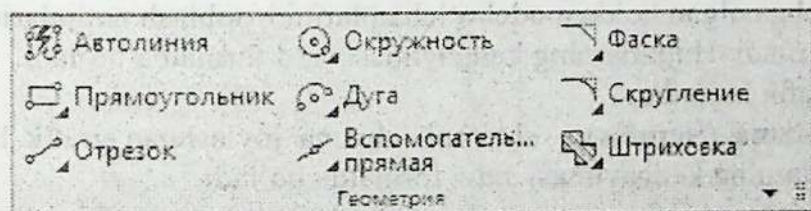
- yorqin;
- qorong'i.

Belgilar – uskunalar maydonida belgilarni tanlash imkonini beradi:

- bir rangli;
- turli rangli.

Panelda buyruq nomlarining ko'rinishi:

Panel faqat tugmachalarni yoki buyruq nomlari bo'lgan tugmalar-ni ko'rsatishi mumkin. Panelning kengligini mos ravishda o'zgartirish uchun uning o'ng chegarasini «sudrab o'tkazamiz». Panel avtomatik ravishda minimallashtiriladi yoki maksimallashtiriladi.

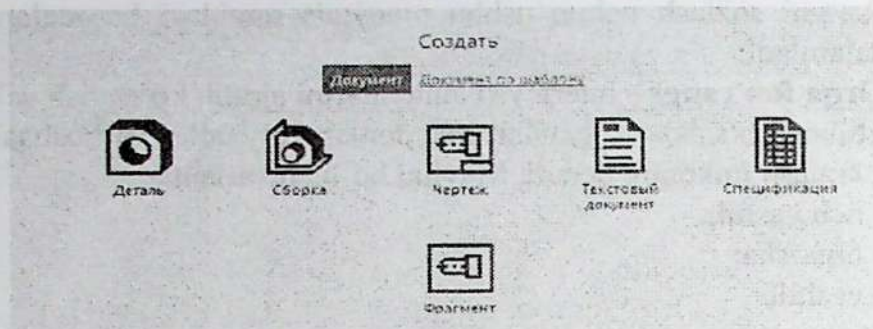


7.12-rasm. Buyruqlarning ko'rinishi.

Yaratiladigan hujjat turlari quyidagilardan iborat:

«Компас-3Д» tizimida yaratilgan hujjat turi ma'lumot turiga bog'liq bo'lishi kerak. Hujjatning har bir turiga mos keladigan fayl nomi, kengaytmasi va o'z piktogramma belgisi mavjud.

«Компас-3Д» tizimida quyidagilarni yaratish mumkin:



7.13-rasm. Hujjat turlari.

- 3 o'lchovli model;
- grafik hujjatlar;
- matnli hujjatlar;
- spetsifikatsiya.

3 D modellar:

Detal (Деталь) – yig'ilmagan yakka holdagi model. Hujjatning kengaytmasi .m3d formatda bo'ladi.

Yig'ish (Сборка) – bir nechta detallar yig'indisidan hosil bo'lgan model. Hujjatning kengaytmasi .a3d formatda bo'ladi.

Texnologik yig'ish (Технологическая сборка) – texnologik ma'lumotlarni, masalan, yig'ish, bardoshlilik, texnologik obyektlarni hisobga olgan holda model o'lchamlarini hisoblash natijalarini o'z ichiga oladi. Hujjatlarning kengaytmasi .a3d formatda bo'ladi.

Grafik hujjatlar:

Chizma (Чертеж) – shtampli chizma joylashgan grafik hujjat. Hujjatlarning kengaytmasi .cdw formatda bo'ladi.

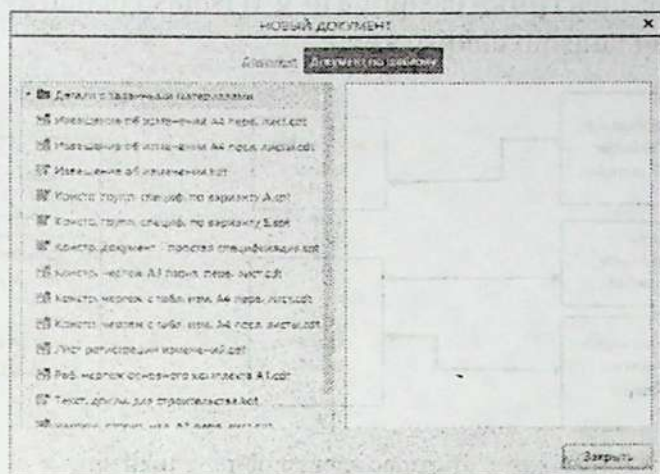
Fragment (Фрагмент) – grafik hujjatning qo'shimcha tipi. Hujjatlarning kengaytmasi .frw formatda bo'ladi.

Matnli hujjat va spetsifikatsiya:

Matnli hujjat (Текстовый документ) – matnli ma'lumotlar yozilgan hujjat. Hujjatlarning kengaytmasi .kdw formatda bo'ladi.

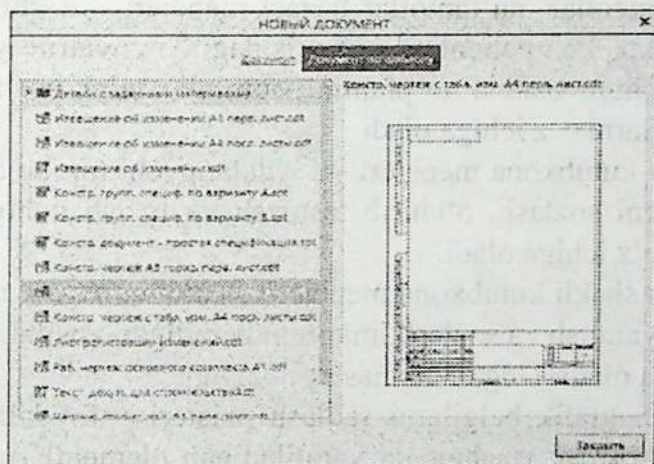
Spetsifikatsiya (Спецификация) – yig'ma modellar haqida ma'lumotlar yoziladigan hujjat. Hujjatlarning kengaytmasi .spw formatda bo'ladi.

Ro'yxatdan kerakli ishchi oyna tanlab olingandan so'ng **Документ по шаблону** bo'limi orqali chizmaga format tanlab olinadi.



7.14-rasm. Format tanlash oynasi.

Masalan, A4 format tanlab olingandan so'ng ekranda quyidagi interfeys paydo bo'ladi.

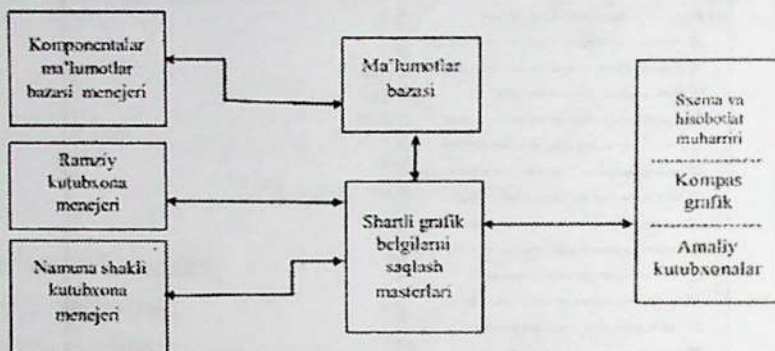


7.15-rasm. Tanlangan format oynasi.

Tanlangan format oynasi ochiladi va A4 formatda grafik tasvir chizish mumkin bo'ladi.

7.2. «Компас электрик»да sxemalar yaratish

«Компас электрик» dasturida to'g'ri ishlash uchun avvalo uning tuzilishi bilan tanishib chiqish kerak.



7.16-rasm. «Компас электрик»ning tuzilishi.

Dasturning asosini dastur ishlatadigan barcha ma'lumotlarni saqlaydigan ma'lumotlar bazasi tashkil etadi. Ma'lumotlar bazasiga uchta menejer birlashtirilgan:

- komponentlar ma'lumotlar bazasi menejeri – dasturda ishlatiladigan elektr komponentlari to'g'risidagi yozuvlarni yaratish va tahrirlash uchun vositalar to'plamini, nomini, sonini, boshqarishlarni va parametrlarni o'z ichiga oladi;

- ramziy kutubxona menejeri – foydalanuvchi tomonidan yaratilgan belgilarni sozlash, o'chirib tashlash va tuzish uchun vositalar to'plamini o'z ichiga oladi.

Namuna shakli kutubxona menejeri – kutubxonaga yangi hisobot shakllarini yaratish va sozlash imkonini beradigan vositalar to'plamini o'z ichiga oladi. Unga 3 ta master ulangan:

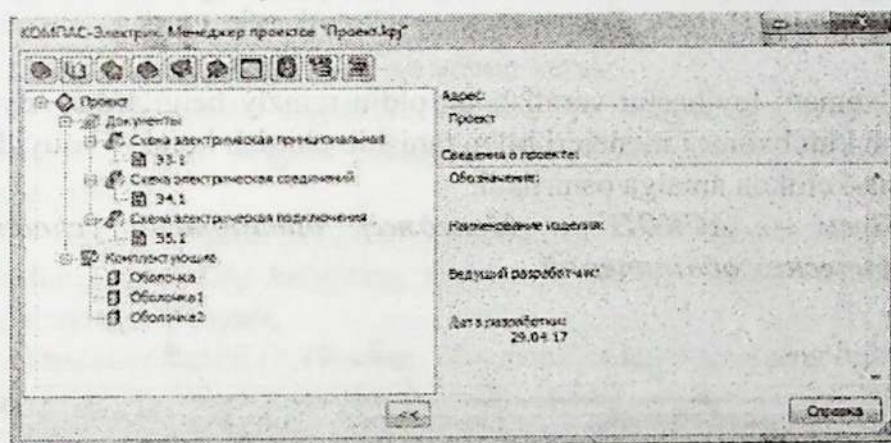
a) shartli grafik belgilarni saqlash masteri – foydalanuvchi tomonidan «Компас график»da yaratiladigan elementlarning ramziy grafik belgilarini ma'lumotlar bazasiga saqlash uchun mo'ljallangan;

b) montaj-kommunikatsion masteri – ma'lumotlar bazasiga ulanish sxemalari uchun mo'ljallangan elementlarning ramziy grafik belgilarini saqlashga mo'ljallangan;

d) qurilmalarning rasmlarini saqlash masteri – chizmalarda qo‘llaniladigan elektr qurilmalarning umumiy ko‘rinishini saqlash uchun mo‘ljallangan, bu ko‘rinishlar maketlarda va chizmalarda qo‘llaniladi.

Yuqorida tavsiflangan elementlarni o‘z ichiga olgan ma’lumotlar bazasi sxematik va namuna muharriri ishida qo‘llaniladi. Ushbu muharrir mo‘ljallangan mahsulotning elektr jihozlari loyihasi uchun hujjatlarni yaratish va chop etishda xizmat qiladi.

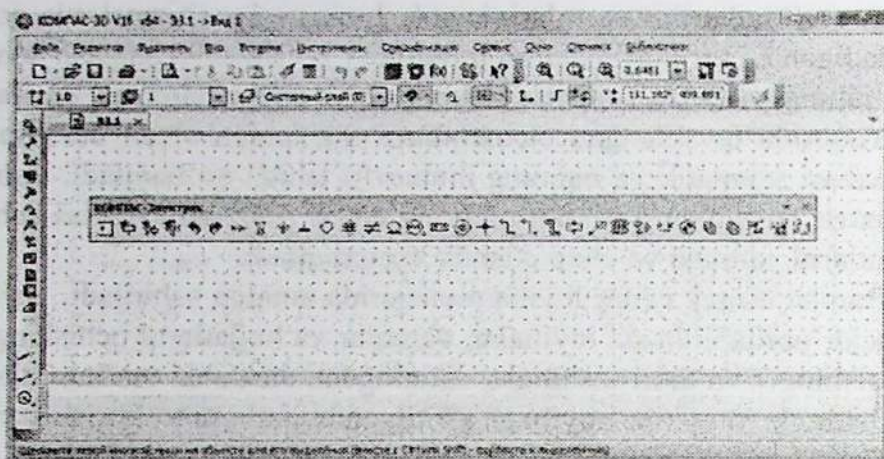
Barcha asosiy ishlar loyiha menejerida amalga oshiriladi. Ushbu menejer yordamida siz loyihalar, varaqlar va hujjatlarni ochishingiz, yaratishingiz va tahrirlashingiz, shuningdek dastlabki ma’lumotlarni kiritishingiz mumkin. Loyihaga kiritilgan hujjatlarni to‘ldirish sxema va hisobot muharriri yordamida amalga oshiriladi.



7.17-*расм.* «Компас электрик»ning loyihasi menejeri.

Menejer bir vaqtning o‘zida bir nechta loyihalarni ochishi va ularning ustida ishlashi mumkin, u loyihalar orasidagi sxemalarni nusxalash, boshqa yerga o‘tkazish imkonini beradi. Bunda faqat bitta loyiha varag‘i faol bo‘lishi mumkin. Menejer obyektlariga quyidagilar kiradi:

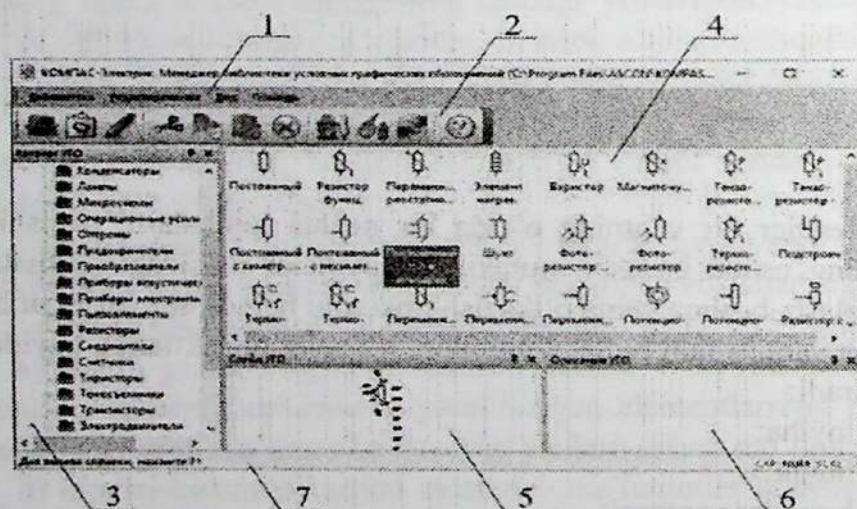
- loyiha;
- hujjat;
- hujjatlar varaqasi;
- komponentlar.



7.18-rasm. «Компас электрик»ning uskunalar paneli.

Sxemani loyihhasini yaratishdan oldin ramziy belgilardan foydalanish kutubxonasi menejeri bilan tanishib chiqish kerak. U quyidagi ketma-ketlikda amalga oshiriladi.

Пуск → АСКОН → Менеджер библиотеки условных графических обозначений



7.19-rasm. Ramziy belgilar kutubxonasining menejeri.

Ushbu menejering interfeys elementlariga qisqacha tavsif beramiz:

1) asosiy menyu (7.19-rasm, 1) – kontekst menyusi sahifalarining nomlarini o‘z ichiga oladi (Kutubxona, Tahrirlash, Ko‘rish, Yordam). Kerakli menyu sahifasini kengaytirish uchun sichqoncha bilan nomiga bosish kifoya;

2) uskunalar paneli (7.19-rasm, 2) – menejer buyruqlarini chaqirish tugmalaridan iborat;

3) ramziy belgilar katalogi (7.19-rasm, 3) – dastur bazasida mavjud bo‘lgan ramziy belgilar kutubxonasi kataloglarini namoyish qilish uchun mo‘ljallangan;

4) ramziy belgilar display maydoni (7.19-rasm, 4) – kataloglarda mavjud bo‘lgan ramziy belgilar guruhlarini matnlar bilan piktogramma ko‘rinishida namoyish qilish uchun kerak;

5) ramziy belgilar slaydi (7.19-rasm, 5) – belgilangan katalogdan tanlangan ramziy belgilar piktogrammasini namoyish qilish uchun xizmat qiladi;

6) ramziy belgilar tavsifi (7.19-rasm, 6) – belgilangan katalogda ta’kidlangan ramziy belgining matn tavsifini ko‘rsatish va kiritish uchun xizmat qilinadi;

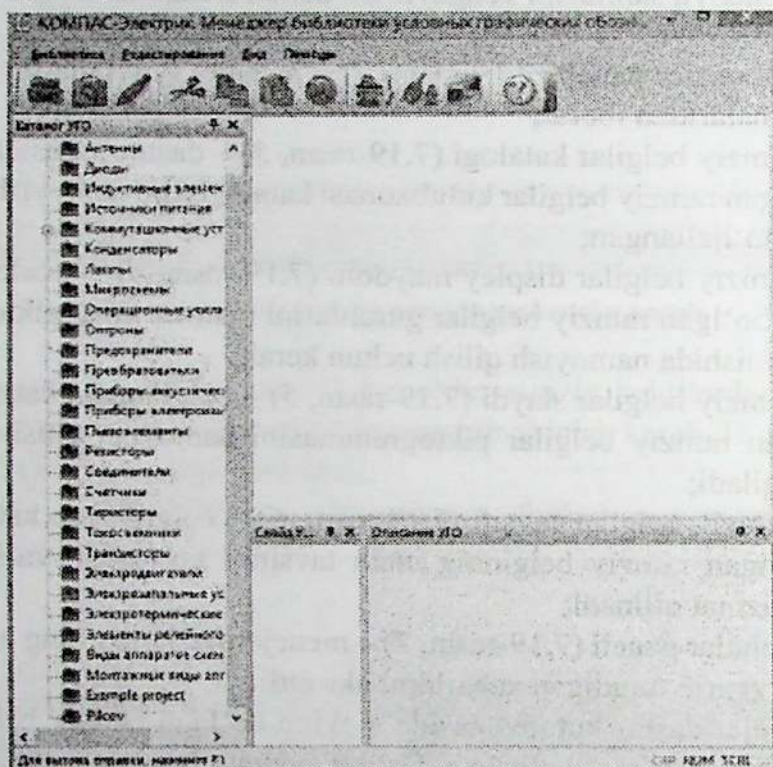
7) holatlar paneli (7.19-rasm, 7) – menejerdan kursorning holatiga qarab o‘zgarib turadigan xabarlarini aks ettiradi.

Menejer dastur kutubxonasida mavjud bo‘lgan ramziy belgilarni tahrirlash uchun mo‘ljallangan. Ushbu menejer buning uchun bir qator vositalarni o‘z ichiga oladi. Eng keraklilariga tavsif beramiz.

7.3. «Компас электрик»da kataloglar bilan ishlash

Katalog yaratish  – bu yangi kataloglar yoki ostki kataloglarni yaratishga imkon beruvchi vosita. Ushbu buyruqni amalga oshirish uchun 7.20-rasmdagi kerakli maydonda sichqonchaning tugmasini bosishingiz kerak. Yaratilgan katalog «Yangi katalog» deb nomlanadi, uni tahrirlash mumkin. Ushbu katalog ilgari sichqoncha bilan tanlangan katalogga nisbatan iyerarxiyaning keyingi darajasida joylashgan bo‘ladi.

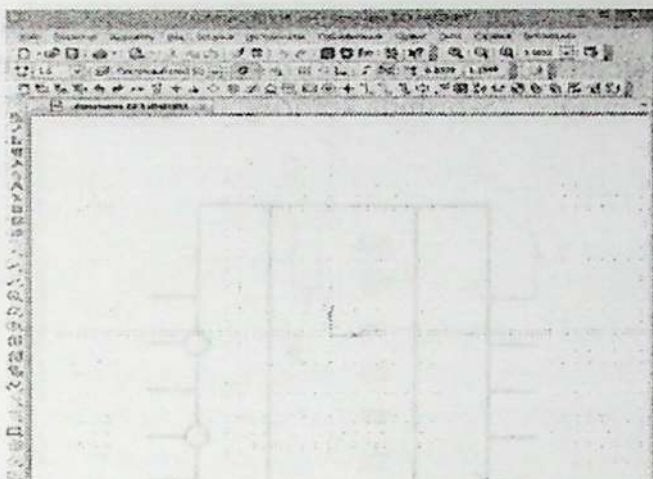
«Tahrirlash»  – ramziy belgilarni tahrirlash vositasi allaqachon yaratilgan bo‘ladi. Tahrirlash «Компас график» amalga oshiriladi. Tahrirlashdan so‘ng ramziy belgilar tegishli master yordamida kutubxonaga saqlanishi kerak.



7.20-rasm. Yaratilgan katalog.

Chizilishi kerak bo‘lgan barcha ramziy belgilar komponentlari ushbu katalogga saqlanadi.

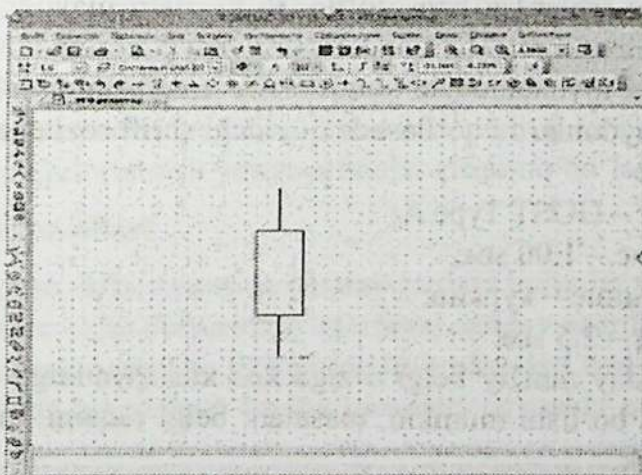
Qarshilikka misol yordamida ramziy belgilar yaratish jarayonini ko‘rib chiqamiz. Ramziy belgilar komponentini yaratishni boshlash uchun «Sxema va hisobot muharriri»ni ochish kerak, **Меню** yorlig‘idan **Файл** → **Создать** → **Новые документы** tanlanadi. Har bir komponent uchun o‘zining fragmenti kerak bo‘ladi.



7.21-*рasm*. Ramziy belgilar yaratish uchun ish maydoni.

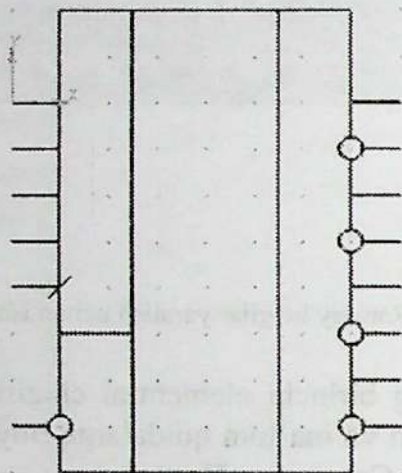
Ramziy belgining birinchi elementini chizishdan oldin, tarmoq sozlamalarini bajarish va ma'lum qoidalarga rioya qilish kerak. Tarmoqni sozlash uchun Сервис → Параметры → Текущее окно → Сетка tanlanadi, uning qadami X va Y bo'ylab 1 mm ga teng bo'lishi lozim. Nuqtalar orasidagi piksellar soni 2 ga teng bo'ladi.

10 dan 4 mm gacha bo'lgan qarshilik uchun to'g'ri chizilgan belgi 7.22-rasmda keltirilgan.



7.22-*рasm*. Ramziy belgi qarshiligining chizmasi.

Ramziy belgili raqamli mikrosxemalarni yaratishda ularning vertikal yoʻnalishiga rioya qilish shart emas. Mikrosxemaning toʻgʻri joylashishiga misol 7.23-rasmda keltirilgan.



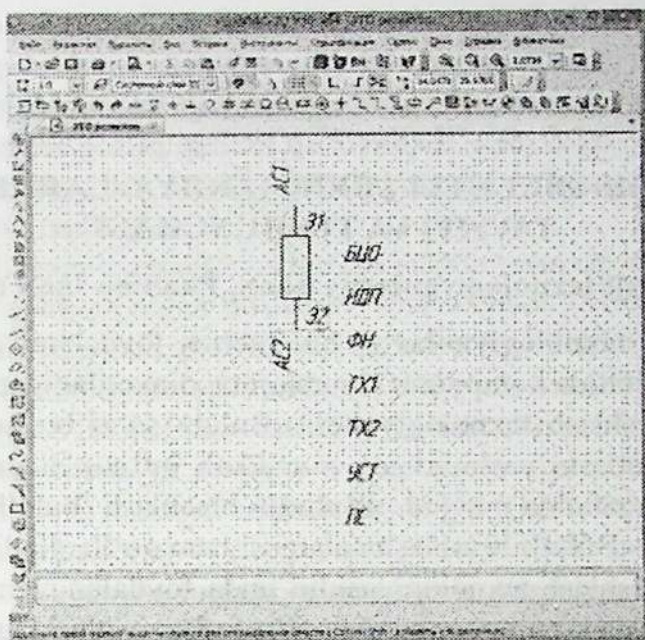
7.23-rasm. Ramziy belgili mikrosxemaning toʻgʻri joylashishi.

Avval yaratilgan ramziy belgilarning grafik qismida maʼlum matn maydonlarini qoʻllash kerak. Har bir ramziy belgi uchun, baʼzilari kamdan kam ishlatilsa ham, toʻliq 36 ta matn maydonini qoʻllash maqsadga muvofiqdir. Har qanday matn maydonining koʻrinishi yoki koʻrinmasligi foydalanuvchi tomonidan oʻrnatilishi mumkin.

Matn maydonlarini qoʻllashda quyidagi shrift sozlamalariga rioya qilish kerak:

- шрифт – GOST Type A;
- сужение – 1.00 мм;
- начертание – курсив;
- высота – 2.5 мм.

Har qanday ramziy belgi oʻziga xos xususiyatlarga ega boʻlgan noyob qism boʻlishi mumkin, masalan: belgi raqami, mos yozuvlar belgisi va h.k.



7.24-rasm. Ramziy belgi qarshiligi.

«Компас электрик» – bu loyihalash hujjatlarini yaratish jarayonini sezilarli darajada avtomatlashtirish, tezlashtirish va soddalashtirishga imkon beradigan juda universal tizim hisoblanadi.

Albatta, dastur ma'lumotlar bazasi bo'sh bo'lganda va ramziy belgilar komponentlarini yaratib, uni to'ldirish kerak bo'ladi. Elektron uchun birinchi hujjatlarni yaratish jarayoni uzoq davom etadi. Biroq, ma'lumotlar bazasini to'ldirish bilan ikkinchi va keyingi sxemalar uchun hujjatlarni yaratish jarayoni ancha qisqaroq bo'ladi.

Nazorat savollari

1. «Компас -3Д» dasturida qanday turdagi hujjatlar yaratiladi?
2. «Компас -3Д» dasturining interfeysi qanday qismlardan iborat?
3. «Компас электрик» tizimining imkoniyatlari qanday?
4. «Компас электрик» loyiha menejeri oynasida qanday vazifa bajariladi?
5. Grafik oynada hujjat formati qanday ketma-ketlikda tanlanadi?

8-BOB. «BULUTLI TEXNOLOGIYA»LARNING TAVSIFI VA TUSHUNCHASI

8.1. «Bulutli texnologiya»larda SaaS, PaaS va IaaS modellari

«Bulutli texnologiya»lar – bu internet-foydalanuvchiga internet-xizmat sifatida kompyuter resurslari taqdim etiladigan ma'lumotlarni qayta ishlash texnologiyalari. «Bulut» so'zi bu yerda barcha texnik tafsilotlarni yashiradigan murakkab infratuzilmani ifodalovchi metafora sifatida mavjud. «Bulutli» hisoblash (bulutli hisoblash, shuningdek, «Bulutli hisoblash» atamasi ham qo'llaniladi) ma'lumotlarni qayta ishlash texnologiyasidir, unda foydalanuvchiga internet xizmati sifatida kompyuter resurslari va imkoniyatlari taqdim etiladi. Foydalanuvchi o'z ma'lumotlariga kirish huquqiga ega, ammo uni boshqara olmaydi va infratuzilma, operatsion tizim va o'zi ishlayotgan dasturiy ta'minot haqida qayg'urmasligi kerak. «Bulut» atamasi kompyuter tarmog'i diagrammasidagi internet tasviriga asoslangan metafora yoki barcha texnik tafsilotlar yashiringan murakkab infratuzilma tasviri sifatida ishlatiladi. «Bulutli hisoblash» paradigma bo'lib, unda ma'lumot internet serverlarida doimiy ravishda va mijoz tomonidan vaqtincha saqlanadi, masalan, shaxsiy kompyuterlarda, o'yin pristavkalari, noutbuklar, smartfonlarda va hokazo. «Bulutli» ma'lumotlarni konsepsiya sifatida qayta ishlash quyidagilarni o'z ichiga oladi:

- 1) xizmat sifatida infratuzilma;
- 2) xizmat sifatida platforma;
- 3) dastur xizmat sifatida;
- 4) ma'lumotlar xizmat sifatida;
- 5) xizmat sifatida ish joyi.

Internetning foydalanuvchilarga ma'lumotlarni qayta ishlovga bo'lgan ehtiyojlarini qondira olishga umumiy ishonch mavjud

bo'lgan boshqa texnologik tendensiyalar. «Bulutli» texnologiyalar uchun eng muhim xususiyat foydalanuvchilarning internet-manbalarini notekis talab qilishidir. Ushbu notekislikni bartaraf etish uchun yana bir oraliq qatlam qo'llaniladi – serverni virtualizatsiya qilish. Shunday qilib, vazifa virtual serverlar va kompyuterlar o'rtasida taqsimlanadi.

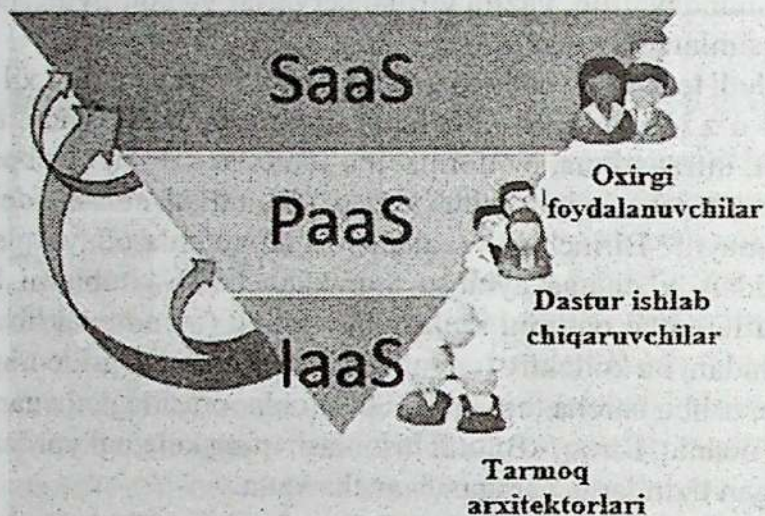
«Bulutli texnologiya»lar xizmatlarni taqdim etuvchi turli xil tushunchalarni o'z ichiga olgan bitta katta tushunchadir. Masalan, dasturiy ta'minot, infratuzilma, platforma, ma'lumotlar, ish joyi va boshqalar. Bularning barchasi nima uchun kerak? «Bulutli hisoblash» deb nima ga aytilmaydi? Birinchisi – mahalliy kompyuterda oflayn hisoblash. Ikkinchidan, xizmatga ayniqsa murakkab hisob-kitoblarni bajarish yoki ma'lumotlar qatorini saqlash buyuriladi (kommunal hisoblash). Uchinchidan, bu kollektiv (tarqatilgan) hisoblashdir (gridcomputing). Amalda, ushbu barcha turdagi hisob-kitoblar orasidagi chegaralar yetarlicha noaniq. Biroq, «Bulutli hisoblash»ning kelajagi yordamchi va tarqatilgan tizimlarga qaraganda ancha katta.

«Bulutli ma'lumotlar»ni saqlash – Onlayn saqlash modeli, unda ma'lumotlar tarmoqda tarqatilgan ko'plab serverlarda saqlanadi va mijozlar, asosan uchinchi tomondan foydalanish uchun taqdim etiladi. O'zining maxsus serverlarida ma'lumotlarni saqlash uchun ushbu maqsadlar uchun maxsus sotib olingan yoki ijaraga olingan modeldan farqli o'laroq, serverlar soni yoki har qanday ichki tuzilishi umuman mijozga ko'rinmaydi. Ma'lumotlar mijoz nuqtayi nazaridan bitta yirik virtual serverni ifoda etadigan bulut deb nomlangan holda saqlanadi va qayta ishlanadi. Jismoniy jihatdan, bunday serverlar bir-biridan uzoqroq, turli qit'alarga qadar joylashgan bo'lishi mumkin. «Bulut» nima ekanligini tushunish uchun ushbu sonning tarixidan boshlash kerak. Biz bu texnologiya haqiqatan ham yangi g'oyalar toifasiga kiradimi yoki bu g'oya unchalik yangi emasmi ekanligini tushunishimiz kerak.

«Bulutli hisoblash» – tarqatilgan ma'lumotlarni qayta ishlash texnologiyasi, unda foydalanuvchiga internet xizmati sifatida kompyuter resurslari va imkoniyatlari taqdim etiladi (8.1-rasm).

«Bulutli texnologiya»larning asosiy turlari quyidagilardan iborat:

- 1) xizmat sifatida infratuzilma (IaaS – Infrastructure-as-a-Service);
- 2) xizmat sifatida platforma (PaaS – Platform-as-a-Service);
- 3) dastur xizmat sifatida (SaaS – Software-as-a-Service).



8.1-rasm. Foydalanuvchilar uchun «Bulut» bilan ishlash modellari.

Ushbu texnologiyalarning har birini batafsil ko‘rib chiqamiz.

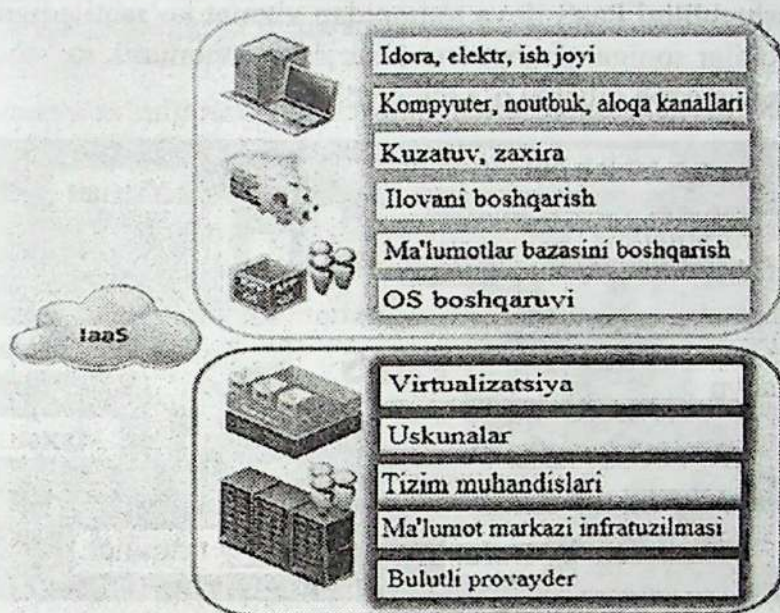
Xizmat sifatida infratuzilma (IaaS):

IaaS – «Bulutli hisoblash» tushunchasiga asoslangan xizmat sifatida kompyuter infratuzilmasini taqdim etish. IaaS uchta asosiy tarkibiy qismdan iborat:

- uskuna (serverlar, saqlash tizimlari, mijoz tizimlari, tarmoq uskunalari);
- operatsion tizimlar va tizim dasturlari (virtualizatsiya, avtomatlashtirish, asosiy resurslarni boshqarish vositalari);
- o‘rta dastur (masalan, tizimlarni boshqarish uchun).

IaaS virtualizatsiya texnologiyasiga asoslangan bo‘lib, uskunadan foydalanuvchiga uni biznesning hozirgi ehtiyojlariga javob beradigan qismlarga ajratish va shu bilan mavjud hisoblash quvvatidan foydalanish samaradorligini oshirishga imkon beradi. Foydalanuvchi (kom-

paniya yoki dasturiy ta'minot ishlab chiqaruvchisi) faqat server vaqti, disk maydoni, tarmoqning o'tkazish qobiliyati va ishlashi uchun zarur bo'lgan boshqa manbalar uchun haq to'lashi kerak. Bundan tashqari, IaaS mijozga bitta integratsiyalashgan platformada boshqarish funksiyalarini to'liq taqdim etadi (8.2-rasm).



8.2-rasm. IaaSning xizmat ko'rsatish infratuzilmasi.

IaaS kapital va operatsion xarajatlarni kamaytirgan holda, murakab ma'lumotlar markazi, mijozlar va tarmoq infratuzilmalarini qo'llab-quvvatlash zaruriyatini yo'q qiladi.

Xizmat sifatida platforma (PaaS):

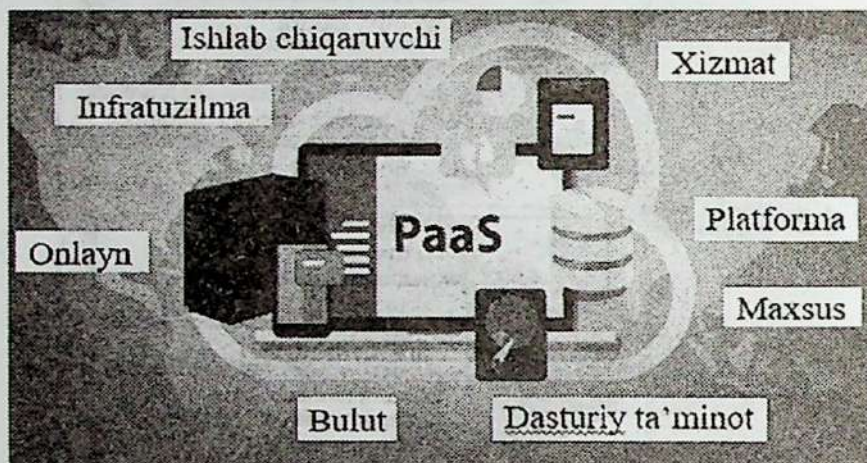
PaaS – bu veb-ilovalarni xizmat sifatida ishlab chiqish, sinovdan o'tkazish, joylashtirish va qo'llab-quvvatlash uchun o'rnatilgan platformani taqdim etish.

Veb-ilovalarni o'rnatish uchun ishlab chiqaruvchidan apparat va dasturiy ta'minot sotib olishning va ularni qo'llab-quvvatlashni tashkil qilishning hojati yo'q. Mijoz uchun kirish ijara asosida tashkil qilinishi mumkin.

Ushbu yondashuv quyidagi afzalliklarga ega:

- kengayish qobiliyati
- xatolarga bardoshlilik;
- virtualizatsiya;
- xavfsizlik.

Masshtabligi PaaS ilova tomonidan xizmat ko'rsatiladigan foydalanuvchilar soniga qarab, zarur resurslarni avtomatik ravishda taqsimlashni va ozod qilishni o'z ichiga oladi.



8.3-rasm. PaaS platformasining xizmat ko'rsatish sxemasi.

PaaS veb-ilovalarni ishlab chiqish, sinovdan o'tkazish, joylashtirish va qo'llab-quvvatlash uchun o'rnatilgan platforma sifatida veb-ilovalarni ishlab chiqish, sinovdan o'tkazish va joylashtirish bo'yicha operatsiyalarning butun ro'yxati bitta integratsiyalashgan muhitda bajarilishi mumkin.

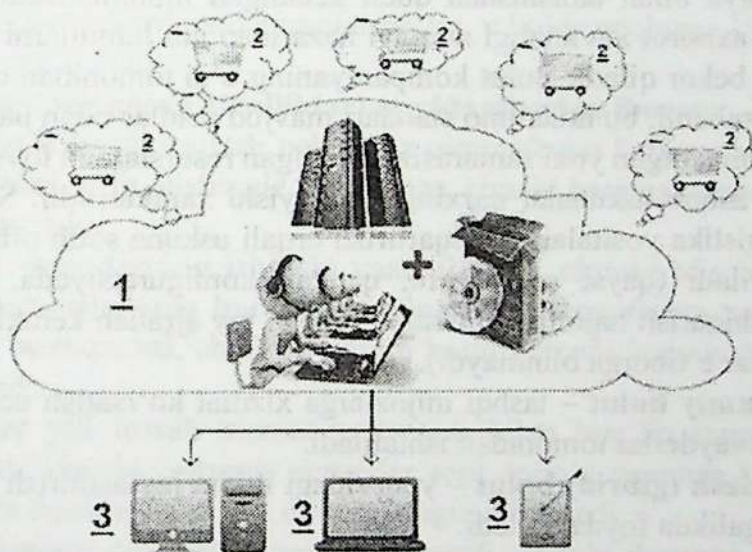
Dastur xizmat sifatida (SaaS).

SaaS – dasturni oxirgi foydalanuvchilarga talab bo'yicha xizmat sifatida taqdim etishni o'z ichiga oladigan dasturni tarqatish modeli. Bunday dasturga kirish tarmoq va ko'pincha internet-brauzer orqali amalga oshiriladi.

SaaS modelining xususiyati:

- ilova masofadan foydalanish uchun moslangan;
- bir nechta mijoz bitta dasturdan foydalanishi mumkin;
- xizmat uchun to'lov har oylik obuna sifatida yoki tranzaksiyalarning umumiy miqdoridan kelib chiqib olinadi;
- dasturni qo'llab-quvvatlash allaqachon to'lovga kiritilgan;
- dastur xodimlarga mijozlarga muammosiz va oshkora yangilanishi mumkin.

Dasturiy ta'minotni ishlab chiquvchilar nuqtayi nazaridan, SaaS modeli mijozga dasturiy ta'minotni saqlash, nusxalash va o'rnatish imkoniyatiga ega emasligi sababli dasturiy ta'minotdan litsenziyasiz foydalanish bilan samarali kurashishga imkon beradi. Aslida, SaaS dasturiy ta'minoti ichki axborot tizimlariga yanada qulay va tejamkor muqobil sifatida qaralishi mumkin.



8.4-rasm. SaaS xizmatlarining ishlash prinsipi.

1. SaaS platforma;
2. SaaS internet do'kon;
3. SaaS foydalanuvchilar va mijozlar.

SaaS mantig'ini rivojlantirish – bu tushuncha Vaas (ish joyi xizmat sifatida – xizmat sifatida ish joyi). Ya'ni mijoz o'z ixtiyorida barcha kerakli dasturiy ta'minot bilan to'liq jihozlangan virtual ish stansiyasini oladi.

8.2. Ma'lumotni saqlash va zaxiralash

Bulut xizmatlarining har uch turi bir-biri bilan bog'langan va birlashtirilgan tuzilmani anglatadi. Xizmatlarni taqdim etishning turli usullariga qo'shimcha ravishda, bulutli tizimlarni joylashtirishning bir nechta variantlari mavjud.

Xususiy bulut – mijoz va xizmat ko'rsatuvchi provayder bo'lgan bitta kompaniya ichida xizmatlarni ko'rsatish uchun foydalaniladi. Bu kompaniya tashkilot ichida o'zi uchun yaratganda «bulutli konsepsiya»ni amalga oshirish uchun imkoniyat. Bulut mijozlar ushbu konsepsiya bilan tanishishda duch keladigan muhim masalalardan birini – axborot xavfsizligi nuqtayi nazaridan ma'lumotlarni himoya qilishni bekor qiladi. Bulut kompaniyaning o'zi tomonidan cheklanganligi sababli, bu muammo standart mavjud usullar bilan hal qilinadi. Ishlamayotgan yoki samarasiz ishlatilgan resurslardan foydalanish sababli asbob-uskunalar narxining pasayishi xarakterlidir. Shuningdek, logistika vositalarini qisqartirish orqali uskuna sotib olish narxi pasaytiriladi (qaysi serverlarni, qanday konfiguratsiyada, qanday ishlab chiqarish hajmini, har safar qancha joy ajratish kerakligini va hokazolar e'tiborga olinmaydi).

Umumiy bulut – tashqi mijozlarga xizmat ko'rsatish uchun bulutli provayderlar tomonidan ishlatiladi.

Aralash (gibrid) bulut – yuqoridagi ikkita joylashtirish modelidan birgalikda foydalaniladi.

Internet odamlar hayotining ajralmas qismiga aylandi va muntazam ravishda turli xil vazifalarni bajarishga yordam beradigan yangi imkoniyatlar paydo bo'ldi. Yangilik – bu turli sohalarda qo'llanilishi mumkin bo'lgan bulutli texnologiyalar: ta'lim, tibbiyot, logistika va boshqalar.

8.3. Bulutli texnologiya xizmatlarining afzalliklari

Bulutli texnologiyalar foydalanuvchiga qurilmaga dasturlarni o'rnatmasdan ma'lumotlarga kirish imkoniyatini anglatadi, chunki barcha qo'llab-quvvatlovchilar serverlar tomonidan ta'minlanadi. Bunday xizmatlar bepul va pullik bo'lishi mumkinligini bilish juda muhim va barchasi so'rovlarga bog'liq. Bulutli texnologiyalardan foydalanish nimani anglatishini tushunish uchun ularning oddiy sharoitlardan farqini tushunishingiz kerak:

1) masalan, elektron pochta mijosi kompyuterga o'rnatilganda va barcha ma'lumotlar qattiq diskda saqlanganda elektron pochta oling. Bunday holda, foydalanuvchi fayllar bilan nima qilishni hal qiladi;

2) agar biz brauzer orqali foydalanadigan pochta haqida gapiradigan bo'lsak, unda bu allaqachon texnologiya hisoblanadi. Bu yerda aniqki, agar serverda muammolar bo'lsa, u holda pochtaga kirish imkoniyati yo'qoladi.

Texnologiyaning afzalliklari quyidagilardan iborat:

1) ma'lumotni saqlash uchun siz qimmatbaho kompyuter va aksessuarlarni sotib olishingiz shart emas, chunki hamma narsa «bulutda» saqlanadi;

2) kompyuterning ishlashi yaxshilanadi, chunki ofis ishlarida va boshqa sohalarda bulutli texnologiyalar masofadan turib dasturlarni boshqaradi, shuning uchun kompyuterda juda ko'p bo'sh joy qoladi;

3) har yili texnik xizmat ko'rsatish bilan bog'liq muammolar kamayadi, chunki jismoniy serverlar soni doimiy ravishda kamayib boradi va dasturiy ta'minot doimo yangilanib turadi;

4) dasturni sotib olish narxi kamayadi, chunki dasturni «bulut» uchun faqat bir marta sotib olish kerak va ba'zida siz uni ijaraga buyurtma qilishingiz ham mumkin;

5) bulutli texnologiyalar saqlanadigan ma'lumotlar miqdorida cheklolarga ega emas. Aksariyat hollarda bunday xizmatlar hajmi millionlab gigabaytni tashkil qiladi;

6) dasturlar avtomatik ravishda yangilanadi, shuning uchun yuklab olingan dasturlarda bo'lgani kabi, bunga rioya qilishning hojati yo'q;

7) «bulut»dan har qanday operatsion tizimda foydalanish mumkin, chunki dasturlarga kirish veb-brauzerlar orqali amalga oshiriladi;

8) yangi bulutli texnologiyalar sizga har doim hujjatlarga kirish huquqini beradi, chunki asosiysi internetning mavjudligi;

9) yaxshi xavfsizlik va ma'lumotlarni yo'qotishdan himoya qilish, chunki yuborilgan ma'lumotlar avtomatik ravishda saqlanadi va nusxalari zaxira serverlarga tashlanadi.

Nazorat savollari

1. Bulutli texnologiyalar deb qanday texnologiyalarga aytiladi?
2. Bulutli ma'lumotlarni saqlash deganda nima tushuniladi?
3. IaaS bulutli hisoblashning xususiyati qanday?
4. SaaS modeli qanday xususiyatga ega?
5. PaaSning vazifasi nimadan iborat?

9-BOB. TEXNIK TIZIMLARDA AXBOROTNI HIMOYA QILISHNING ZAMONAVIY USULLARI

9.1. Texnik tizimlarda axborot xavfsizligini ta'minlash

Axborot tizimining xavfsizligi deganda tizimning normal ishlash jarayonida tasodifiy yoki qasddan o'chrish, ma'lumotni o'g'irlashga (ruxsatsiz olishga) urinishlar, uning tarkibiy qismlarini o'zgartirish yoki jismoniy yo'q qilish xavfsizligi tushuniladi.

Axborot xavfsizligi tahdidi deganda boshqariladigan tizimning, shuningdek dasturiy ta'minotning axborot resurslarini buzish, ruxsatsiz foydalanish yoki hatto yo'q qilishga olib kelishi mumkin bo'lgan harakatlar tushuniladi.

Axborotning himoyasi deb boshqarish va ishlab chiqarish faoliyatining axborot xavfsizligini ta'minlovchi va tashkilot axborot zaxiralarining yaxlitligi, ishonchliligi, foydalanish osonligi va maxfiyligini ta'minlovchi qat'iy reglamentlangan dinamik texnologik jarayonga aytiladi.

Axborot xavfsizligini ta'minlash. Axborot xavfsizligini ta'minlash – bu foydalanuvchining axborotlarini himoyalashga qo'yilgan me'yor va talablarni bajarishidir. Axborot xavfsizligi esa axborot foydalanuvchilariga va ko'plab axborot tizimlariga zarar keltiruvchi tabiiy yoki sun'iy xarakterga ega tasodifiy va uyushtirilgan ta'sirlardan axborotlarni va axborot kommunikatsiya tizim obyektlarining himoyalanganligidir.

Login tushunchasi. Login – shaxsning, o'zini axborot-kommunikatsiya tizimiga tanishtirish jarayonida qo'llaniladigan belgilar ketma-ketligi bo'lib, axborot-kommunikatsiya tizimidan foydalanish huquqiga ega bo'lish uchun foydalanuvchining maxfiy bo'lmagan qayd yozuvi hisoblanadi.

Parol tushunchasi. Parol – uning egasi haqiqiyligini aniqlash jarayonida tekshiruv axboroti sifatida ishlatiladigan belgilar ketma-ketligi. U kompyuter bilan muloqot boshlashdan oldin, unga klaviatura yoki identifikatsiya kartasi yordamida kiritiladigan harfli, raqamli yoki harfli-raqamli kod shaklidagi maxfiy soʻzdan iborat.

Avtorizatsiya tushunchasi. Avtorizatsiya – foydalanuvchining resursdan foydalanish huquqlari va ruxsatlarini tekshirish jarayoni. Bunda foydalanuvchiga hisoblash tizimida baʼzi ishlarni bajarish uchun muayyan huquqlar beriladi. Avtorizatsiya shaxs harakati doirasini va u foydalanadigan resurslarni belgilaydi.

Roʻyxatdan oʻtish tartibi. Roʻyxatdan oʻtish – foydalanuvchilarni roʻyxatga olish va ularga dasturlar va maʼlumotlarni ishlatishga huquq berish jarayoni. Ayrim veb-saytlar foydalanuvchilarga qoʻshimcha xizmatlarni olish va pullik xizmatlarga obuna boʻlish uchun roʻyxatdan oʻtishni, yaʼni oʻzi haqida ayrim maʼlumotlarni kiritishni (anketa toʻldirishni) hamda login va parol olishni taklif qiladilar. Foydalanuvchi roʻyxatdan oʻtgandan soʻng tizimda unga qayd yozuvi (account) yaratiladi va unda foydalanuvchiga tegishli axborotlar saqlanadi.

Login va parolga ega boʻlish shartlari. Biror shaxs oʻzining login va paroliga ega boʻlishi uchun u birinchidan axborot-kommunikatsiya tizimida roʻyxatdan oʻtgan boʻlishi kerak va shundan soʻng u oʻz logini va parolini oʻzi hosil qilishi yoki tizim tomonidan berilgan login parolga ega boʻlishi mumkin. Login va parollar maʼlum uzunlikdagi belgilar ketma-ketligidan tashkil topadi. Login va parollarning uzunligi va qiyinligi uning qanchalik xavfsizligini taʼminlaydi.

Login va parolni buzish – bu buzgʻunchining biror-bir maqsad yoʻlida axborot-kommunikatsiya tizimi obyektlaridan foydalanish uchun qonuniy tarzda foydalanuvchilarga tegishli login va parollarini buzishdir. Bunda maxsus dastur yordamida login va parollar generatsiya qilib topiladi. Login va parollarning uzunligi bu jarayonning uzoq vaqt davom etishiga yoki generatsiya qilaolmasligiga ishora boʻladi.

Login va parolni oʻgʻirlash – bu foydalanuvchilarning maxfiy maʼlumotlari boʻlgan login va parollarga ega boʻlish maqsadida

amalga oshiriladigan internet firibgarligining bir turidir. Bu mashhur brendlar, masalan, ijtimoiy tarmoqlar, banklar va boshqa servislar nomidan elektron xatlarni ommaviy jo'natish yo'li orqali amalga oshiriladi. Xatda odatda tashqi ko'rinishi asl saytdan farq qilmaydigan saytga to'g'ri ishora mavjud bo'ladi. Bunday saytga tashrif buyurgan foydalanuvchi firibgarga akkauntlar va bank hisob raqamlariga kira olishga ega bo'lishga imkon beruvchi muhim ma'lumotlarni bildirishi mumkin. Fishing –ijtimoiy muhandislikning bir turi bo'lib, foydalanuvchilarning tarmoq xavfsizligi asoslarini bilmasligiga asoslangan. Jumladan, ko'pchilik oddiy fakti bilishmaydi: servislar qayd yozuvingiz ma'lumotlari, parol va shu kabi ma'lumotlarni yuborishni so'rab hech qachon xat yubormaydi.

Bugungi kunda yangi zamonaviy texnologiya – kompyuter axborot tizimlarida va ma'lumotlarni uzatish tarmoqlarida axborotni himoya qilish texnologiyasi tug'ilayotgani haqida bahslashish mumkin.

Ushbu texnologiyani amalga oshirish xarajatlar va kuchlarni oshirishni talab qiladi. Biroq bularning barchasi axborot tizimi va axborot texnologiyalariga tahdidlarni real amalga oshirishda yuzaga kelishi mumkin bo'lgan sezilarli darajada yuqori yo'qotishlar va zararlarning oldini olishga imkon beradi.

Faol tahdidlar, uning tarkibiy qismlariga maqsadli ravishda ta'sir o'tkazish orqali axborot tizimining normal ishlashini buzishga qaratilgan. Faol tahdidlarga quyidagilar kiradi:

- kompyuter yoki uning operatsion tizimining ishdan chiqishi;
- ma'lumotlar bazalaridagi ma'lumotlarning buzilishi;
- dasturiy ta'minot kompyuterlarini yo'q qilish;
- aloqa liniyalarining uzilishi va boshqalar.

Faol tahdidlar manbayi – buzg'unchining harakatlari, zararli dastur va hokazolar bo'lishi mumkin.

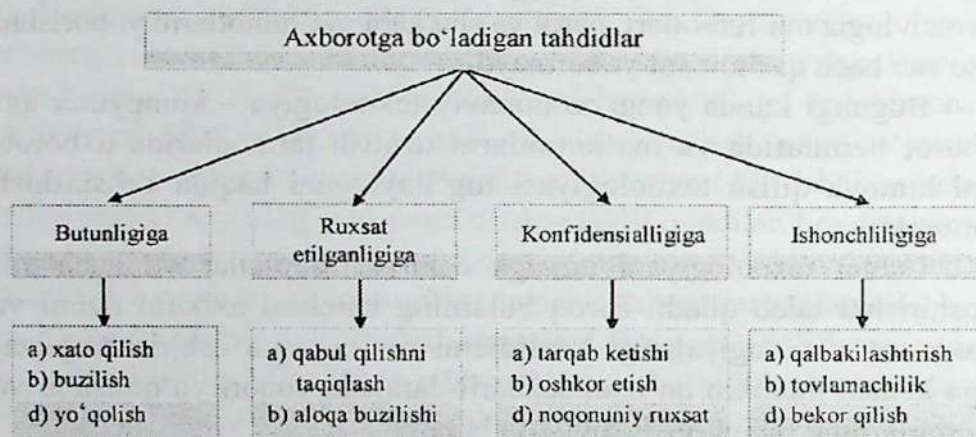
Ruxsatsiz kirish – bu himoyalangan ma'lumotlarga kirish huquqiga ega bo'lmagan shaxs tomonidan maxfiy ma'lumotlarni noqonuniy ravishda, qasddan egallab olish.

Axborot tahdidlarining xilma-xilligi:

Mantiqiy ma'lumotlar, nomidan ko'rinib turibdiki, ma'lumotni buzish yoki yo'q qilish uchun ishlatiladi, kamroq hollarda ular o'g'irlash yoki aldash uchun ishlatiladi.

Mantiqiy ma'lumotlarni manipulyatsiya qilish, odatda, tashkilotdan ketmoqchi bo'lgan norozi xodimlar tomonidan amalga oshiriladi, ammo ular maslahatchilar, muayyan siyosiy e'tiqodga ega bo'lgan xodimlar va boshqalar bo'lishi mumkin.

Axborot xizmatining buzilishi – bu axborot texnologiyasining o'zi tomonidan tahdidi tushuniladi.



9.1-rasm. Axborotga bo'ladigan tahdidlar.

9.2. Axborot xavfsizligini yaratish vositalari

Axborot xavfsizligi tizimlarini axborot tizimi va axborot texnologiyasida yaratish quyidagi prinsiplarga asoslanadi:

1) mahalliy va xorijiy himoya tizimlarini yaratish amaliyoti bilan tasdiqlangan va axborotni qayta ishlashning texnologik tizimining barcha bosqichlarida qo'llaniladigan o'zaro bog'liq bo'lgan tashkiliy, dasturiy ta'minot, fizikaviy va boshqa xususiyatlarning maqbul kombinatsiyasini anglatadigan himoya tizimini yaratishga tizimli yondashuv;

2) tizimning uzluksiz rivojlanish prinsipi. Kompyuter axborot tizimlari uchun asoslardan biri bo'lgan ushbu prinsip axborot tizimining xavfsizligida yanada dolzarbdir;

3) qayta ishlangan ma'lumotlarga kirish uchun vakolatlarni ajratish va minimallashtirish va ishlov berish tartib-taomillari, ya'ni foydalanuvchilarga ham, axborot tizimlari xodimlariga ham o'zlarining rasmiy vazifalarini bajarishlari uchun yetarli bo'lgan qat'iy belgilangan vakolatlar bilan ta'minlash;

4) ruxsatsiz kirish urinishlarini nazorat qilish va ro'yxatdan o'tkazishning to'liqligi, ya'ni har bir foydalanuvchining shaxsini aniqlash, mumkin bo'lgan tekshiruv uchun uning harakatlarini qayd etish zaruriyati va axborot texnologiyalarida har qanday ma'lumotni qayta ishlash operatsiyasini oldindan ro'yxatdan o'tkazmasdan amalga oshirish mumkin emasligi;

5) himoya tizimining ishonchliligini ta'minlash, ya'ni buzilishlar, nosozliklar, o'g'ring qasddan qilingan harakatlari yoki tizimdagi foydalanuvchilar va xizmat ko'rsatuvchi xodimlarning xatolari paytida ishonchlik darajasini pasaytirishning iloji yo'qligi;

6) himoya tizimining ishlashi ustidan nazoratni ta'minlash, ya'ni himoya mexanizmlarining ishlashini nazorat qilish vositalari va usullarini yaratish;

7) zararli dasturlarga qarshi vositalarning barcha turlarini taqdim etish.

Axborotni himoyalashning quyidagi usullari mavjud:

- axborotni muhofaza qilishning huquqiy ta'minoti – axborot xavfsizligi tizimidagi o'z faoliyati doirasida talablari majburiy bo'lgan qonunchilik hujjatlari, normativ hujjatlar, ko'rsatmalar yig'indisi;

- axborotni muhofaza qilishni tashkiliy jihatdan ta'minlash – bu shuni anglatadiki, axborot xavfsizligi muayyan tarkibiy bo'linmalar tomonidan amalga oshiriladi, masalan, kompaniyaning xavfsizlik xizmati va uning tarkibiy tuzilmalari: rejim, xavfsizlik va boshqalar;

- axborotni axborot xavfsizligi bilan ta'minlash – u axborot tizimi bazasi faoliyatini ta'minlaydigan muammolarni hal qilish asosida yotadigan ma'lumotlar, ko'rsatkichlar, parametrlarni o'z ichiga oladi;

- texnik (apparat) axborot xavfsizligi – axborotni himoya qilish uchun ham, axborot tizimi bazasi faoliyatini ta'minlash uchun ham texnik vositalardan keng foydalaniladi;

- axborot xavfsizligini ta'minlash dasturi – bu turli xil kanallarning mavjudligi va xavfliligini baholaydigan turli xil axborot, buxgalteriya hisobi, statistik va hisoblash dasturlariga va ma'lumotlarga ruxsatsiz kirish usullariga taalluqlidir;

- axborot xavfsizligini matematik qo'llab-quvvatlash – bu tajovuzkorlar ixtiyorida bo'lgan texnik vositalar, zarur himoya zonolari va standartlari xavfini baholash bilan bog'liq turli xil hisob-kitoblar uchun ishlatiladigan matematik usullar.

9.3. Axborotlarni texnik himoyalash

Axborot xavfsizligi tizimlarini yaratishning 7 ta bosqichlari quyidagi jadvalda keltirilgan:

1. Himoya obyektini tahlil qilish. Nimani himoya qilish kerakligini aniqlashdan iborat:

- himoyaga muhtoj bo'lgan ma'lumotlar aniqlanadi;
- himoyalangan ma'lumotlarning eng muhim elementlari (muhim) ajratib ko'rsatiladi;
- ma'lumotlarning ishlash muddati belgilanadi (raqibga chiqarilgan ma'lumotlarni sotish uchun zarur bo'lgan vaqt);
- himoya qilinadigan ma'lumotlarning xususiyatini aks ettiruvchi ma'lumotlarning asosiy ko'rsatkichlari aniqlanadi;
- ko'rsatkichlar korxonaning funksional yo'nalishlari (ishlab chiqarish va texnologik jarayonlar, ishlab chiqarishni moddiy-texnik ta'minlash tizimi, boshqaruv bo'linmalari) bo'yicha tasniflanadi.

2. Tahdidlarni aniqlash quyidagilarni o'z ichiga oladi:

- himoyalangan ma'lumotga kim qiziqishi mumkinligi aniqlanadi;
- ushbu ma'lumotlarni olish uchun raqobatchilar tomonidan qo'llaniladigan usullar baholanadi;
- axborotlar tarqalishining ehtimoliy kanallari taxmin qilinadi;
- raqib yoki har qanday xakerning harakatlarini bostirish uchun chora-tadbirlar tizimi ishlab chiqiladi.

3. Qabul qilingan va doimiy ishlaydigan xavfsizlik tizimlarining samaradorligi tahlil qilinadi (hujjatlarning jismoniy xavfsizligi, xodimlarning ishonchligi, maxfiy ma'lumotlarni uzatish uchun ishlatiladigan aloqa liniyalarining xavfsizligi va boshqalar).

4. Zarur himoya choralarini belgilaydi. Dastlabki uch bosqichda olib borilgan tahliliy tadqiqotlar asosida korxona xavfsizligini ta'minlash uchun zarur qo'shimcha chora-tadbirlar va vositalar ishlab chiqiladi.

5. Firma (tashkilot) rahbarlari barcha zarur xavfsizlik choralari bo'yicha takliflarni ko'rib chiqadilar va ularning narxi va samaradorli hisoblanadi.

6. Belgilangan ustuvor yo'nalishlarni hisobga olgan holda qabul qilingan qo'shimcha xavfsizlik choralarini amalga oshirishdan iborat bo'ladi.

7. Monitoring va amalga oshirilgan xavfsizlik choralarini kompaniya xodimlariga yetkazishni o'z ichiga oladi.

9.4. Ochiq va yopiq kalitlar bilan shifrlash tizimi

Shifrlash – kriptografik uslublardan (shifr matnga va dastlabki matnga o'girish, elektron raqamli imzoni shakllantirish va tekshirish, xesh-funksiya shakllantirish va tekshirish) foydalanishga asoslangan axborotni o'zgartirish jarayoni. Axborotni shifrlash uni begonalar tomonidan o'rganish yoki o'zgartirish imkoniyatini yo'qqa chiqaradi. Shuningdek, ma'lumotlarga va dasturlarga, ulardan noqonuniy foydalanish maqsadida, ruxsatsiz raqamli imzo tizimiga kirishning oldini olishni ta'minlaydi. Shifrlashning ikki usuli mavjud: simmetrik va asimmetrik. Simmetrik shifrlashda, kodlash va kodni ochish uchun birgina kalitning o'zidan foydalaniladi.

Asimmetrik shifrlashda ikkita kalitdan foydalaniladi. Ulardan biri (ochiq kalit) dastlabki matnni shifr matnga o'girishni, ikkinchisi esa (yopiq kalit) dastlabki matnga o'girishni ta'minlaydi. Samaradorlikni yanada oshirish maqsadida simmetrik va asimmetrik shifrlash algoritmlari birgalikda ishlatiladi. Bu holatda simmetrik shifrlashdan ma'lumotlarni ochiq kanallar orqali uzatishda ma'lumotlarni shifrlashda,

asimmetrik shifrlashdan esa simmetrik shifrlash algoritmlarining kalitlarini shifrlashda ishlatiladi.

CryptoAPI yordamida simmetrik shifrlash sxemasini amalga oshirish. Shifrlash algoritmlari uchun kalitlarni generatsiya qilishning usullaridan biri parollardan (kalit soʻzdan) kriptografik kalit yaratishdir. Ushbu parol oʻzgaruvchan uzunlikda boʻladi va kriptografik kalitdan uzunligi kichik boʻladi. Lekin parol kalit yaratish uchun uncha yaxshi boʻlmagan material hisoblanadi, chunki foydalanuvchilar parol sifatida esda qoluvchi lugʻatlarda mavjud soʻzlardan koʻp foydalanadi, buni esa lugʻat yordamidagi xurujlardan osonlik bilan buzish mumkin. Odatda lugʻat yordamidagi xurujlar operatsiya yordamida amalga oshiriladi, shu sababli parollardan kalit yasash uzoq vaqt davom etishi kerak, bu vaqt hisoblash mashinalari uchun uzoq sanaladi, foydalanuvchilar uchun esa bilinmaydigan vaqtdir. Agar lugʻatdagi soʻzlar soni N ta, kalitni saqlash vaqti T boʻlsa, u holda kalitdan parol yaratish uchun ketadigan vaqt t_0 quyidagi formula orqali hisoblanadi: $t_0 = 2T/N$. Agar kalitni saqlash muddati oʻrtacha $T = 7$ kun = $6,1 \cdot 10^5$ sekund, lugʻat kattaligi $N = 10^6$ ta boʻlsa, paroldan kalit yaratish vaqti $t_0 = 1,2$ sekunddan kam boʻlmasligi zarur. Agar xavfsizlikning yuqori bosqichi talab etilsa, bunday hollarda parol tasodifiy ketma-ketliklardan tashkil topishi zarur. Bunday ketma-ketliklarni kriptoproviderda CryptGenRandom funksiyasi yordamida yoki boshqa tasodifiy sonlar datchigidan olish mumkin. Parollar yordamida kalit yaratishning bir qancha shartlari mavjud:

- parol ixtiyoriy uzunlikda berilishi mumkin, kalit esa fiksirlangan uzunlikda boʻladi;
- bir-biriga oʻxshash va yaqin parollardan, bir-biriga oʻxshagan va uzoq kalitlar yaratilishi kerak;
- turli xil parollar yordamida bir xil kalitlar yaratilmasligi kerak.

Himoyalangan kanal mavjud boʻlmagan holatda kalitlar asimmetrik shifrlash algoritmi yordamida shifrlab joʻnatiladi. Buning uchun xabar joʻnatuvchida qabul qiluvchining ochiq kaliti boʻlishi kerak, ushbu ochiq kalit yordamida shifrlash kaliti asimmetrik shifrlash usulida shifrlanadi va shifrlangan axborot bilan qoʻshib ochiq

kanal orqali qabul qiluvchiga uzatiladi. Shifrlash kalitini esa tasodifiy sonlar generatori orqali yaratiladi.

Identifikatsiya va autentifikatsiya tushunchalari. Identifikatsiya (identification) – foydalanuvchini uning nomi bo'yicha aniqlash. Foydalanuvchi tizimga uning so'rovi bo'yicha o'zining identifikatori-ni bildiradi, tizim esa o'zining ma'lumotlar bazasida uning borligini tekshiradi.

Autentifikatsiya (authentication) – ma'lum qilingan foydalanuvchi, jarayon yoki qurilmaning haqiqiy ekanligini tekshirish tartibi. Bu tekshirish foydalanuvchi haqiqatan aynan o'zi ekanligiga ishonch hosil qilishga imkon beradi.

9.5. Tarmoq va kompyuter viruslari, ularning turlari

Kompyuter virusining ko'p ta'riflari mavjud. Birinchi ta'rifni 1984-yili Fred Koen bergan: Kompyuter virusi – boshqa dasturlarni, ularga o'zini yoki o'zgartirilgan nusxasini kiritish orqali, ularni modifikatsiyalash bilan zaharlovchi dastur. Bunda kiritilgan dastur keyingi ko'payish qobiliyatini saqlaydi.

Hozirgi kunda kompyuter viruslari g'arazli maqsadlarda ishlatiluvchi turli xil dasturlarni olib kelib tatbiq etishda eng samarali vositalardan biri hisoblanadi.

Kompyuter viruslarini dasturli viruslar deb atash to'g'riroq bo'ladi. Viruslar bilan zararlangan dasturlar virus tashuvchi yoki zararlangan dasturlar deyiladi.

Zararlangan disk – bu ishga tushirish sektorida virus dastur joylashib olgan diskdir.

Hozirgi paytda kompyuterlar uchun ko'pgina noqulayliklar tug'dirayotgan har xil turlardagi kompyuter viruslari keng tarqalgan. Shuning uchun ham ulardan saqlanish usullarini ishlab chiqish muhim masalalardan biri hisoblanadi. Viruslarning katta guruhini kompyuterning ish bajarish tartibini buzmaydigan, ya'ni «ta'sirchan bo'lmagan» viruslar guruhi tashkil etadi.

Troyan oti – bu asosiy harakatlarga qo'shimcha ravishda amalga oshiriladigan dastur, ya'ni hujjatlarda tasvirlanmagan harakatlarni ishlab chiqadi va hujjatlashtiradi.

Virus – bu boshqa dasturlarga zarar yetkazishi mumkin bo‘lgan dastur bo‘lib, ularga keyinchalik takrorlash imkoniyatiga ega bo‘lgan o‘zgartirilgan nusxasini qo‘shadi.

Qurt – bu tarmoq orqali tarqaladigan va o‘z nusxasini magnit muhitda qoldirmaydigan dastur. Virus yuqtirishi mumkin bo‘lgan tizimni aniqlash uchun qurt tarmoqni qo‘llab-quvvatlash mexanizmlaridan foydalanadi. Keyin, xuddi shu mexanizmlardan foydalangan holda, u tanasini yoki uning bir qismini ushbu tugunga o‘tkazadi yoki faollashadi yoki buning uchun mos sharoitlarni kutadi.

Viruslarning boshqa guruhiga kompyuterning ish tartibini buzuvchi viruslar kiradi. Bu viruslarni quyidagi turlarga bo‘lish mumkin: xavfsiz viruslar (fayllar tarkibini buzmaydigan), xavfli viruslar (fayllar tarkibini buzuvchi) hamda juda xavfli viruslar (kompyuter qurilmalarini buzuvchi va operator sog‘lig‘iga ta‘sir etuvchi). Bu kabi viruslar odatda professional dasturchilar tomonidan tuziladi.

Kompyuter virusi – bu maxsus yozilgan dastur bo‘lib, boshqa dasturlar tarkibiga yoziladi, ya‘ni zararlaydi va kompyuterlarda o‘zining g‘arazli maqsadlarini amalga oshiradi. Kompyuter virusi orqali zararlanish oqibatida kompyuterlarda quyidagi o‘zgarishlar paydo bo‘ladi:

- ayrim dasturlar ishlamaydi yoki xato ishlay boshlaydi;
- bajariluvchi faylning hajmi va uning yaratilgan vaqti o‘zgaradi;
- ekranda anglab bo‘lmaydigan belgilar, turli xil tasvir va tovushlar paydo bo‘ladi;
- kompyuterning ishlashi sekinlashadi va tezkor xotiradagi bo‘sh joy hajmi kamayadi;
- disk yoki diskdagi bir necha fayllar zararlanadi (ba‘zi hollarda disk va fayllarni tiklab bo‘lmaydi);
- vinchester orqali kompyuterning ishga tushishi yo‘qoladi.

Viruslar asosan disklarning yuklanuvchi sektorlarini va exe, com, sys va bat kengaytmali fayllarni zararlaydi. Hozirgi kunda bular qatoriga ofis dasturlarini o‘rnatuvchi fayllarni ham kiritish mumkin. Oddiy matnli fayllarni zararlaydigan viruslar kamdan kam uchraydi.

Kompyuterning viruslar bilan zararlanish yoʻllari quyidagilardir:

- USB flesh orqali;
- kompyuter tarmoqlari orqali.

Shuni aytib oʻtish lozimki, hozirgi paytda har xil turdagi axborot va dasturlarni oʻgʻirlab olish niyatida kompyuter viruslaridan foydalanish eng samarali usullardan biri hisoblanadi.

Dasturli viruslar kompyuter tizimlarining xavfsizligiga tahdid solishning eng samarali vositalaridan biridir. Shuning uchun ham dasturli viruslarning imkoniyatlarini tahlil qilish masalasi hamda bu viruslarga qarshi kurashish hozirgi paytning dolzarb masalalaridan biri boʻlib qoldi.

Viruslardan tashqari fayllar tarkibini buzuvchi «troyan» dasturlari mavjud. Virus koʻpincha kompyuterga sezdirmasdan kiradi. Foydalanuvchining oʻzi «troyan» dasturini foydali dastur sifatida diskka yozadi. Maʼlum bir vaqt oʻtgandan keyin dastur oʻz taʼsirini koʻrsata boshlaydi.

Oʻz-oʻzidan paydo boʻladigan viruslar mavjud emas. Virus dasturlari inson tomonidan kompyuterning dasturiy taʼminotini, uning qurilmalarini zararlash va boshqa maqsadlar uchun yoziladi. Viruslarning hajmi bir necha baytdan to oʻnlab kilobaytgacha boʻlishi mumkin.

Troyan dasturlari foydalanuvchiga zarar keltiruvchi boʻlib, ular buyruqlar (modullar) ketma-ketligidan tashkil topgan, omma orasida juda keng tarqalgan dasturlar (tahrirlovchilar, oʻyinlar, translatorlar) ichiga oʻrnatilgan boʻlib, bir qancha amallar bajarilishi bilan ishga tushadigan «mantiqiy bomba» deb ataladigan dasturdir. Oʻz navbatida, «mantiqiy bomba»ning turli koʻrinishlaridan biri «soat mexanizimli bomba» hisoblanadi.

Shuni taʼkidlab oʻtish kerakki, «troyan» dasturlari oʻz-oʻzidan koʻpaymasdan, kompyuter tizimi boʻyicha dasturlovchilar tomonidan tarqatiladi. Troyan dasturlardan viruslarning farqi shundaki, viruslar kompyuter tizimlari boʻylab tarqatilganda, ular mustaqil ravishda hosil boʻlib, oʻz ish faoliyatida dasturlarga oʻz matnlarini yozgan holda ularga zarar koʻrsatadi.

Zararlangan dasturda dastur bajarilmasdan oldin virus o'zining buyruqlari bajarilishiga imkoniyat yaratib beradi. Shuning uchun ham virus dasturning bosh qismida joylashadi yoki dasturning birinchi buyrug'i unga yozilgan virus dasturiga shartsiz o'tish bo'lib xizmat qiladi. Ishga tushgan virus boshqa dasturlarni zararlaydi va shundan so'ng virus tashuvchi dasturga ishini topshiradi.

Virus hayoti odatda quyidagi davrlarni o'z ichiga oladi: qo'llanilish, inkubatsiya, replikatsiya (o'z-o'zidan ko'payish) va hosil bo'lish. Inkubatsiya davrida virus passiv bo'lib, uni izlab topish va yo'qotish qiyin. Hosil bo'lish davrida u o'z funksiyasini bajaradi va qo'yilgan maqsadiga erishadi.

Tarkibi jihatidan virus juda oddiy bo'lib, bosh qism va ba'zi holatlarda dumdan iborat. Virusning bosh qismi deb boshqarilishni birinchi bo'lib ta'minlovchi imkoniyatga ega bo'lgan dasturga aytiladi. Virusning dum qismi zararlangan dasturda bo'lib, u bosh qismidan alohida joyda joylashadi.

Kompyuter viruslari xarakterlariga nisbatan norezident, rezident, butli, gibriddli va paketli viruslarga ajratiladi.

Faylli norezident viruslar to'liqligicha bajarilayotgan faylda joylashadi, shuning uchun ham u faqat virus tashuvchi dastur faollashgandan so'ng ishga tushadi va bajarilgandan so'ng tezkor xotirada saqlanmaydi.

Rezident virus norezident virusdan farqliroq tezkor xotirada saqlanadi.

Rezident viruslarning yana bir ko'rinishi but viruslar bo'lib, bu virusning vazifasi vinchester va egiluvchan magnitli disklarning yuklovchi sektorini ishdan chiqarishdan iborat. But viruslarning boshi diskning yuklovchi but sektorida va dumi disklarning ixtiyoriy boshqa sektorlarida joylashgan bo'ladi.

Paketli virusning bosh qismi paketli faylda joylashgan bo'lib, u operatsion tizim topshiriqlaridan iborat.

Gibridli viruslarning boshi paketli faylda joylashadi. Bu virus ham faylli, ham sektorli bo'ladi. Tarmoq viruslar kompyuter tarmoqlarida tarqalishga moslashtirilgan, ya'ni tarmoqli viruslar deb axborot almashishda tarqaladigan viruslarga aytiladi.



9.2-rasm. Viruslarning turlari.

Asoslangan algoritmlar bo'yicha dasturli viruslarni quyidagicha tasniflash mumkin:

- parazitli virus – fayllarning tarkibini va diskning sektorini o'zgartiruvchi virus. Bu virus oddiy viruslar turkumidan bo'lib, osonlik bilan aniqlanadi va o'chirib tashlanadi;

- replikatorli virus – «chuvalchang» deb nomlanadi. U kompyuter tarmoqlari bo'yicha tarqalib, kompyuterlarning tarmoqdagi manzilini aniqlaydi va u yerda o'zining nusxasini qoldiradi;

- ko'rinmas virus – stels-virus deb nom olib, zararlangan fayllarga va sektorlarga operatsion tizim tomonidan murojaat qilinsa, avtomatik ravishda zararlangan qismlar o'rniga diskning toza qismini taqdim etadi. Natijada ushbu viruslarni aniqlash va tozalash juda katta qiyinchiliklarga olib keladi;

- mutant virus – shifrlash va deshifrlash algoritmlaridan iborat bo'lib, natijada virus nusxalari umuman bir-biriga o'xshamaydi.

Ushbu viruslarni aniqlash juda qiyin muammo. Shu bois ham ular o'zlarida mukammallashtirilgan algoritmlarni to'siqsiz bajarib, qo'yilgan maqsadlariga erishishlari mumkin.

Virus bilan zararlanganlik belgilari. Garchi ko'pchilik zararli dasturlar aniq iz qoldirmasa ham va sizning kompyuteringiz yaxshi ishlayotgan bo'lsa-da, ba'zida siz hali ham mavjud bo'lgan virus belgilarini ko'rishingiz mumkin:

1) kompyuterning ishlashi pasayadi, ya'ni jarayonlar sekin bajariladi, dastur oynalarini yuklash uchun ko'proq vaqt ketadi va ba'zi tasodifiy dasturlar fonda ishlaydi;

2) brauzeringizdagi bosh sahifalar o'zgaradi yoki harakatlanuvchi reklamalar odatdagidan ko'ra tez-tez paydo bo'ladi, bu ham ogohlantirish belgisi hisoblanadi;

3) zararli dastur hatto kompyuterning asosiy funksiyalariga ham ta'sir qilishi mumkin: Windows ochilmaydi, internetga ulanib bo'lmaydi yoki tizimning yuqori darajadagi boshqaruv funksiyalariga kirish imkoni yo'q bo'ladi.

Agar kompyuterga zarar yetkazilganligiga shubha bo'lsa, darhol tizimni tekshirish kerak bo'ladi. Agar virus topilmasa, lekin virus mavjudligida hali ham shubha bo'lsa, kompyuterdagi muqobil antivirus skanerini ishga tushirish kerak.

Nazorat savollari

1. Axborot himoyasi qanday bosqichlarga bo'linadi?
2. Gibriddli viruslar deb qanday viruslarga aytiladi?
3. Antivirus dasturlarining vazifalari.
4. Axborot xavfsizligini oshirishning usul va vositalari.
5. Login va parolga ega bo'lish bosqichlari qanday olib boriladi?

10-BOB. DASTURLASH TEXNOLOGIYALARI. OBYEKTGA YO'NALTIRILGAN DASTURLASH TILLARI

10.1. Dasturlash asoslari va uning imkoniyatlari

Dasturlash – kompyuterlar va boshqa mikroprotessorli elektron qurilmalar uchun dasturlar tuzish, sinash va o'zgartirish jarayonidan iborat.

Dasturlash elektron mashinalarda masalalarni yechish hamda ularda har xil aqliy mehnat turlarini bajarish nazariyasi va usullarini ishlab chiqish bilan shug'ullanadigan fan, algoritmlar nazariyasining amaliy bo'limi, insonning mashina bilan aloqa qilish vositasi hisoblanadi.

Dasturlashning asosiy vazifalaridan biri elektron mashinalar uchun programma (dastur) tuzish usullari, ularni tekshirish va takomillashtirishdan iborat. Yechilishi lozim bo'lgan masala algoritmi dasturlashda «mashina tili»ga o'tkaziladi.

Dasturlash bevosita dasturlash va avtomatik dasturlashga bo'linadi:

- bevosita dasturlashda dasturning umumiy sxemasini ishlab chiqishdan kodlash va mashinaga kiritishgacha bo'lgan barcha ishni dasturchi bajaradi;

- avtomatik dasturlashda esa dasturchi faqat dastur sxemasini tuzib, uni qisqartirilgan simvolik ko'rinishda yozadi, dastur tuzish va uni kodlash kabi texnikaviy ishlarni esa mashinaning o'zi maxsus dasturlash tizimi yordamida bajaradi.

Dasturlash dastur tuzuvchilarga dasturlarni kommutatsiya bloki orqali kompyuterning asosiy xotirasiga to'g'ridan to'g'ri kiritish imkonini berdi. Dasturlar mashina kodida ikkilik sanoq tizimida yozilgan. Dasturlarni mashina tilida yozishda tez-tez xatolarga yo'l qo'yilar edi. Bundan tashqari, mashina kodlaridagi dastur tushunish

uchun g'oyat murakkab hisoblangan. Vaqt o'tishi bilan kompyuterlar tobora kengroq qo'llanila boshlandi hamda yuqoriroq darajadagi protsedura tillari paydo bo'ldi. Bularning dastlabkisi FORTRAN tili bo'lgan. Biroq obyektga mo'ljallangan yondashuv rivojiga asosiy ta'sir keyinroq paydo bo'lgan, masalan, ALGOL kabi protsedura tillari. Protседura tillari dasturchiga axborotga ishlov berish dasturini pastroq darajadagi bir nechta protseduraga bo'lib tashlash imkonini beradi. Pastroq darajadagi bunday protseduralar dasturning umumiy tuzilmasini belgilab beradi. Ushbu protseduralarga izchil murojaatlar protseduralardan tashkil topgan dasturlarning bajarilishini boshqaradi. Dasturlashning bu yangi paradigmasi mashina tilida dasturlash paradigmasiga nisbatan ancha ilg'or bo'lib, unga tuzilmalashtirishning asosiy vositasi bo'lgan protseduralar yig'indisidan iborat bo'lgan. Har bir protsedura ma'lumotlarga kirish usullarini dasturlashi lozim bo'lganligi tufayli, ma'lumotlar taqdimotining o'zgarishi dasturning ushbu kirish amalga oshirilayotgan barcha o'rinlarining o'zgarishiga olib keladi. Shunday qilib, hatto eng kichik to'g'rilash ham butun dasturda qator o'zgarishlar sodir bo'lishiga olib kelgan.

Modulli dasturlashda, masalan, Modula2 kabi tilda protsedurali dasturlashda topilgan ayrim kamchiliklarni bartaraf etishga urinib ko'rildi. Modulli dasturlash dasturni bir necha tarkibiy bo'laklarga, yoki, boshqacha qilib aytganda, modullarga bo'lib tashlaydi. Agar protsedurali dasturlash ma'lumotlar va protseduralarni bo'lib tashlasa, modulli dasturlash, undan farqli o'laroq, ularni birlashtiradi. Modul ma'lumotlarning o'zidan hamda ma'lumotlarga ishlov beradigan protseduralardan iborat. Dasturning boshqa qismlariga moduldan foydalanish kerak bo'lib qolsa, ular modul interfeysiga murojaat etib qo'ya qoladi. Modullar barcha ichki axborotni dasturning boshqa qismlarida yashiradi.

Biroq modulli dasturlash ham kamchiliklardan xoli emas. Modullar kengaymas bo'ladi, bu degani kodga bevosita kirishsiz hamda uni to'g'ridan to'g'ri o'zgartirmay turib modulni qadamma-qadam o'zgartirish mumkin emas. Bundan tashqari, bitta modulni ishlab chiqishda, uning funksiyalarini boshqasiga o'tkazmay turib bosh-

qasidan foydalanib bo'lmaydi, garchi modulda turni belgilab bo'lsada, bir modul boshqasida belgilangan turdan foydalana olmaydi.

Modulli dasturlash – bu yana protseduraga mo'ljallangan gibridli sxema bo'lib, unga amal qilishda dastur bir necha protseduralarga bo'linadi. Biroq endilikda protseduralar ishlov berilmagan ma'lumotlar ustida amallarni bajarmaydi, balki modullarni boshqaradi. Obyektga yo'naltirilgan dasturlash texnologiyalari – dasturiy ta'minotning inqiroziga javob sifatida yuzaga kelgan dasturlash texnologiyalari hisoblanadi. Bu inqirozning sababi shunda ediki, strukturali dasturlash metodlari murakkablik darajasi borgan sari ortib borayotgan masalalar uchun dasturiy ta'minot yaratish imkonini bera olmay qoldi. Buning natijasida turli loyihalarni bajarish rejalari buzildi, qilinayotgan xarajatlar belgilangan budjetdan ortib ketdi, dasturiy ta'minotning funksionalligi buzildi, xatoliklari ortdi. Dasturiy ta'minotning eng muhim tomonlaridan biri – uning murakkablik darajasidir. Biror dasturchi tizimning barcha xususiyatlarini to'liq hisobga ola olmaydi. Shuning uchun uni ishlab chiqishda dasturchi va boshqa mutaxassislarning yirik jamoasi qatnashadi. Demak, qo'yilgan masalaga to'g'ridan to'g'ri bog'liq bo'lgan murakkabliklarga ana shu jamoaning ishini bir maqsadga qaratilgan boshqarish ham qo'shiladi. An'anaviy dasturlash tillarida bunday murakkabliklarni hal qilishda «ajrat va boshqar» prinsipidan foydalanilgan. Ya'ni masala kichik-kichik masalalarga ajratilib, keyin har bir masala uchun alohida dastur ishlab chiqilgan va birlashtirilgan.

Obyektga yo'naltirilgan dasturlash texnologiyalari esa masalaga boshqacha usulda yondashadi. Unda masalalarning yechish uchun kerak bo'ladigan elementlarni muammoli sohaning turli abstraksiyalariga taalluqli ekanligi asosiy o'rinda turadi. Bu abstraksiyalarni dasturchilar ishlab chiqishgan. Dasturchilar tomonidan ma'lum bir soha o'rganilib, uning alohida obyektlari ajratib olingan. Bu obyektlar uchun masalalarni yechishda qo'llash mumkin bo'lgan xususiyatlar aniqlangan. Ehtiyojga qarab har bir xususiyat ustida bajarish mumkin bo'lgan amallar aniqlangan. So'ngra o'rganilayotgan sohaning har bir real obyektiga mos dasturiy obyekt ishlab chiqilgan. Ma'lumki,

kompyuter yordamida hal qilinadigan har bir masala uchun maxsus dastur ishlab chiqish talab qilinadi. Bunday masalalar sinfining kengayib borishi albatta yangi-yangi dasturlar yaratishga olib keladi. Yangi dasturlarni yaratish uchun «eski» dasturlash tillarining imkoniyati yetmay qolganda yoki dastur yaratish jarayonini mukammallashtirish uchun yangi dasturlash tiliga ehtiyoj paydo bo'ladi.

Obyektga yo'naltirilgan dasturlash – bu dasturlashning shunday yangi yo'nalishiki, dasturiy sistema o'zaro aloqada bo'lgan obyektlar majmuasi sifatida qaraladi va har bir obyektga ma'lum bir klassga mansub hamda har bir klass qandaydir shajarani hosil qiladi deb hisoblanadi. Obyektga yo'naltirilgan dasturlash dasturni obyektlar to'plami sifatida ifodalashga asoslangan dasturlash metodologiyasi, ularning har biri ma'lum bir klassning namunasidir va sinflar meros iyerarxiyasini hosil qiladi.

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 sistemalarning ishonchliligini ta'minlaydi.

Obyektga yo'naltirilgan dasturlashning eng asosiy tushunchasi obyekt va klass hisoblanadi. Obyektga yo'naltirilgan dasturlash uzoq yillar davomida an'anaviy, ya'ni standart hisoblangan dasturlashga nisbatan xos bo'lgan tasavvurlarni bir chetga qo'yishni talab qiladi. Natijada obyektga yo'naltirilgan dasturlash juda ham sodda, ko'rgazmaliligi yuqori bo'lib, dasturiy ta'minot yaratishdagi ko'plab muammolarni hal qilishning juda ajoyib vositasiga aylanadi.

10.2. Dasturlash tillari va ularning turlari

Hozirgi kunda ko'plab masalalarni kompyuter yordamida yechish uchun mo'ljallangan zamonaviy dasturlash tillari mavjud. GitHub xalqaro saytining dunyoning axborot texnologiyalari sohasida yuqori o'rinlarda turadigan kompaniya dasturchilari o'rtasida o'tkazilgan so'rovnomada eng mashhur dasturlash tillari reytingida e'lon qilingan

ro'yxatdan foydalanib, ba'zi bir zamonaviy dasturlash tillarini ko'rib chiqamiz:

JavaScript – multiparadigmali dasturlash tili hisoblanib, obyektga yo'naltirilgan, funksional uslublarni qo'llab-quvvatlaydi. JavaScript – yuqori darajali dasturlash tili bo'lib, ishlash jarayonida mijoz brauzerida, ya'ni oxirgi foydalanuvchining kompyuterida buyruqlarni qayta ishlaydi, bu esa serverdagi yukni kamaytiradi va dastur ishlash tezligini oshiradi. JavaScript Netscape kompaniyasi tomonidan ishlab chiqilgan. Bu dasturlash tili internet texnologiyalarida keng qo'llaniladi.

Java – chuqur darajada klasslarga asoslangan (class-based), obyektga yo'naltirilgan (object-oriented) dasturlash tili bo'lib, imkoni boricha oson ko'chirib yurishga va ko'plab platformalarda ishlashga moslashtirilgan. Java dasturlash tili Sun Microsystems kompaniyasi tomonidan ishlab chiqilgan. Keyinchalik bu kompaniyani Oracle kompaniyasi tomonidan sotib olgan. 1995-yili birinchi rasmiy ko'rinishi chiqqan. Java ilovalari odatda maxsus bayt kodlarda amalga oshirilganligi sababli, ular virtual Java mashinasi yordamida har qanday kompyuter arxitekturasida ishlashlari mumkin. Java dasturlash tili Jeyms Gosling tomonidan yaratilgan bo'lib, boshlang'ich davrda elektron qurilmalarni dasturlash uchun ishlab chiqilgan. Keyinchalik mijoz va server dasturlarini yozish uchun foydalanilgan. Java dasturlash tili veb sohasida uzoq vaqtdan beri foydalanib kelinayotgan, barqaror va kuchli tillardan biri hisoblanadi. Javadan barcha platformalar, operatsion tizimlar va qurilmalarda keng foydalaniladi.

Python – yuqori darajadagi dasturlash tili bo'lib, soddaligi, o'qilishi va sintaksisi tufayli eng oson til hisoblanadi. 80-yillarda Gvido van Rossum tomonidan ishlab chiqilgan. Python odatda dasturchilarni qisqa vaqtda ko'p miqdorda oson o'qiladigan va funksional kodlarni yozishini ta'minlaydigan skript til sifatida foydalaniladi, biroq u dinamik hamda obyektga yo'naltirilgan, protsedurali va funksional dasturlashni ham ta'minlaydi. Tez moslashuvchanligi hisobiga, Python bugungi kunda yuqori darajada keng qo'llaniladigan dasturlash tillaridan biri hisoblanadi. Veb ilovalarni yaratishda keng foydalaniladi.

Ruby – dinamik, ochiq kodli (open source), obyektga yo'naltirilgan dasturlash tili bo'lib, 90-yillarda kompyuter olimi Yukixiro Matsumoto tomonidan ishlab chiqilgan eng yosh tillardan biri hisoblanadi. Bu tilda o'qish va yozish uchun sodda sintaksisdan foydalanilgan, juda ko'p buyruqlarni o'rganib chiqish shart emas. Shuning hisobiga, bu tilni o'rganish nisbatan oson. Tilning o'zi obyektga yo'naltirilgan bo'lsa-da, protsedurali, funksional va imperativ dasturlashni ta'minlashi bu tilni juda ham tez moslashuvchi tillardan qiladi. Veb ilovalarni yaratishda keng foydalaniladi.

PHP (Hypertext Preprocessor – «PHP: gipermatnli protsessor») dinamik veb-saytlarni yaratish va rivojlantirish uchun eng keng tarqalgan tillardan biridir. PHP 1995-yilda ishlab chiqilgan. PHP ma'lumotlarni serverda qayta ishlaydi va natijada foydalanuvchi kerakli ma'lumotlarni HTML shaklida oladi. PHP – ochiq kodli dasturlash tili, shuning uchun minglab modullarning tayyor andozalari yozib qo'yilgan, ular kerakli funksiyalarga moslashtirilishi mumkin. U juda ko'p mavjud ma'lumotlar bazasini boshqarish tizimlari bilan birgalikda o'zaro ishlash imkonini beradi (MySQL, MySQLi, SQLite, PostgreSQL, Oracle (OCI8), Oracle, Microsoft SQL Server, Sybase, ODBC, mSQL, IBM DB2, Cloudscape va Apache Derby, Informix, Ovrinos SQL, Lotus Notes, DB++, DBM, dBase, DBX, FrontBase, FilePro, Ingres II, SESAM, Firebird/InterBase, Paradox File Access, MaxDB).

CSS (Cascading Style Sheets) – format tili bo'lib, CSSning to'liq shakli 1997-yil tashkil topgan. Dastlab, CSS kodlarini Netscape Navigator 4.0 va Internet Explorer 4.0 brauzerlari tushuna olgan, hozirgi kunda CSS kodlarni istalgan brauzerlar qo'llab-quvvatlay oladi. CSS elementlarning ekranda, qog'ozda va yoki boshqa medialarda ko'rinishini tasvirlaydi. CSS veb-sahifalarga har xil stillar berish uchun ishlatiladi hamda bir vaqtning o'zida bir nechta veb-sahifalarning dizaynini o'zgartirish mumkin. Biror-bir veb-sahifaning turli xil qurilmalarda turli xil ko'rinishini ham ta'minlaydi. CSS file.css formati orqali tashqi xotira saqlab qo'yish va kezi kelganda veb-sahifaning HTML kodini o'zgartirmasdan, faqat CSS faylni o'zgartirish orqali

veb-sahifaga yangidan ko'rinish berish mumkin. Qisqacha qilib aytadigan bo'lsak, HTML faqat veb-sahifaning kodlarini yozish uchun ishlatiladi, HTMLda ham veb-sahifaga dizayn bersa bo'ladi, lekin bu juda uzoq jarayon talab qilgani bois, CSS bu vazifani o'z bo'yniga olgan. CSSning boshqa tillardan bitta ustun tarafi shundan iboratki, CSS kodlarini kompilatsiya qilish uchun, hech qanday kompilatorlar, muharrirlar yoki biror-bir dasturlar shart emas. Bizga oddiygina Internet Explorer va shu kabi brauzerlar kifoya. Brauzerlar orqali biz CSS kodlarini bimalol tahlil qilishimiz, natijamizni to'g'ridan to'g'ri ko'rishimiz mumkin.

C++ – kompilatsiyalanadigan, statik usulda kiritilgan umumiy maqsadli dasturlash tilidir. 1979-yili Bell Labsda Byarn Stroustrup tomonidan C dasturlash tilining imkoniyatlarini kengaytirish va OOP (Object-oriented Programming) xususiyatini kiritish maqsadida ishlab chiqarilgan. Boshida «C with Classes» deb atalgan, 1983-yili hozirgi nom bilan, ya'ni C++ deb o'zgartirilgan. C++ tili operatsion tizimlarga aloqador qismlarni, mijoz-server dasturlarni, kompyuter o'yinlarini, kundalik ehtiyojda qo'llaniladigan dasturlarni va shu kabi turli maqsadlarda ishlatiladigan dasturlarni ishlab chiqarishda qo'llaniladi. C++ tili erkin va tijorat maqsadidagi turli platformalar uchun juda ko'p qo'llanishi mumkin. Masalan, x86 platformasida bu GCC, Visual C++, Intel C++ kompilatori, Embarcadero (Borland) C++ Builder va boshqalar.

Kompyuterda dasturlash oxirgi yillarda juda tez rivojlanib dastur tuzishga qiziquvchilar soni oshib bormoqda. Dasturlashtirish vositalarining zamonaviy texnologiyalari yuqori darajali ilovalarni yaratish imkoniyatlarini amalga oshirish uchun qaratilgan.

C++ dasturlash vositasining yaratilishi esa nafaqat professional dasturchilar, balki oddiy dastur tuzuvchilar uchun ham keng yo'l ochib berdi. Juda qisqa vaqt ichida Borland kompaniyasi C++ning bir qator lahjalarini ishlab chiqdi. C++ ning oxirgi lahjalarida ma'lumotlar bazasini yaratish va qayta ishlash, Internet tarmog'idan foydalangan holda ma'lumotlar alamashinuvini o'rnatish, dasturlashning obyektga yo'naltirilgan modelini keng qo'llash, vizual dasturlashda

yangi komponentlar kutubxonasini (VCL) yaratish kabi asosiy farqli imkoniyatlarni o'z ichiga oladi.

Dasturlashda qulay bo'lgan Borland C++ Builder 6 platformasini to'liqroq ko'rib chiqamiz. Borland C++ Builder 6 – Windows operatsion tizimida dastur yaratishga obyektga yo'naltirilgan dasturlash muhitidir. Borland C++ Builder 6 dasturlash muhitida dastur tuzish zamonaviy vizual loyihalash texnologiyalariga asoslangan bo'lib, unda dasturlashning obyektga yo'naltirilgan g'oyasi mujassamlashgan. Borland C++ Builder 6 – bir necha muhim ahamiyatga ega bo'lgan texnologiyalar kombinatsiyasini o'zida mujassam etgan:

- yuqori darajadagi mashinali kodda tuzilgan kompilyator;
- obyektga yo'naltirilgan komponentlar modellari;
- dastur ilovalarini vizual tuzish;
- ma'lumotlar bazasini tuzish uchun yuqori masshtabli vosita.

Borland C++ Builder 6 – Windows muhitida ishlaydigan dastur tuzish uchun qulay bo'lgan vosita bo'lib, kompyuterda dastur yaratish ishlarini avtomatlashtiradi, xatoliklarni kamaytiradi va dastur tuzuvchi mehnatini yengillashtiradi. Borland C++ Builder 6 da dastur zamonaviy vizual loyihalash texnologiyasi asosida obyektga yo'naltirilgan dasturlash nazariyasini hisobga olgan holda tuziladi. Ma'lumki, dastur tuzish sermashaqqat jarayon, lekin Borland C++ Builder 6 tizimi bu ishni sezilarli darajada soddalashtiradi va masala turiga qarab dastur tuzuvchi ishining 50–80% ini tizimga yuklaydi.

Borland C++ Builder 6 tizimi dasturni loyihalash va yaratish vaqtini kamaytiradi hamda Windows muhitida ishlovchi dastur ilovalarini tuzish jarayonini osonlashtiradi. Borland C++ Builder 6 o'zida bir qancha zamonaviy ma'lumotlar bazasini boshqarish tizimlari, dasturlash texnologiyalarini ham ma'lumotlar bazasini yaratishda ishlatadi. Komponentlarning obyektga yo'naltirilgan modeli tayyor obyektlardan foydalanib yangi ilovalar yaratish, shu bilan birga foydalanuvchining shaxsiy obyektlarini yaratish imkonini beradi. Borland C++ Builder 6 ning standart obyektlari 270 tadan ortiq asosiy sinflarni birlashtiradi. Borland C++ Builder 6 sinflari murakkab iyerarxik strukturaga ega bo'lgan vizual komponentlar kutubxonasini (Visual Com-

ponent Library – VCL) tashkil qiladi. VCL tarkibiga kiruvchi yuzlab sinflar mavjud.

Vizual dasturlash texnologiyasida obyekt deganda muloqot oynasi va boshqarish elementlari (kiritish va chiqarish maydoni, buyruq tugmalari, pereklyuchatellar va boshqa) tushuniladi.

Vizual loyihalash jarayonida dasturda yaratilayotgan ilovaning dizayni shakllanadi. Dasturchi vizual loyihalash jarayonini bajarganda Borland C++ Builder 6 avtomatik ravishda dastur kodini yaratishni boshlaydi. Dasturchi loyihagini ishlashi mobaynida dastur kodini C++ tilining maxsus operatorlari bilan to'ldiradi.

10.3. Obyektga yo'naltirilgan Borland C++ Builder 6 tizimi

Borland C++ Builder 6 dasturida dasturlash ikkita o'zaro ta'sir etuvchi bir-biri bilan bog'liq jarayon asosida tashkil qilinadi:

- dasturni vizual loyihalash jarayoni;
- dastur kodlarini konsol muhitida kiritish jarayoni.

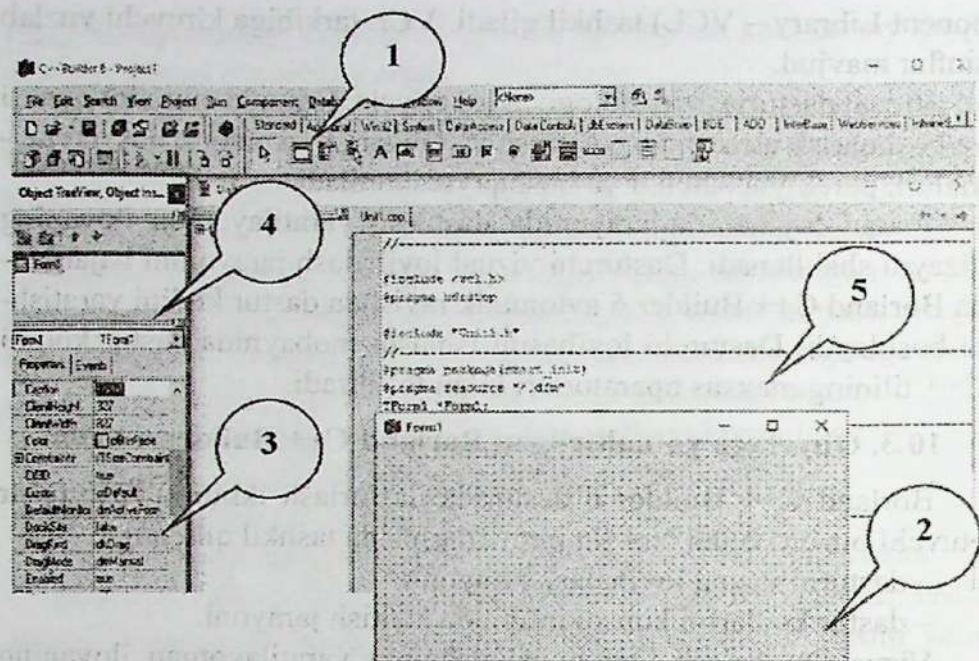
Vizual loyihalash jarayonida dasturda yaratilayotgan ilovaning dizayni shakllanadi.

Dasturchi vizual loyihalash jarayonini bajarganda Borland C++ Builder 6 avtomatik ravishda dastur kodini yaratishni boshlaydi. Dasturchi loyihagini ishlashi mobaynida dastur kodini C++ tilining maxsus operatorlari bilan to'ldiradi. Borland C++ Builder 6 dastur ham Windows amaliyot tizimining boshqa dasturlari kabi ishga tushiriladi:

Пуск => Все программы => Borland C++ Builder 6 => C++ Builder 6

Dastur yuklangandan keyin ekranda (10.1-rasm) darcha paydo bo'ladi. Borland C++ Builder 6 dasturi quyidagi beshta asosiy darchani o'z ichiga oladi:

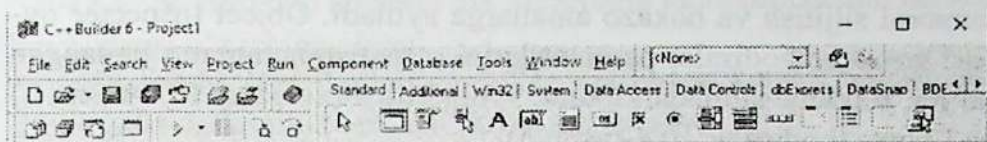
- 1) bosh oyna – C++ Builder 6 (Project1);
- 2) forma oynasi (Form1);
- 3) obyekt xossalarini tahrirlash oynasi (Object Inspector);
- 4) obyektlar ro'yxatini ko'rish oynasi (Object tree View);
- 5) dastur kodlarini kiritish oynasi (Unit.cpp).



10.1-rasm. Borland C++ Builder 6 ning interfeysi.

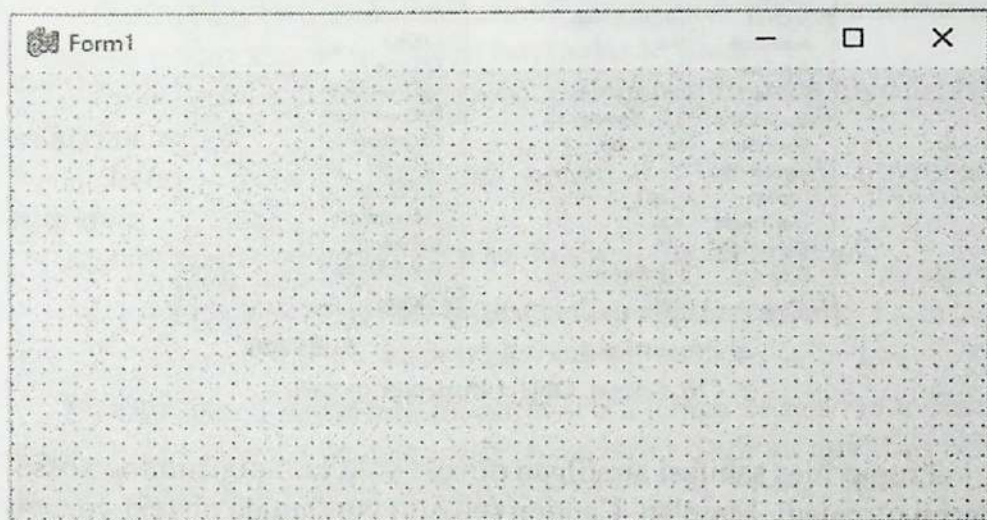
Bosh oyna (Project1) ekranning yuqori qismida joylashgan bo'lib, uning birinchi qatorida sarlavha, ya'ni loyihaning nomi (C++ Builder 6 – Project1) joylashgan. Ikkinchi qatorda asosiy menyular qatori gorizontall ko'rinishda joylashgan. Asosiy menyular qatori dasturni yaratish uchun kerak bo'ladigan barcha buyruqlar va funksiyalarga murojaat qilish imkonini yaratadi.

Keyingi qatorning chap tarafida tezkor murojaat etish tugmalari mavjud. Bajaradigan vazifasiga qarab ketma-ketlikda birlashtirilgan. Ular tez-tez ishlatiladigan buyruqlarga tezkor murojaat etish imkonini beradi. O'ng tarafida vizual komponentlar palitrasi VCL (Visual Component Library – vizual komponentlar kutubxonasi) keltirilgan. Windows operations tizimi ilovalarni yaratish uchun vizual komponentlarni o'z ichiga oladi. Vizual komponentlar palitrasi bir nechta qismlardan iborat guruhlariga bo'lingan. Bu vizual komponentlar palitrasi yordamida tezkor va oson usulda dasturlarni yaratish mumkin.



10.2-rasm. Bosh oyna.

Forma oynasi (Form1) yangi yaratilishi kerak bo'ladigan dasturning ko'rinishi hisoblanadi. Forma oynasi C++ Builder 6 ilovalari uchun asos bo'lib, unda yaratilayotgan dasturga komponentlarni joylashtirish mumkin. U dasturning sarlavhasidan boshlanadi.



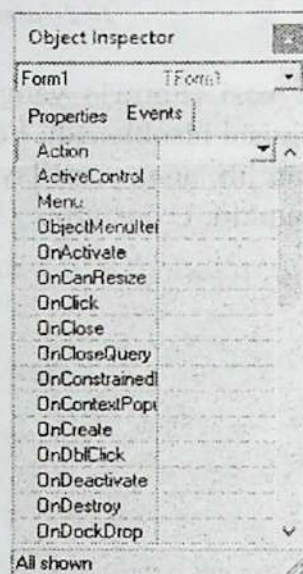
10.3-rasm. Forma oynasi.

Object Inspector oynasi (Object Inspector) obyekt xossalari va hodisalarini tahrirlash uchun xizmat qiladi. Obyektga yo'naltirilgan dasturlashda dastur bu obyektlar tizimi bo'lib, har bir obyekt bir qator xossalarga ega bo'lishi mumkin. Xossa esa ma'lumotlar va ularni boshqarish usullaridan iborat. Obyekt xossalari – bu obyektga berilgan xarakteristika bo'lib, uning ko'rinishi, joylashishi va holatidir. Bundan tashqari, obyekt turli hodisalarni ham o'rnatishi mumkin. Hodisa deb bajarish, boshqarish usuliga aytiladi, masalan, sichqonni bosish,

kursorni siljitish va hokazo amallarga aytiladi. Object Inspector oynasi xossa va hodisalar parametrlarini o'rnatish uchun mo'ljallangan bo'lib, u ikkita sahifadan iborat: Propierities (xossalar 10.4-rasm, a) va Events (hodisalar 10.4-rasm, b).



a) Properties



b) Events

10.4-rasm. Object Inspector oynasi.

Propierities sahifasi ajratilgan obyekt yoki komponentning xossalarini o'rnatadi. Masalan, Caption (yozuv) yordamida obyekt yozuvin o'rnatish mumkin, Color (rang) xossasi obyektning rangani o'rnatadi. Events (hodisalar) sahifasida Forma ilovasida tanlangan obyekt uchun dastur bajarilishi jarayonida hodisa, ya'ni uni ishga tushirish holati belgilanadi. Har bir holatning standart nomi belgilangan. Masalan, OnClick – sichqonchanning chap tugmasini bir marta bosish, OnDblClick – sichqonchanning chap tugmasini bir marta bosish.

Borland C++ Builder 6 asosiy menyusi quyidagi bo'limlarni o'z ichiga oladi:

– File bo'limi fayllar ustida ish bajarish uchun kerakli buyruqlarni o'z ichiga olgan;

- *Edit* bo‘limi fayl ichidagi ma’lumotlarni tahrirlash uchun kerakli buyruqlarni o‘z ichiga olgan;
- *Search* bo‘limi fayllar, modullar tarkibidagi kerakli bo‘laklarni izlab topish imkonini beruvchi buyruqlar to‘plami;
- *View* bo‘limi dastur oynasiga kerakli instrumentlar palitrasini o‘rnatish, loyiha kodi, shuningdek, loyiha menejerini ochish va ko‘rish uchun mo‘ljallangan buyruqlar to‘plami;
- *Compile* bo‘limi loyiha va dasturlarni ishga tushirish, kompilatsiya buyruqlaridan tashkil topgan;
- *Run* – dasturni ishga tushirish va to‘xtatish uchun kerak bo‘ladigan buyruqlar to‘plami;
- *Options* bo‘limi muhit oynasining konfiguratsiya parametrlarini o‘rnatish uchun xizmat qiladigan buyruqlar to‘plami;
- *Tools* – servis xizmatidan va qo‘shimcha utilitlardan foydalanish imkonini beradi;
- *Help* – yordam chaqirish uchun mo‘ljallangan buyruqlar to‘plami.

Asosiy menyular qatori ko‘p bo‘limlarni o‘z ichiga oladi.

10.4. C++ Builder 6 dasturlash tilining asosiy konstruksiyalari

C++ Builder 6 dasturlash tili alifbosi o‘z ichiga 26 bosh lotin harflarini, 0 dan 9 gacha bo‘lgan arab raqamlarini oladi va quyidagi belgilarni ishlatadi: bo‘sh joy belgisi; - + * / : ; . , % ? ! = «» № < > { } [] () \$ # & ^ va h.k.

Alifbo simvollaridan tilning leksemasi shakllanadi, jumladan: xizmatchi so‘zlar, izohlar, identifikatorlar, amal belgilari, o‘zgarmas va o‘zgaruvchilar, ajratish belgilari.

Dasturda izohlar istalgan joyda berilishi mumkin. Ular odatda { } – katta qavslar ichida yoziladi.

Xizmatchi so‘zlar: Tilda ishlatiluvchi, ya’ni dasturchi tomonidan o‘zgaruvchilar nomlari sifatida ishlatish mumkin bo‘lmagan identifikatorlar xizmatchi so‘zlar deyiladi.

C++ tilida quyidagi xizmatchi so‘zlar mavjud:

C++ tilida ishlatiladigan xizmatchi so'zlar			
int	short	goto	for
char	unsigned	return	do
float	auto	sizeof	while
double	extern	break	switch
struct	register	continue	case
union	typedef	if	default
long	static	else	entry

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

Misol uchun: A1, _MAX, adress_15, LAB, lab.

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

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

O'zgaruvchilar. Dastur bajarilishi jarayonida o'z qiymatini o'zgartira oladigan kattaliklar o'zgaruvchilar deyiladi. O'zgaruvchilarning nomlari harfdan boshlanuvchi harf va raqamlardan iborat bo'lishi mumkin. O'zgaruvchilarni belgilashda katta va kichik harflarning farqlari bor (A va a harflari 2 ta o'zgaruvchini bildiradi). Har bir o'zgaruvchi o'z nomiga, toifasiga, xotiradan egallagan joyiga va son qiymatiga ega bo'lishi kerak. O'zgaruvchiga murojaat qilish uning ismi orqali bo'ladi. O'zgaruvchi uchun xotiradan ajratilgan joyning tartib raqami uning manzili hisoblanadi. O'zgaruvchi ishlatilishidan oldin u aniqlangan bo'lishi lozim.

O'zgarmlar. O'zgaruvchilar kabi o'zgarmlar ham ma'lumotlarni saqlash uchun mo'ljallangan xotira yacheykalarini o'zida ifodalaydi. O'zgaruvchilardan farqli ravishda ular dasturning bajarilishi jarayonida qiymati o'zgarib qolmaydi. O'zgarma e'lon qilinishi bilan unga qiymat berish lozim, keyinchalik bu qiymatni o'zgartirib bo'lmaydi.

C++ tilida ikki turdagi – literal va belgili o'zgarmlar aniqlangan. Literalli o'zgarmlar to'g'ridan to'g'ri dasturga kiritiladi. Masalan:

int myAge =40;

Belgili o'zgarma – bu nomga ega bo'lgan o'zgarma. Masalan:

const unsigned short int St=26

Belgili o'zgarmlarni literal o'zgarmlarga nisbatan ishlatish qulayroqdir. Chunki agarda bir xil nomli literalli o'zgaruvchining qiymatini o'zgartirmoqchi bo'lsangiz, butun dastur bo'yicha uni o'zgartirishga to'g'ri keladi, belgili o'zgarmlarni esa faqatgina birining qiymatini o'zgartirish yetarli.

Amallar. Amallar undagi qatnashayotgan operandlarga ko'ra unar, binar, ternar toifalarga bo'linadi.

Tilda quyidagi amallardan foydalanish mumkin:

- arifmetik amallar: +, -, /, *, %. Barcha amallar odatdagidek bajariladi, faqat bo'lish amali butun sonlar ustida bajarilayotgan bo'lsa, natija doim butun bo'ladi, ya'ni kasr qism tashlab yuboriladi ($9/5=1$; vaholangki $1,8$ bo'lishi kerak). Shuning uchun surat yoki maxrajiga nuqta (.) qo'yilsa, natija ham haqiqiy bo'ladi ($9./5=1.8$). % belgisi (modul operatori) esa butun sonni butun songa bo'lgandan hosil bo'ladigan qoldiqni bildiradi. Masalan: $9 \% 5=4$;

- taqqoslash amallari: == (tengmi?); != (teng emas); < ; > ; >=; <=;

- mantiqiy amallar: && (and) mantiqiy ko'paytirish; || (or) mantiqiy qo'shish; ! (not) mantiqiy inkor.

Mantiqiy amallarni ixtiyoriy sonlar ustida bajarish mumkin. Agar javob rost bo'lsa, natija 1 bo'ladi, agar javob yolg'on bo'lsa, natija 0 bo'ladi. Umuman olganda 0 (nol)dan farqli javob rost deb qabul qilinadi. Masalan:

$i > 50 \ \&\& \ j = 24$ yoki $s1 < s2 \ \&\& \ (s3 > 50 \ || \ s4 \leq 20)$;

yoki $8 \leq x \leq 12$ yozuvini $x \geq 8 \ \&\& \ x \leq 12$ deb yoziladi.

Qiymat berish amallari:

a) qiymat berish amali belgisi «=» bo'lib, uning yordamida odatda ma'lum o'zgaruvchiga qiymat o'zlashtiriladi, masalan, $a=7$; $b=9*c$; $x=y=z=10$;

a) inkrement amali (++) ikki ma'noda ishlatiladi: o'zgaruvchiga murojaat qilinganidan keyin uning qiymati 1 ga oshadi (a++) va o'zgaruvchining qiymati uning murojaat qilishdan oldin 1 ga oshadi (++a);

b) dekrement amali (--), xuddi inkrement amali kabi, faqat kamaytirish uchun ishlatiladi. Masalan: $c = a + b++$ (a ga b ni qo'shib, keyin b ning qiymatini 1 ga oshiradi); $c = a + (--b)$ (b ning qiymatini 1 ga kamaytirib, keyin a ga qo'shadi);

c) C++ tilida ishlatiladigan qisqartirilib yoziladigan amallar ham ishlatiladi (10.2-jadval).

10.2-jadval

Qisqartirilgan yozuv	To'liq yozuv
$x += a;$	$x = x + a;$
$x -= a;$	$x = x - a;$
$x *= a;$	$x = x * a;$
$x /= a;$	$x = x / a;$
$x \% = a;$	$x = x \% a;$

C++ tilida ishlatiladigan amallar quyidagi ko'rinishda ishlatiladi:

10.3-jadval

Arifmetik amallar	Razryadli amallar	Nisbat amallari	Mantiqiy amallar
+ qo'shish	& va	= = teng	&& va
- bo'lish	yoki	!= teng emas	yoki

* ko'paytirish	^ inkor	> katta	! inkor
/ bo'lish	<< chapga surish	>= katta yoki teng	
% modul olish	>>o'ngga surish	< kichik	
- unar minus	~ inkor	<= kichik yoki teng	
+ unar plyus			
++ oshirish			
-- kamaytirish			

Preprotessor direktivalari. Direktivalar kompilatsiya oldidan dasturning boshlang'ich matnini qayta ishlash uchun mo'ljallangan. Har qanday direktiva «#» belgisidan boshlanadi. Bitta qatorda faqat bitta direktiva yozilishi mumkin. Masalan, #include «myfile» dastur matniga myfile nomli sarlavha faylining tarkibi qo'yiladi. Sarlavha fayli dastur kompilatsiyasining muvaffaqiyatli bajarilishi uchun zarur bo'lgan turli axborotlarni o'zida saqlaydi.

10.5. Texnik tizimlardagi ma'lumotlarning toifalari strukturasi

Dasturda ma'lumotlarning toifasi berilganlarning qiymatlar to'plamini va shu bilan birga ular ustida bajariladigan amallarni belgilaydi. Kompilator buyruqlarni shakllantirish uchun ma'lumotlar xotirada qancha joy egallashini va bajariladigan amallarni aniq bilishi lozim. Bularning hammasi ma'lumotlarning toifasini tavsiflash bilan belgilanadi. Dasturda ishlatiladigan o'zgaruvchilar, o'zgarmaslar va amallarning natijalari aniq toifaga mansub bo'lishi lozim.

C++ tili toifalari elementar (asosiy-tayanch) va tarkiblashgan turlarga bo'linadi. Elementar toifa butun, haqiqiy, mantiqiy va simvolli toifalarni o'z ichiga oladi:

- bool (mantiqiy);
- char (simvolli);

- wchar_t (katta diapazondagi simvolli);
- int (butun);
- float (haqiqiy);
- double (ikkilangan aniqlangandagi haqiqiy toifa).

Bular asosida tarkibiy toifalar shakllanadi. Tarkibiy toifalar massivlar, strukturali (tarkiblashgan), ko'rsatkichlar va sinflarni o'z ichiga oladi.

Quyidagi rasmda C++ tili standartidagi toifalar sxemasi keltirilgan.

Butun toifalar. Butun toifalar butun sonlarni tavsiflash uchun ishlatiladi. C++ tilida butun toifadagi ma'lumotlar quyidagi turlarda beriladi:

- short (qisqa);
- long (uzun);
- signed (ishorali);
- unsigned (ishorasiz).

4 ta ishorali butun sonlar: signed char, short int, int, long int va 4 ta ishorasiz butun sonlar: unsigned char, unsigned short int, unsigned int, unsigned long int ishlatiladi.

Unsigned manfiy butun sonlarni ifodalaydi. Dasturda short int, long int, signed int va unsigned int nomlarni mos ravishda short, long, signed va unsigned nomlar bilan almashtirish mumkin. Bundan tashqari, butun sonlarga avtomat ravishda signed int toifasi beriladi.

Tartiblangan o'zgarmaslar enum xizmatchi so'zi orqali ifodalanaadi. Masalan, enum {one=1, two=2, three=3, four=4}; enum {zero, one, two, three};

– agar ro'yxatda o'zgarmaslarga qiymat berilmagan bo'lsa, ularga qiymatlar 0 dan boshlab avtomat ravishda o'zlashtiriladi: zero=0, one=1, two=2, three=3.

Visual C++.NET 2005 bir nechta nostandart butun sonlarni ishlatish imkonini beradi. Masalan, long long toifasi -9 223 372 036 854 775 808 dan +9 223 372 036 85 775 807 gacha oraliqdagi butun sonlarni ifodalaydi, xotirada 8 bayt joy egallaydi.

Haqiqiy toifa. C++ standartining float, double va long double turlari aniqlangan. Ularning hammasi ishorali. Haqiqiy son mantissa

va tartibdan tashkil topib, mantissa sonning aniqligini, tartibi uning qiymatlar diapazonini belgilaydi. Qo'zg'aluvchan nuqtali haqiqiy sonlar avtomat arvishda double toifasi bilan beriladi. Sonlar toifasini aniq ko'rsatish mumkin, buning uchun F, f (float) va L, l(long) sufikslarini ko'rsatish mumkin. Masalan, 3.14F, 2E+6L (long double toifasiga tegishli).

Mantiqiy toifa. Mantiqiy toifadagi kattaliklar true va false qiymatlarini qabul qiladi. Ular arifmetik amallarda ishtirok etishi mumkin. Bu kattaliklarni butun toifaga o'g'irganda mos ravishda true- 1 ga, false- 0 ga tenglashtiriladi.

Belgili (simvolli) toifa. C++ tilida 3 ta simvolli toifa belgilangan: char, signed char va unsigned char. Har bir simvolga xotirada faqat 1 bayt joy ajratiladi:

$\text{sizeof(char)} = \text{sizeof(signed char)} = \text{sizeof(unsigned char)} = 1.$

Quyidagi jadvalda C++ da ishlatiladigan toifalar ro'yxati va ularning qiymat diapazonlari keltirilgan (10.4-jadval).

10.4-jadval

Toifa	Qiymatlar diapazoni	Olchami (bayt)
bool	True va false	1
signed char	-128 ... 127	1
unsigned char	0 ... 255	1
signed short int	-32 768 ... 32 767	2
unsigned short int	0 ... 65 535	2
signed int	-2 147 483 648 ... 2 147 483 647	4
unsigned int	0 ... 4 294 967 295	4
signed long int	-2 147 483 648 ... 2 147 483 647	4
unsigned long int	0 ... 4 294 967 295	4
float	3.4e-38 ... 3.4e+38	4

double	1.7e-308 ... 1.7e+308	8
long double	3.4e-4932 ... 3.4e+4932	10

Standart matematik funksiyalarning yozilishi:

Dasturlarda buyruqlar tarkibida turli ko'rinishdagi standart funksiyalardan foydalanish mumkin. Matematik funksiyalar arifmetik amallarda ishlatiladi. Quyidagi jadvalda standart matematik funksiyalar keltirilgan (10.5-jadval).

10.5-jadval

Funksiya	Ifodalanishi	Funksiya	Ifodalanishi
Sin x	sin(x)	$\sqrt{x}$	sqrt(x); pow(x,1/2.)
Cos x	cos(x)	x	abs(x) ёки fabs(x)
Tg x	tan(x)	Arctan x	atan(x)
e ^x	exp(x)	Arcsin x	asin(x) ?
Ln x	log(x)	Arccos x	acos(x) ?
Lg x	log10(x)	$\sqrt[3]{x^2}$	pow(x,2/3.)
x ^a	pow(x,a)	Log ₂ x	log(x)/log(2)

Masalan:

$$\frac{k + \sqrt{k^3 - 8k}}{2c} \rightarrow (k + \text{sqrt}(\text{pow}(k,3) - 8*k*c)) / (2*c);$$

yoki

$$e^{\sin x} + \lg^2(x+3) \rightarrow \text{exp}(\text{sin}(x)) + \text{pow}(\text{tan}(x+3), 2);$$

$$k = (m*5) + ((7 \% n) / (9+x));$$

Quyida keltirilgan funksiyalar ma'lumotlar toifasini o'zgartirish (o'girish) vazifasini bajaradi (10.6-jadval).

10.6-jadval

Funksiyaning C++dagi ifodasi	Funksiyaning vazifasi
CHR(N)	Kodi n (butun son)ga teng simvolni aniqlash
INTTOSTR(K)	Butun k sonini satrga o'tkazish
FLOATTOSTR (N)	Haqiqiy n sonini satrga o'tkazish
FLOATTOSTR(N,F,K,M)	Haqiqiy n sonini satrga o'tkazish, bunda: f – format; k – aniqlik; m – kasr qismidagi raqamlar soni
STRTOINT (S)	S Satrni butun songa o'tkazish
STRTOFLOAT (S)	S Satrni haqiqiy songa o'tkazish
ROUND (N)	Haqiqiy sonni yaxlitlash
TRUNC (N)	Haqiqiy son kasr qismini olib tashlash
FRAC(N)	Kasrli sonning kasr qismi
INT (N)	Kasr sonning butun qismi

C++ algoritmik tilida arifmetik, mantiqiy va belgili ifodalar aniqlangan.

Arifmetik ifodalar o'zgaruvchilar, o'zgarmaslar, arifmetik amal belgilari qavslardan va standart funksiyalardan tashkil topadi. Arifmetik ifodada qatnashayotgan amallar ustuvorligiga qarab bajariladi.

Amallar quyidagi ustuvorlikka ega:

- qavs ichidagi amallar;
- funksiyalarni chaqirish;
- ko'paytirish, bo'lish, qoldiqni aniqlash;
- qo'shish, ayirish.

Rang	Amallar	Yo'nalish
1	() [] -> :: .	Chapdan o'ngga
2	! ~ + - ++ -- & * (tip) sizeof new delete tip()	O'ngdan chapga
3	. * -> *	Chapdan o'ngga
4	* / % (multiplikativ binar amallar)	Chapdan o'ngga
5	+ - (additiv binar amallar)	Chapdan o'ngga
6	<<>>	Chapdan o'ngga
7	<<= >= >	Chapdan o'ngga
8	= !=	Chapdan o'ngga
9	&	Chapdan o'ngga
10	^	Chapdan o'ngga
11		Chapdan o'ngga
12	&&	Chapdan o'ngga
13		Chapdan o'ngga
14	?:(shartli amal)	Chapdan o'ngga
15	= *= /= %= += -= &= ^= = <<= >>=	Chapdan o'ngga
16	, (vergul amali)	Chapdan o'ngga

Arifmetik ifodada bir xil ustuvorlikka ega amallar ketma-ket kel-sa, ular chapdan o'ngga qarab bajariladi.

Misol:

$$e^{\sin(x)+\cos x} + \lg(x+3).$$

C++da bu ifoda quyidagi ko'rinishda yoziladi:

$$\exp(\sin(\sqrt{x})+\cos(\sqrt{x}))+\sin(x+\sqrt{3})/\cos(x+\sqrt{3}).$$

Mantiqiy ifodalar o'z ichiga mantiqiy o'zgarmlarni, mantiqiy o'zgaruvchilarni, mantiqiy amal belgilarini, mantiqiy natija beradigan standart funksiyalarni qamrab oladi.

Ifodada solishtirish amallari ($>$, $<$, $=$, $<=$, $=>$, $<>$), mantiqiy ko'paytirish $\&\&$, mantiqiy qo'shish II va mantiqiy inkor qilish $!$ amallarini bajarish mumkin. Quyidagi jadvalda BOOLEAN toifasidagi A va B o'zgaruvchilari ustida mantiqiy amallarning bajarilishi natijasi keltirilgan (10.8-jadval).

10.8-jadval

A	B	! A	!B	A && B	A II B
TRUE	TRUE	FALSE	FALSE	TRUE	TRUE
TRUE	FALSE	FALSE	TRUE	FALSE	TRUE
FALSE	TRUE	TRUE	FALSE	FALSE	TRUE
FALSE	FALSE	TRUE	TRUE	FALSE	FALSE

Mantiqiy ifodada ham amallar ustuvorligiga ko'ra bajariladi:

1. Qavs ichidagi amal va mantiqiy inkor amali (!).
2. Mantiqiy ko'paytirish ($\&\&$) amallari.
3. Mantiqiy qo'shish (II) va solishtirish amallari.

Quyidagi misolda berilgan mantiqiy ifodada amallarning bajarilish ketma-ketligi keltirilgan, bu yerda $A=2$, $B=4$, $C=5$;

$(A>3) \&\& (A=C+3) \text{ II } ! (B=5)$

1) $(A>3) = \text{FALSE}$

2) $(A=C+3) = \text{FALSE}$

3) $(B=5) = \text{FALSE}$

4) $!(B=5) = \text{TRUE}$

5) $(A>3) \&\& (A=C+3) = \text{FALSE}$

6) $(A>3) \&\& (A=C+3) \text{ II } ! (B=5) = \text{TRUE}$

Bu mantiqiy ifodaning qiymati TRUEga teng.

10.6. C++ dasturlash tilining standart kutbxonalari

C++ tilida dastur preprotssessor direktivalari, global obyektlar va funksiyalarning tavsiflari va ta'riflari ketma-ketligidan tashkil topadi. Direktivalar dasturni kompilatsiya qilingungacha boshqaradi. Ta'riflar funksiyalarni va obyektlarni kiritadi. Obyektlar dasturdagi qayta ishlanadigan ma'lumotlarni tasvirlash uchun zarur. Funksiyalar dasturdagi amallarni aniqlaydi.

Tavsiflar kompilatorga dasturning turli qismlari yoki boshqa fayllarda aniqlangan obyekt va funksiyalarning nomi va xossalari haqida ma'lumot beradi.

Dastur bitta yoki bir nechta matnli fayl ko'rinishida shakllanadi. Matnli fayl qatorlarga ajratilgan bo'ladi. Har bir qator qator oxiri belgisi bilan tugatiladi.

Dastur qayta ishlashning uchta bosqichidan o'tadi:

- matnni preprotssessor yordamida o'g'irish;
- kompilatsiya qilish;
- kompanovkalash (barcha bog'lanishlarni tahrirlash yoki yig'ish).

Yuqorida keltirilgan bosqichlar muvaffaqiyatli tugatilgan so'ng dasturning mashina kodi shakllanadi.

Preprotssessorning vazifasi dastur matnini uni kompilatsiyaga cha o'g'irishdir. Preprotssessorli qayta ishlashning tartibini dasturchi direktivalar yordamida belgilaydi. Har bir direktiva «#» belgisidan boshlanadi, masalan, #include yoki #define. #define matnda o'zgartirish qoidalarini, «#»include esa dasturda qanday matnli fayllarni qo'shish lozimligini ko'rsatadi. #include direktivasi kompilatorning standart kutubxonalari bilan birgalikda uzatiladigan sarlavha fayllari katalogidan kerakli fayl matnini dastur matniga qo'shadi. Standart kutubxonalar uchun sarlavha fayllari ro'yxati C++ tili standartida belgilangan. Global obyektlar va funksiyalarning tavsiflari va ta'riflari albatta main nomli funksiya bo'lishi shart. Chunki bu funksiya dasturning asosiy funksiyasi bo'lib, usiz dastur bajarilmaydi. C++da dastur tuzish va bajarish natijasida turli ko'rinishagi ilovalar yaratiladi. Asosan 2 ta muhitda dastur tuzish va bajarish mumkin: konsol va forma ilova muhitida.

C++ tilidagi dasturning har bir bo'lagini qarab chiqamiz.

Direktivalar – # include <file.h> direktiva – instruksiya degan ma'noni beradi. C++ tilida dasturning tuzilishiga, ya'ni ehtiyojiga qarab, kerakli direktivalar ishlatiladi. Ular <> belgisi orasida keltiriladi. Umuman olganda, quyidagi asosiy direktivalar mavjud (jami 32 ta):

– # include <stdio.h> – C da oddiy kiritish/chiqarish dasturi uchun. Bu yerda std – standart, i – input, o – output degani;

– # include <iostream.h> – C++ da kiritish/chiqarish uchun, oddiy amallar bajarilsa;

– # include <math.h> – standart funksiyalarni ishlatish uchun;

– # include <conio.h> – dasturning tashqi ko'rinishini shakllantirish uchun;

– # include <string.h> – satr toifasidagi o'zgaruvchilar ustida amallar bajarish uchun;

– # include <stdlib.h> – standart kutubxona fayllarini chaqirish uchun;

– # include <time.h> – kompyuter ichidagi soat qiymatlaridan foydalanish uchun;

– # include <graphics.h> – C++ tilining grafik imkoniyatlaridan foydalanish uchun.

Bu fayllar maxsus kutubxona e'lon fayllari hisoblanadi va ular alohida include deb nomlanadigan papkada saqlanadi. Hozirda C++ kutubxonasi yangilandi va undagi fayllarning nomlaridan «.h» (head – sarlavha ma'nosida) kengaytmasi olib tashlandi va oldiga «c» harfi qo'shildi (C dan qolgan 18 tasiga). Bu fayllarda funksiya prototiplari, toifalari, o'zgaruvchilar, o'zgarmaslar ta'riflari yozilgan bo'ladi.

Dastur normal ishga tushgandan so'ng ekranda standart dastur oynasi namoyon bo'ladi va unda qiymatlar kiritiladi, dastur natijasida olingan qiymatlar chiqadi.

Loyiha fayli yaratilishi bilan avtomatik tarzda quyidagi fayllar ham tuziladi, ularni bitta papkada saqlash maqsadga muvofiq:

– Project1.bpr – bosh loyiha fayl bo'lib, asosiy komponentlarni saqlovchi va ishga tushiruvchi fayl;

– Unit1.cpp – dastur matni yozilgan fayl;

– Project1.exe – ilova fayli yoki bajariluvchi fayl. Bu fayl kompilator yordamida, ya'ni kompilatsiya jarayonida tuziladi. [F9] tugmasini bosish bilan, bajariluvchi fayl avtomatik ravishda tuziladi. Bajarliluvchi fayl avtonom fayl bo'lib, uning uchun boshqa fayl yoki biror dasturiy tizim mavjud bo'lishi shart emas. Uni siz boshqa dasturlar kabi ishga tushirishingiz mumkin;

– Project1.res – loyiha resurs fayli.

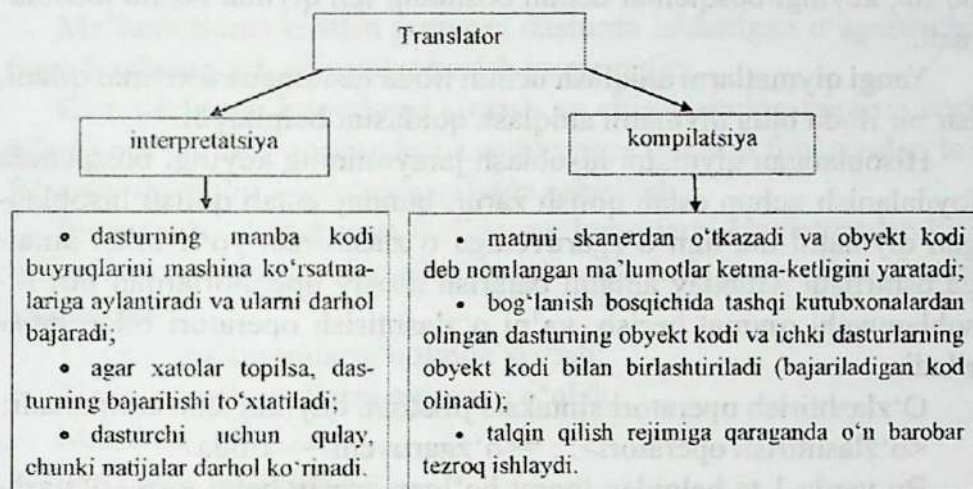
Nazorat savollari

1. Vizual loyiha deb nimaga aytiladi?
2. Dasturlashda qanday toifalar mavjud?
3. Dasturda izohlar qanday ishlatiladi?
4. C++ Builder 6 ning interfeysi nechta oynadan tashkil topgan?
5. Include qanday ma'noga ega?

11-BOB. C++ BUILDER 6 DASTURLASH TIZIMIDA CHIZIQLI DASTURLARNING QO'LLANILISHI

11.1. Borland C++ builder 6 tizimidagi operatorlar tasnifi

Yuqori darajali dasturlash tillarida yozilgan dasturlarni mashina tiliga tarjima qiladigan maxsus dasturlar translatorlar deyiladi.



11.1-rasm. Translatorning vazifalari.

C++ tilida operatorlar bajariladigan va bajarilmaydigan, sodda va tarkibiy operatorlarga bo'linadi. Bajariladigan operatorlar ma'lumotlar ustida biron-bir amallar bajaradi, bajarilmaydigan operatorlar ma'lumotlarni tavsiflash uchun ishlatiladi, masalan, `int a;` – butun toifali `a` – o'zgaruvchini tavsiflash operatori bo'lib xizmat qiladi.

Sodda operatorlar qatoriga kiritish va chiqarish, o'zlashtirish operatorlari kiradi.

Tarkibiy operator yoki blok – «`{ }`» figurali qavslar ichiga olingan operatorlar guruhidir.

Bundan tashqari, boshqaruvchi operatorlar toifasi ham ajratilgan, ularga quyidagilar kiradi:

- tanlash operatorlari – if va switch;
- takrolanish (sikl) operatorlari – while, do while, for;
- o'tish operatorlari – break, goto, return.

Masalani yechish jarayoni qator bajariluvchi bosqichlarga bo'linib ketadi. Bu bosqichlarning har birida ma'lum qiymatlar bo'yicha yangi qiymatlar aniqlanadi (hisoblanadi). Bu aniqlangan qiymatlarning ba'zilar natijaviy qiymatlar bo'lsa, ba'zilar esa oraliq qiymatlar bo'lib, keyingi bosqichlar uchun boshlang'ich qiymat bo'lib hisoblanadi.

Yangi qiymatlarni aniqlash uchun ifoda tushunchasi xizmat qiladi, har bir ifoda bitta qiymatni aniqlash qoidasini belgilaydi.

Hisoblangan qiymatni hisoblash jarayonining keyingi bosqichida foydalanish uchun eslab qolish zarur, bunday eslab qolish hisoblangan qiymatni ma'lum o'zgaruvchiga o'zlashtirish yo'li bilan amalga oshiriladi. Bunday amalni bajarish asosiy operatorlardan biri hisoblanuvchi qiymat berish, ya'ni o'zlashtirish operatori bilan bajariladi.

O'zlashtirish operatori sintaksis jihatdan quyidagicha aniqlanadi:
<o'zlashtirish operatori> : = <o'zgaruvchi> = <ifoda>.

Bu yerda 1 ta belgidan iborat bo'lgan asosiy belgi «=» «o'zlashtirish» deb o'qiladi.

O'zlashtirish operatorining bajarilishida «=» belgisining o'ng tomonidagi ifodaning qiymati hisoblanib, belgining chap tomonidagi o'zgaruvchi tomonidan o'zlashtiriladi. Operatorning o'ng tomonidagi ifoda o'rnida konstanta, qiymati aniqlangan o'zgaruvchi ham ishlatilishi mumkin.

Masalan:

$$y = \cos(y + 2.88) + \cos(y + 5.67); y = 13;$$

b = 'Texnik tizimlarda axborot texnologiyasi';

Operatorning chap va o'ng tomonida yozilgan kattaliklar bir toifada e'lon qilinishi zarur. O'zlashtirish operatori bajarilishi natijasi-

da ma'lum o'zgaruvchilar keyingi bosqichlarda ishlatilishi mumkin bo'lgan yangi joriy qiymatni qabul qiladi.

Operatorning o'ng tomonidagi ifoda o'rnida yana o'zlashtirish operatori qo'llanishi mumkin, masalan, $x = y = c$;

O'zlashtirish operatorining o'ng va chap tomonida $+=$, $*=$, $/=$ va h.k. amallarini ham ishlatish mumkin, masalan, $a += b$ yoki $b = a++$. Bu amallarni keyingi bo'limlarda to'liqroq ko'rib chiqamiz.

11.2. C++ tilida ma'lumotlarni kiritish va chiqarish

Ma'lumotlarni kiritish jarayoni dasturda ishlatilgan o'zgaruvchilarga boshlang'ich qiymatlar berish jarayonidir.

C++ tilida ma'lumotlarni kiritish va chiqarishning maxsus vositalari yoq, bu jarayon standart kutubxonaga mansub funksiyalar, toifalar va obyektlar yordamida amalga oshiriladi.

`iostream.h` – kutubxona faylida ma'lumotlarni klaviaturadan kiritish «cin» va ekranga chiqarish «cout» oqimlari, ularga mos amallar aniqlangan:

- 1) `<<` – ma'lumotlarni oqimga yozish;
- 2) `>>` – ma'lumotlarni oqimdan o'qish.

Masalan:

```
#include<iostream.h>;
```

```
.....  
cout << «a o'zgaruvchining natijasi= »;
```

```
cin >> x;
```

Ma'lumotlarni chiqarish oqimi «cout<<» ko'rinishida va ma'lumotlarni kiritish «cin>>» ko'rinishida yoziladi. cout – console output va cin – console input ma'nolarini bildiradi.

Ma'lumotlarni kiritishda «cin» so'zidan keyin o'zgaruvchilarning nomlari keltiriladi. Agar ular bir nechta bo'lsa, har bir o'zgaruvchi orasiga >> belgisi qo'yiladi.

Masalan:

```
cin >> a >> b;
```

```
cin >> max >> min >> a2;
```


Bu o'zgaruvchilarning son qiymatlari dastur kompilatsiyadan o'tganidan keyin klaviatura orqali beriladi. Sonlarni bo'sh joy yoki «Enter» klavishi orqali kiritish mumkin. «cin» operatori orqali faqat son qiymatlar kiritiladi, ifodalar yozilishi mumkin emas.

Ma'lumotlarni chiqarish uchun «cout <<» operatori ishlatiladi. Bu yerda o'zgaruvchi nomlari, qo'shtirnoq ichida istalgan so'z yoki iboralar yozilishi mumkin. Agar qiymati chiqarilayotgan o'zgaruvchilar bir nechta bo'lsa, ularni alohida qatorlarda yoki ketma-ket chiqarilishi mumkin. Masalan:

```
cout << a;
```

```
cout << x << y;
```

```
cout << «y=» << y;
```

```
cout << «ifodaning natijasi teng =» << fax;
```

Bir nechta natijalar chiqarilayotgan bo'lsa, ular ekranda ketma-ket ko'rinishda namoyon bo'ladilar. Bu holat esa natijalarni o'qishda noqulaylik tug'dirishi mumkin. Chiqarish operatorini boshqarishda turli ko'rinishdagi manipulatorlardan foydalanish mumkin.

Manipulator – bu oqimlarga uzatiladigan ma'lumotlarni formatlash uchun kiritish-chiqarish oqimlarini boshqaruvchi obyektning maxsus turi. Ularning vazifalari 11.1-jadvalda keltirilgan.

11.1-jadval

Manipulator	Vazifasi	Masalan	Natija
endl	Natija chiqarishda yangi qatorga o'tish	cout<<«Salom!<<endl<<«Qiymat kiriting:»;	Salom! Qiymat kiriting:
oct	Sakkizlik sanoq sistemasida kattaliklarni chiqarish	intx= 64; cout<<oct<<x<<endl;	100
hex	O'n oltilik sanoq sistemasida kattaliklarni chiqarish	intx= 64; cout<<hex<<x<<endl;	40

showpos	Musbat raqamlar uchun + belgisini chiqarish	intx= 255; cout<<showpos<<x<<endl	+255
setprecision (n)	Verguldan so'ng belgilar sonini chiqarish	cout<<fixed << setprecision(3)<<(13.5/2)<< endl;	6.750

«cout» operatori bilan yana quyidagi belgilar ham ishlatilishi mumkin (ularni ESC simvollari ham deyiladi):

- \n – yangi satr. Kursor yangi qator boshiga o'tadi;
- \t – gorizontal tabulatsiya, kursor bir nechta harf o'ngga siljiydi;
- \v – vertikal tabulatsiya, kursor bir nechta satr tashlab o'tadi;
- \r – qaytish, ya'ni kursor ayni satr boshiga qaytadi, yangi satrga o'tmaydi;
- \a – kompyuter dinamik ovoz chiqaradi;
- \n belgisi bilan endl so'zi orasida farq bor.

Haqiqiy sonlarni ekranga chiqarishda ularni formatlash, ya'ni nuqtadan keyin necha xonagacha aniqlikda olishni boshqarish mumkin. Buning uchun

cout.precision(n); – funksiyasi ishlatiladi, bu yerda n – aniqlik ko'rsatkichi.

Masalan: cout.precision(4);

cout << y;

Bu funksiya dasturda bir marta yozilsa kifoya qiladi.

C++ tilida simvollarni kiritish va chiqarish uchun bir qator standart funksiyalar ishlatiladi:

– getch(arg) – bitta simvol kiritilishini kutish. Kiritilayotgan simvol monitorda aks etmaydi. Bu funksiyani dastur oxirida argumentsiz ishlatilsa, monitorda ma'lumotlarni toki klavish bosilguncha o'qish mumkin bo'ladi;

– putch(arg) – bitta simvolni standart oqimga chiqarish uchun ishlatiladi. Simvol monitorda aks etmaydi;

– `getchar(arg)` – bitta simvol kiritilishini kutish. Kiritilayotgan simvol monitorda aks etadi. Bu funktsiyani dastur oxirida argumentsiz ishlatilsa, monitorda ma'lumotlarni klavish bosilguncha o'qish mumkin bo'ladi;

– `putchar(arg)` – bitta simvolni standart oqimga chiqarish uchun ishlatiladi. Simvol monitorda aks etadi. Bu funktsiyalar `iostream.h` modulida joylashgan.

11.3. Chiziqli hisoblash jarayonlarni dasturlash

C++ Builder 6 tizimida chiziqli jarayonlarni konsol muhitida dasturlash uchun quyidagi ketma-ketlik bajariladi:

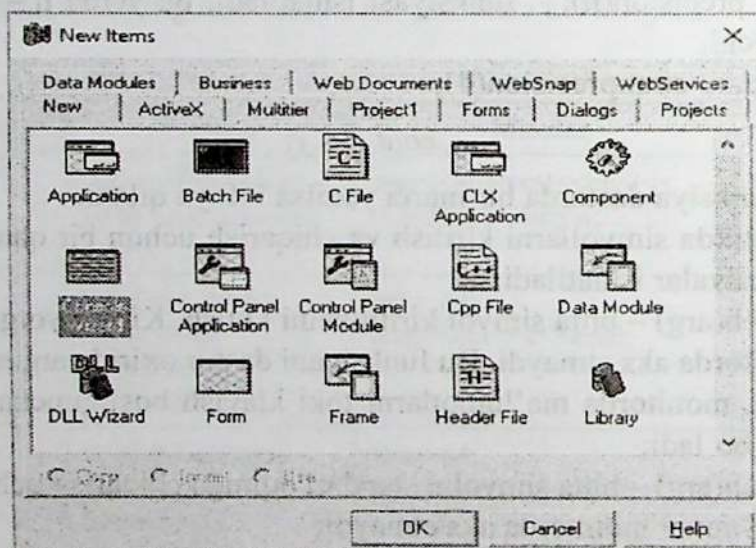
1) Borland C++ Builder 6 muhiti ishga tushiriladi;

Пуск -> Все программы -> Borland C++ Builder 6 -> C++ Builder 6

2) bosh menyuning «File» bo'limidan «New» va undan «Other» buyruqlari ketma-ketligi tanlanadi:

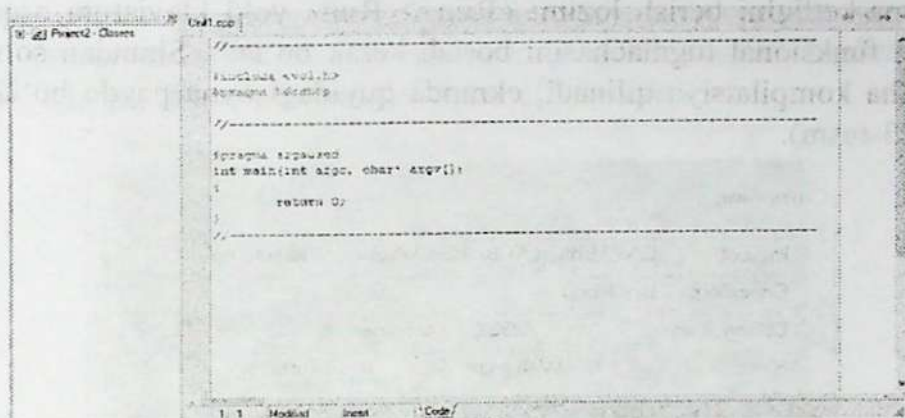
File -> New -> Other

3) Forma va loyihalarni saqlash uchun ochilgan maxsus oynadan «Console Wizard» piktogrammasi tanlanadi va «Ok» tugmasi bosiladi (11.1-rasm).



11.1-rasm. Konsol muhitini tanlash.

Natijada ekranda Unit.cpp kengaytmali nom bilan loyiha oynasi ochiladi.



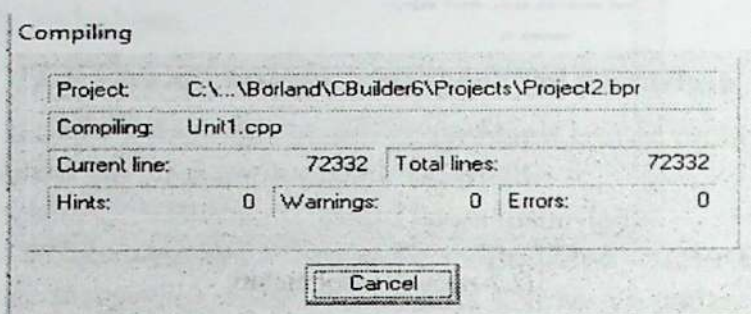
11.2-rasm. Konsol muhiti.

C++ tizimida { } qavslari ichiga loyiha faylining dastur matni kiritiladi (int main (int argc, char* argv[]) qatoridan so'ng).

```
//-----  
#include <vcl.h>  
#pragma hdrstop  
//-----  
#pragma argsused  
int main(int argc, char* argv[])  
{  
    //dastur matnini kiritish qismi  
    return 0;  
}  
//-----
```

Tuzilgan dasturni ishga tushirishdan oldin uni saqlash kerak bo'la-di. Uni saqlash uchun asosiy menyudan «File=> Save All» buyruqlar ketma-ketligini tanlash lozim. Har bir loyiha alohida yangi papka-ga saqlanishi tavsiya etiladi. Loyihaga avtomat ravishda «Project1.bpr» nomi berilgan bo'lib, bu nomni dasturchi o'zgartirishi mumkin. Bu yerda «1» – har bir ketma-ket nomlanadigan loyiha nomeri (son,

masalan 1, 2, 3, ...). Loyihani saqlab bo'lgandan so'ng, uni bajarishga beramiz. Buning uchun asosiy menyudan quyidagi buyruqlar ketma-ketligini berish lozim: «Run=> Run» yoki klaviatura orqali [F9] funksional tugmachasini bosish kerak bo'ladi. Shundan so'ng loyiha kompilatsiya qilinadi, ekranda quyidagi oyna paydo bo'ladi (11.3-rasm).



11.3-rasm. Kompilatsiyasi jarayoni oynasi.

Konsol muhitida ba'zi bir oddiy chiziqli jarayonlarning texnik masalalarini yechish dasturlarini ko'rib chiqamiz.

1-misol. Berilgan $(1+x)^2 / \sqrt{1+x^2}$ ifodani hisoblash dasturi tuzilsin.

```
//-----
#include<iostream.h>
#include<conio.h>
#include<math.h>
#include <vcl.h>
#pragma hdrstop
//-----
#pragma argsused
int main(int argc, char* argv[])
{ float x,a;
  cin>>x;
  a=pow((1+x),2)/sqrt(1+pow(x,2));
  cout<<"a="<<a<<endl;
```

```

getch(); return 0;
return 0;
}

```

Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi oyna paydo bo'ladi (11.4-rasm).



11.4-rasm. Dastur natijasi oynasi.

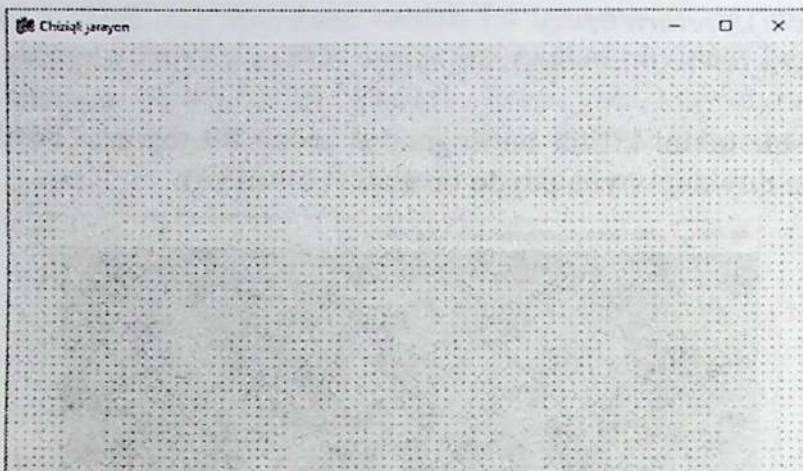
11.4. Forma ilovasida chiziqli jarayonlarni dasturlash

Forma ilovasida chiziqli jarayonlarni dasturlash vizual komponentlar yordamida amalga oshiriladi. Formaga qiymatlarni kiritish, natijani ilovaga chiqarish va qiymatlarni o'zlashtirgan o'zgaruvchilar ustida amallar bajarish maxsus komponentlar yordamida bajariladi. Vizual dasturlash muhitida bu turdagi masalalarni yechish uchun Formaga o'zgaruvchilarga qiymat kiritish – Edit komponenti, matn yozish uchun – Label, dasturni ishga tushirish uchun Button yoki Bit-Button komponentlaridan foydalaniladi.

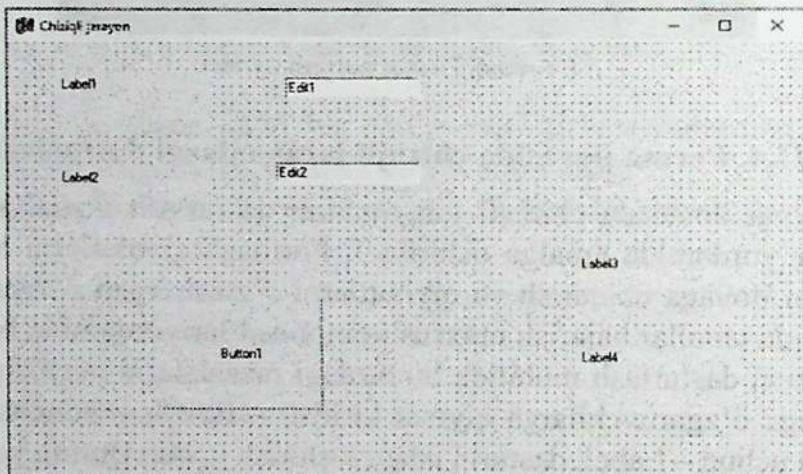
1-misol. va ifodalar berilgan. Bu yerda $x=1.43$, $y=18.5$ ga teng. Ushbu ifodalari hisoblovchi dastur tuzilsin.

Ifodani vizual muhitda dasturlash uchun 4 ta Label, 2 ta Edit va 1 ta Button komponentlari kerak bo'ladi.

Forma oynasiga kerakli komponentlar o'rnatilgandan so'ng dastur ko'rinishi quyidagi holatga keladi:



11.5-rasm. Forma oynasi.



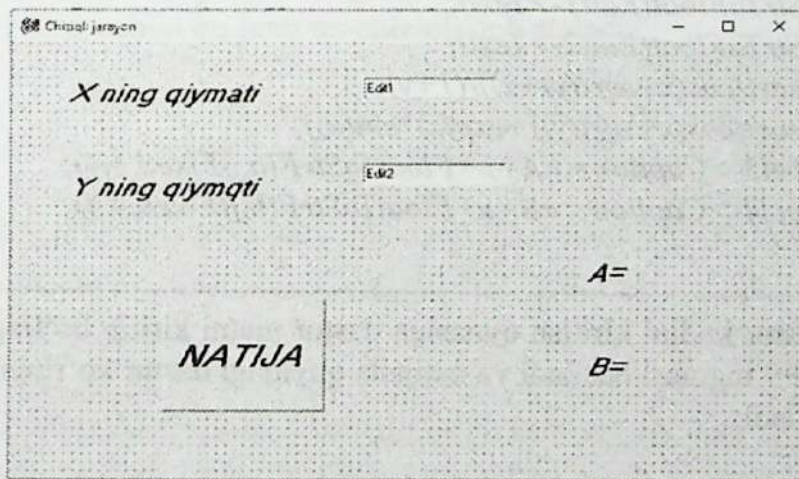
11.6-rasm. Dastur loyihasini tuzish.

Forma darchasiga oʻrnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Chiziqli jarayon» so'zi kiritiladi.
Label1	Caption (Properties)	«X ning qiymati» so'zi kiritiladi.
Label2	Caption (Properties)	«Y ning qiymati» so'zi kiritiladi.
Label3	Caption (Properties)	«A=» so'zi kiritiladi.
Label4	Caption (Properties)	«B=» so'zi kiritiladi.
Edit1	Text (Properties)	«Edit1» so'zini o'chirib tashlang.
Edit2	Text (Properties)	«Edit2» so'zini o'chirib tashlang.
Button1	Caption (Properties)	«Natija» so'zi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.

Komponentlar xususiyatlari kiritilgandan so'ng dastur loyihasi quyidagi ko'rinishga keladi:



11.7-rasm. Forma oynasida dastur loyihasining tuzilishi.

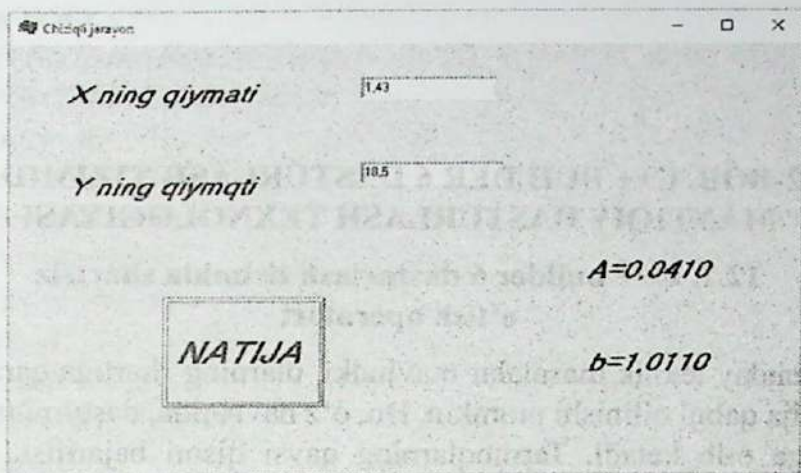
Dastur loyihasi tayyor bo'lganidan so'ng quyidagi dastur matni kiritiladi:

```

/-----
#include <math.h>
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
float x,y,a,b;
x=StrToFloat(Edit1->Text);
y=StrToFloat(Edit2->Text);
a=sqrt(cos(x+sqrt(y)+x))/(1+y);
b=sqrt(cos(x+sqrt(y))+exp(x)*sin(a));
Label3->Caption= «A=»+FloatToStrF(a,ffFixed,4,4);
Label4->Caption= «B=»+FloatToStrF(b,ffFixed,4,4);
}
//-----

```

Dastur kodini kiritish oynasiga dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



11.8-rasm. Natija oynasi.

Nazorat savollari

1. C++Builder 6 dasturlash muhitining asosiy elementlari nimalardan tashkil topgan?
2. Komponentlar xususiyati qanday o'zgartiriladi?
3. C++Builder 6 tilining ifodalari qanday va ularni dasturda ishlatish tartibini ko'rsating.
4. Button1 komponentining vazifasi qanday?
5. Nima uchun bu jarayon chiziqli deb ataladi?

12-BOB. C++ BUILDER 6 DASTURLASH TIZIMIDA MANTIQUIY DASTURLASH TEXNOLOGIYASI

12.1. C++ builder 6 dasturlash tizimida shartsiz o'tish operatori

Shunday texnik masalalar mavjudki, ularning shartiga qarab ikki xil natija qabul qilinishi mumkin. Bu, o'z navbatida, dasturni tarmoqlanishga olib keladi. Tarmoqlarning qaysi qismi bajarilishi ayrim shartlarga qarab aniqlanadi. Shart operatori boshqarishni qaysi tarmoqqa uzatishni ta'minlaydi. Bunday jarayonlar tarmoqlanuvchi jarayonlar deb yuritiladi.

Tarmoqlanuvchi hisoblash jarayonlari oddiy va murakkab bo'lishi mumkin. Bu esa jarayondagi tarmoqlar soniga bog'liq. Ma'lum bir tarmoqlanuvchi jarayon tarkibida yana tarmoqlanishlar bo'lishi mumkin. Bunday tarmoqlanishlari bor bo'lgan hisoblash jarayonlari murakkab tarmoqlanuvchi hisoblash jarayonlari deb ataladi.

C++ tilida tarmoqlanuvchi jarayonlarni dasturlash uchun shartsiz, shartli o'tish va tanlash operatorlaridan foydalaniladi.

Dasturda ba'zi bir hollarda boshqaruvni to'g'ridan to'g'ri biron-bir operatorga uzatishga, ya'ni dasturning bajarilish ketma-ketligini buzishga to'g'ri keladi. Bu jarayon shartsiz o'tish operatori yordamida bajariladi.

Shartsiz o'tish operatorining umumiy ko'rinishi quyidagicha:

goto <operator belgisi>;

Bu yerda operator belgisi boshqaruv uzatiladigan operator belgisidir. Belgi sifatida 0-9999 oraliqdagi natural sonlar, CHAR toifasidagi belgilar (simvollar) va ular aralashmasidan foydalanish mumkin.

Masalan: Ikki butun musbat son: M va N larning eng katta umumiy bo'luvchisini aniqlang.

```

{int m,n,x,y;
  m=StrToInt(Edit1->Text);
  n=StrToInt(Edit2->Text);
  x=m;y=n;
  A:
  if (x=y) goto B;
  if (x>y) x=x-y;
  if (x<y) y=y-x;
  goto A;
B: cout <<"x=" <<x;
  system("PAUSE");
  return 0;
}

```

Belgili operatorida belgi bilan operator o'rtasida «:» belgisi qo'yiladi. Bu operatorning noto'g'ri qo'llanilishi dasturning bajarilishiga xalaqit beradi. Shuning uchun, dasturda bu operatorning kamroq uchrashi maqsadga muvofiqdir.

12.2. C++ builder 6 dasturlash tizimida shartli o'tish operatori

Shart – bu ikkita qiymatdan birini oladigan mantiqiy ifoda: «rost» yoki «yolg'on» ko'rinishida bo'ladi.

Dasturda boshqaruvni ma'lum shart asosida u yoki bu tarmoqqa (ma'lum jarayonlar ketma-ketligi) uzatish shartli o'tish operatori yordamida amalga oshiriladi. Shartli o'tish operatori ikki xil: to'liq va qisqa ko'rinishda ishlatilishi mumkin.

Shartli o'tish operatorining to'liq ko'rinishini ko'rib chiqamiz. Uning dasturda ko'rinishi quyidagicha yoziladi:

```
if (<mantiqiy ifoda>) <1-operator>; else <2-operator>;
```

bu yerda: if – agar, else – aks holda ma'nosini anglatuvchi xizmatchi so'zlar, 1-operator va 2-operator – ixtiyoriy operatorlar.

Operatoridagi mantiqiy ifoda boshqaruvni uzatish shartini belgilaydi.

Operatorning ishlash tartibi quyidagicha: agar keltirilgan mantiqiy ifoda true (rost) qiymatni qabul qilsa, ya'ni qo'yilgan shart bajarilsa; 1-operator bajariladi, aks holda else xizmatchi so'zdan keyingi 2-operator bajariladi.

Mantiqiy ifoda o'rnida munosabat amallari, mantiqiy amallar ishlatilishi mumkin. Masalan,

$$y > x, a = b, x < 7.35, 2 + z > 0, x + y <= 1 \text{ va h.k.}$$

Shartlar oddiy va murakkab bo'lishi mumkin.

Agar mantiqiy ifodada bitta munosabat amali berilgan bo'lsa, «oddiy shart»ni ifodalaydi.

Kattaliklar orasidagi shartlar «hamda» (&&), «yoki» (||), «emas» (!) mantiqiy amallari belgilari orqali bog'lanuvchi bir necha munosabatlardan iborat bo'lsa, murakkab shartlar deb ataladi.

Munosabat amal belgilari 12.1-jadvalda ko'rsatilgan:

12.1-jadval

Munosabat amal belgilari

Munosabat amal belgisi	Nomlanishi	Misolda ko'rinishi
=	Teng	$2=2;$
!=	Teng emas	$5!=4;$
<	Kichik	$2<3;$
>	Katta	$5>4;$
>=	Katta yoki teng	$1<=Z;$
<=	Kichik yoki teng	$1>=Z;$

Masalan, dasturda murakkab mantiqiy ifodalar (shartlar) quyidagicha yoziladi:

- $1 < x <= 4$ mantiqiy ifoda ($x > 1 \ \&\& \ x <= 4$) ko'rinishida yoziladi;
- $x = y = 0$ mantiqiy ifoda ($x = 0 \ \&\& \ y = 0$) ko'rinishida yoziladi;
- $6 <= x < 10$ mantiqiy ifoda ($x >= 6 \ \&\& \ x < 10$) ko'rinishida yoziladi.

Shartli o'tish operatorining ishlatilishini misollarda ko'rib chiqamiz.

1) if ($x > 0$) $d = \text{sqrt}(y)$; else $d = x$;

Shartli operatorlarda $x > 0$ bo'lsa $d = \sqrt{x}$ operatori, aks holda $d = x$ operatori bajariladi.

2) if ($x=0 \ \&\& \ x > 0$) $x = \sqrt{x}$; else $x = \text{pow}(x, 2)$;

Ushbu operatorning bajarilishi natijasida x ning qiymati 0 ga teng va musbat bo'lsa, uning qiymati ildiz ostidan chiqariladi, aks holda kvadratga oshiriladi.

1-misol. Berilgan $y = \begin{cases} \sqrt{b + \sqrt{b}}, & b > 10 \\ \ln|b + 1|, & b \leq 10 \end{cases}$ funksiyaning qiymati

ni aniqlash dasturi tuzilsin. Dastur konsol muhitida amalga oshirilsin.

Konsol muhitida dastur matni quyidagi ko'rinishda bo'ladi:

```
#include <stdio.h>
#include <iostream.h>
#include <conio.h>
#include <math.h>
#include <vcl.h>
#pragma hdrstop
//-----
#pragma argsused
int main(int argc, char* argv[])
{ float b, y;
  cout << «b - ni qiymatini kiriting» << «\n»; cin >> b;
  if (b > 10)
  {
    y = pow(b + sqrt(b), 1/3);
    cout << «Funksiyaning 1-sharti bajarildi, y=» << y;
  }
  else
  {
    y = log(fabs(b + 1));
    cout << «Funksiyaning 2-sharti bajarildi, y=» << y;
  }
  getch();
  return 0;
}
```


Dastur matni kiritib bo'lingandan so'ng kompilatsiya uchun F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



12.1-rasm. Dastur natijasi.

2-misol. Berilgan $y = \begin{cases} 5x^2 + 4ab, a > 1 \\ 6a - b^2x, a \leq 1 \end{cases}$ funksiyaning qiymatini

aniqlash dasturi tuzilsin. Dastur vizual muhitda amalga oshirilsin.

Ifodani vizual muhitda dasturlash uchun 4 ta Label, 2 ta Edit, 1 ta Button va 1 ta RadioGroup komponentlari kerak bo'ladi.

Forma darchasiga o'rnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

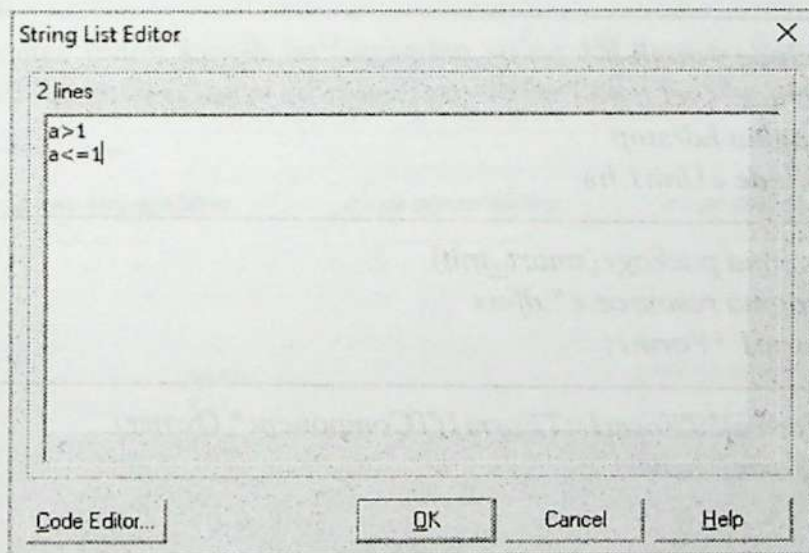
12.2-jadval

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Tarmoqlanuvchi jarayon» so'zi kiritiladi.
Label1	Caption (Properties)	«A ning qiymatini kirit» so'zi kiritiladi.
Label2	Caption (Properties)	«B ning qiymatini kirit» so'zi kiritiladi.

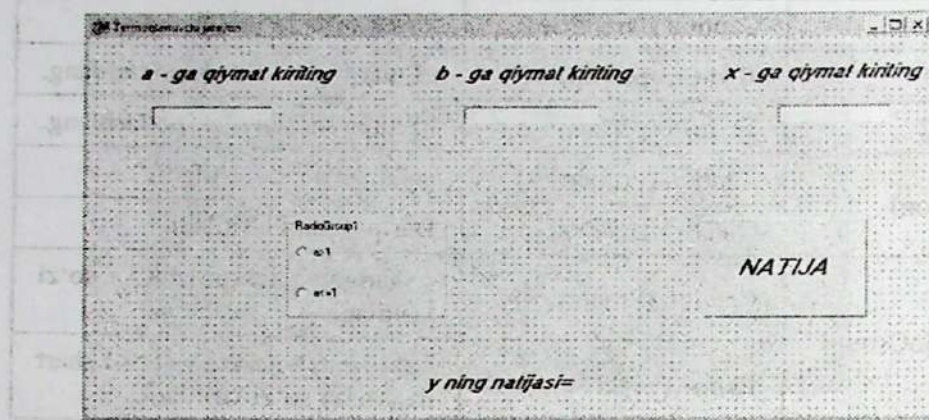
Label3	Caption (Properties)	«X ning qiymatini kirib» so'zi kiritiladi.
Label4	Caption (Properties)	«Y=» so'zi kiritiladi.
Edit1	Text (Properties)	«Edit1» so'zini o'chirib tashlang.
Edit2	Text (Properties)	«Edit2» so'zini o'chirib tashlang.
Button1	Caption (Properties)	«Natija» so'zi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.
RadioGroup1	Caption (Properties)	«Shartga tekshirish» so'zi kiritiladi.
	Items (Properties)	«a>1 shart bajarilsa va a=<1 shart bajarilsa» so'zi kiritiladi.

Forma oynasiga RadioGroup1 komponenti qo'yilgandan so'ng object inspector ilovasidan Items xossasiga kirib String List Editor oynasi ochiladi. So'ngra masala sharti kiritiladi.



12.2-rasm. Items (Properties) xossasiga shartning kiritilishi.

Mavjud komponentlar xususiyatlari kiritilgandan so'ng dastur loyihasi quyidagi ko'rinishga keladi:



12.3-rasm. Forma oynasida dastur loyihasining tuzilishi.

Dastur loyihasi tayyor bo'lganidan so'ng quyidagi dastur matni kiritiladi:

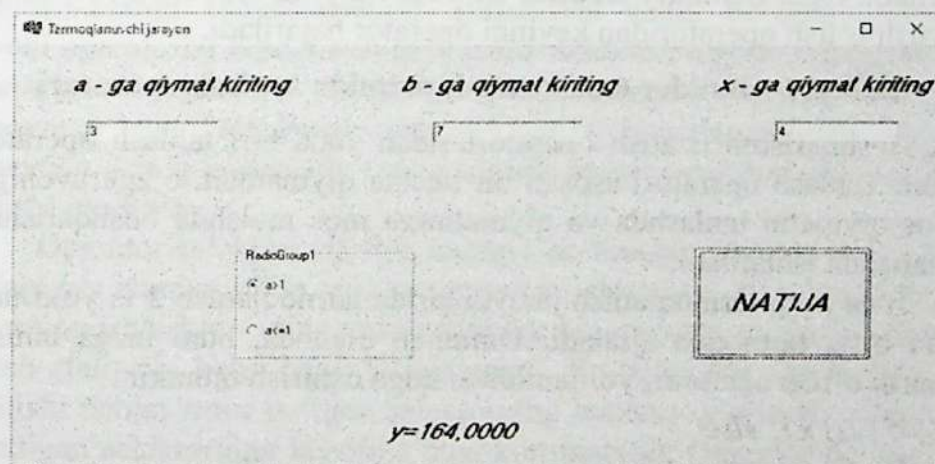
```
//-----
#include <math.h>
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
```

```

float a,b,x,y;
a=StrToFloat(Edit1->Text);
b=StrToFloat(Edit2->Text);
x=StrToFloat(Edit3->Text);
if (a>1)
{
y= 5*pow(x,2)+4*a*b;
RadioGroup1->ItemIndex=0;
Label4->Caption =«y=»+FloatToStrF(y,ffFixed,6,4);
}
else
{
y=6*a-pow(b,2)*x;
RadioGroup1->ItemIndex=1;
Label4->Caption =«y=»+FloatToStrF(y,ffFixed,6,4);
}
}
//-----

```

Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



12.4-rasm. Natija oynasi.

Ayrim algoritmlarda ba'zan shunday hol uchrash mumkinki, bunda hisoblash jarayonida ayrim amallar ba'zi bir shartlar bajarilgandagina hisoblanadi. Aks holda, hech qanday amal bajarilmaydi. Bu holda shartli o'tish operatorini qisqa ko'rinishda ifodalash mumkin. Uning yozilishi quyidagicha:

if (<mantiqiy ifoda>) <operator>/.

Operatorning bajarilish tartibi quyidagicha: agar mantiqiy ifoda TRUE (rost) qiymat qabul qilsa, operator bajariladi, aks holda IF dan keyingi turgan operator bajariladi.

Misol:

*if (x<0) t=x*x;*

Shartli o'tish operatorining metaformulasidagi operator o'rnida, o'z navbatida, yana shartli o'tish operatorining to'la va qisqa ko'rinishlari ishlatilishi mumkin. Masalan,

if (x1) {if (x2) a};

bu yerda: x1, x2 – mantiqiy ifoda, a – operator.

Bu operatorning bajarilishi natijasida x1 mantiqiy ifodaning qiymati hisoblanadi, agar TRUE qiymat qabul qilsa x2 mantiqiy ifodaning qiymati hisoblanadi, u ham rost (TRUE) bo'lsa, a – operator bajariladi. Agar mantiqiy ifodalar x1 yoki x2 yolg'on bo'lsa (FALSE), shartli o'tish operatoridan keyingi operator bajariladi.

12.3. C++ builder 6 dasturlash tizimida tanlash operatori

Boshqarishni uzatish operatorlaridan yana biri tanlash operatoridir. Tanlash operatori asosan bir nechta qiymatdan, o'zgaruvchiga mos qiymatni tanlashda va qiymatlarga mos ravishda boshqarishni uzatishda ishlatiladi.

Juda ko'p tarmoqlanish jarayonlarida tarmoqlanish 2 ta yoki undan ortiq tarmoqqa ajraladi. Umuman olganda, buni bizga tanish shartli o'tish operatori yordamida amalga oshirish mumkin:

*if (y1) x1; else
if (y2) x2; else
.....
if (b) a;*

Lekin bu hollarda shartli o'tish operatorlarining yozilishi noqulay.

Ko'p hollarda dasturchi uchun shartli operatorning umumiylashgan ko'rinishi – tanlash operatorini ishlatish qulay. Tanlash operatorining umumiy ko'rinishi quyidagicha yoziladi:

switch (ifoda yoki o'zgaruvchi – selektor)

```
{  
  case <1-qiymat>: <operator(lar)>; break;  
  case <2-qiymat>: <operator(lar)>; break;  
  case <3-qiymat>: <operator(lar)>; break;  
  case <4-qiymat>: <operator(lar)>; break;  
  ....  
  case <n-qiymat>: <operator(lar)>; break;  
  default: <aks holdagi operator(lar)>;  
}
```

bu yerda: switch (tanlash yoki boshqa holatga o'tkazuvchi) – xizmatchi so'z.

Switch operatori tarmoqlanish jarayonini berilgan bir nechta operatoridan birini tanlash yo'li bilan amalga oshiradi. Tanlash operatorida barcha operatorlar, shu jumladan bajarilishi uchun tanlangan operator ham aniq ravishda keltiriladi (berilgan operatorlar ketma-ketligi chegaralangan). Bajarilishi kerak bo'lgan operator yoki operatorlar ketma-ketligi operator selektorining qiymatiga ko'ra aniqlanadi. Agar biror variant mos kelmasa default orqali ko'rsatilgan operator bajariladi. Break operatori har bir holatdan chiqish, orqaga qaytish uchun ishlatiladi, uning o'rnida return operatori ham qo'llanishi mumkin.

Operator selektori sifatida haqiqiy bo'lmagan, skalyar ko'rinishdagi har qanday ifoda yoki o'zgaruvchi ishlatilishi mumkin. Operatorning ishlashida uning tarkibidagi har bir operator «tanlash belgisi» deb ataluvchi belgi bilan ta'minlanadi. Bu belgi operatorning bajarilishi uchun zarur bo'lgan selektorning maxsus qiymatini qabul qiladigan selektorning tavsifiga mos konstantadir. Operator bir nechta mavjud qiymatlar bilan ishlashi uchun, unda tanlash belgilari ro'yxati keltirilishi kerak.

Operator bajarilishida dastlab selektorning qiymati hisoblanadi. So'ngra selektorning qiymatiga mos belgili operator – «case» so'zidan keyin turgan qiymatga mos kelgan operator bajariladi. Agar operatorlar ketma-ketligida bunday belgili operator topilmasa, dasturda xato qayd etiladi. Shuning uchun dastur bajarilishi jarayonida selektorning qiymatiga mos keladigan maxsus belgili operator yoki operatorlar ketma-ketligida bo'lishi shart. Bunda tanlash operatorida beriladigan belgilar e'lon qilinmaydi.

Tanlash operatorining bajarilishi uning tarkibidagi operatorlar ketma-ketligidagi bitta operatorning bajarilishiga olib keladi.

Masalan:

```
#include <iostream.h>
#include<conio.h>
#include <vcl.h>
#pragma hdrstop
//-----
#pragma argsused
int main(int argc, char* argv[])
{
    int baho;
    cin>> baho;
    switch(baho)
    {
        case 2: cout <<«\n qoniqarsiz»; break;
        case 3: cout <<«\n qoniqarli»; break;
        case 4: cout <<«\n yahshi»; break;
        case 5: cout <<«\n a'lo»; break;
        default: cout <<«\n baho noto'g'ri kiritilgan»;
    }
    getch();
    return 0;
}
//-----
```

Bu operatorning bajarilishi natijasida, agar «baho» – o'zgaruvchining qiymati kiritilgandan so'ng, uning qiymati case variantlarida

keltirilgan qiymatlar bilan solishtiriladi. Agar qiymat 2 ga teng bo'lsa «Qoniqarsiz», 3 ga teng bo'lsa «Qoniqarli» va h.k. so'zlari ekranga chiqariladi, aks holda «baho noto'g'ri kiritilgan» qatori chiqadi.

1-misol. Tanlash operatorini ishlatgan holda hafta kunlarini ekranga chiqarish dasturi tuzilsin. Dastur konsol muhitida ishlansin.

Dastur matni quyidagi ko'rinishda bo'ladi:

```
//-----  
#include <iostream.h>  
#include <conio.h>  
#include <stdio.h>  
#include <vcl.h>  
#pragma hdrstop  
//-----  
#pragma argsused  
int main(int argc, char* argv[])  
{ int kun;  
  cout<<«kun - raqamini kiriting»<<«\n»;  
  cin>>kun;  
  switch (kun)  
  {  
    case 1: cout <<«Dushanba»; break;  
    case 2: cout <<«Seshanba»; break;  
    case 3: cout <<«Chorshanba»; break;  
    case 4: cout <<«Payshanba»; break;  
    case 5: cout <<«Juma»; break;  
    case 6: cout <<«Shanba»; break;  
    case 7: cout <<«Yakshanba»; break;  
    default: cout <<«n noto'g'ri kiritilgan»;  
  }  
  getch();  
  return 0;  
}  
//-----
```

Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



12.5-rasm. Natija oynasi.

Vizual muhitdan foydalangan holda misol ko'rib chiqamiz. Kiritilgan sonning juft yoki toqligini ko'rsatuvchi dastur tuzilsin.

Misolni vizual muhitda dasturlash uchun 3 ta Label, 1 ta Edit, 1 Button komponentlari kerak bo'ladi.

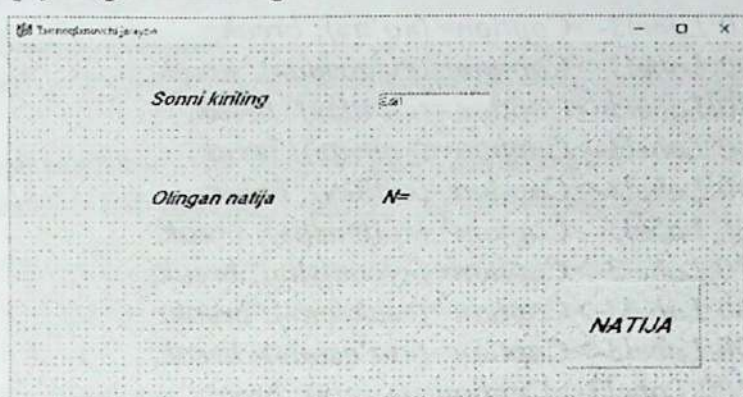
Forma darchasiga o'rnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

11.4-jadval

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Tarmoqlanuvchi jarayon» so'zi kiritiladi.
Label1	Caption (Properties)	«Son kiriting» so'zi kiritiladi.
Label2	Caption (Properties)	«Kiritilgan son:» so'zi kiritiladi.
Label3	Caption (Properties)	«Label3» so'zini o'chirib tashlang.
Edit1	Text (Properties)	«Edit1» so'zini o'chirib tashlang.
Button1	Caption (Properties)	«Natija» so'zi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.

Mavjud komponentlar xususiyatlari kiritilgandan so'ng dastur loyihasi quyidagi ko'rinishga keladi:



12.6-rasm. Forma oynasida dastur loyihasining tuzilishi.

Dastur loyihasi tayyor bo'lganidan so'ng quyidagi dastur matni kiritiladi:

```
//-----
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
int n;
n=StrToInt(Edit1->Text);
```



```
switch (n)
```

```
{
```

```
case 10:Label3->Caption=(«o'n»); break;
```

```
case 20:Label3->Caption=(«yigirma»); break;
```

```
case 30:Label3->Caption=(«o'ttiz»); break;
```

```
case 40:Label3->Caption=(«qiriq»); break;
```

```
case 50:Label3->Caption=(«ellik»); break;
```

```
case 60:Label3->Caption=(«oltmish»); break;
```

```
case 70:Label3->Caption=(«yetmish»); break;
```

```
case 80:Label3->Caption=(«sakson»); break;
```

```
case 90:Label3->Caption=(«to'qson»); break;
```

```
case 100:Label3->Caption=(«yuz»); break;
```

```
default: Label3->Caption=(«Bunday son yo'q»);
```

```
}
```

```
}
```

```
//-----
```

Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:

Sonni kiriting 80

Olingan natija sakson

NATIJJA

12.7-rasm. Natija oynasi.

Nazorat savollari

1. C++ Builder dasturlash tilida qanday jarayonlarga tarmoqlanuvchi jarayonlar deyiladi?
2. Dastur loyihasi ishga tushirilgandan so'ng qaysi tugma bosiladi?
3. Case operatorining vazifasi nimadan iborat?
4. Shartli o'tish operatori qanday holatlarda ishlatiladi?

13-BOB. C++ BUILDER 6 DASTURLASH TIZIMIDA TAKRORLANUVCHI STRUKTURALI DASTURLAR

13.1. Takrorlash operatorlari

Ko'pincha masalalarni dasturlashda bir xil buyruqlar ketma-ketligi bir necha marta bajarilishi kerak bo'ladi. Bunday jarayon takrorlanuvchi hisoblash jarayoni (sikl) deb ataladi.

Takrorlanuvchi hisoblash jarayonining takrorlanib hisoblanadigan qismi takrorlanishning tanasi deb ataladi.

Takrorlanish ichida qiymatlari o'zgarib boradigan o'zgaruvchi takrorlanish o'zgaruvchisi yoki takrorlanishning boshqaruvchi o'zgaruvchisi (sikl parametri) deb yuritiladi.

Takrorlanuvchi jarayonning algoritmi umumiy holda quyidagilarni o'z ichiga olishi kerak:

1) takrorlanishni tayyorlash – takrorlanishni boshlashdan oldin takrorlanishda qatnashadigan o'zgaruvchilarning boshlang'ich qiymatlari yoki takrorlanish o'zgaruvchisining boshlang'ich qiymati o'rnatiladi, takrorlanish o'zgaruvchisining o'zgarish qadami belgilanadi;

2) takrorlanish tanasi – takrorlanish o'zgaruvchilarining turli qiymatlari uchun takror bajariladigan amallar ketma-ketligi ko'rsatiladi;

3) takrorlanish o'zgaruvchisiga yangi qiymat berish – har bir takrorlanishdan avval o'zgaruvchiga o'zgarish qadamiga mos ravishda yangi qiymat beriladi;

4) takrorlanishni boshqarish – takrorlanishni davom ettirish sharti tekshiriladi, takrorlanishning boshiga o'tish ko'rsatiladi.

C++ algoritmik tilida uch xil ko'rinishda takrorlanuvchi hisoblash jarayonini tashkil qilish mumkin va bu jarayonlarni dasturlash uchun maxsus operatorlar belgilangan:

– sharti avval tekshiriladigan (sharti avval kelgan «toki») takrorlanish jarayoni, bu maxsus «While» operatori orqali amalga oshiriladi;

- sharti keyin tekshiriladigan («...gacha») takrorlanish jarayoni: bu jarayon maxsus «Do while» operatori orqali amalga oshiriladi;
- parametrli takrorlanish jarayoni, bu jarayon maxsus «For» operatori yordamida amalga oshiriladi.

Har bir takrorlanish operatorining o'ziga xos qo'llanilish xususiyatlari bor. Yuqoridagi har qanday operatorni boshqasiga almashtirish mumkin.

13.2. While operatorining qo'llanilishi

Sharti avval tekshiriladigan takrorlanish jarayoni takrorlanish soni oldindan noma'lum bo'lgan hollarda, ya'ni takrorlanishdan chiqish ma'lum shartga bog'liq bo'lgan hollarda ishlatiladi. Takrorlanishning bu jarayonida takrorlanishdan chiqish sharti takrorlanish tanasini bajarishdan oldin tekshiriladi.

Ushbu operatorning umumiy ko'rinishi quyidagichadir:

<Sharti avval tekshiriladigan takrorlanish operatori> (mantiqiy ifoda) <operatorlar>;

yoki while (m) l;

bu yerda: while – toki, «m» – mantiqiy ifoda, «l» – operatorlar yoki operatorlar guruhi bo'lib, u takrorlanish tanasini belgilaydi. Takrorlanish tanasida bitta yoki bir nechta operatorlar guruhi bo'lishi mumkin. Bunda operatorlar guruhi albatta {va} orasida yozilishi kerak.

Operatorning bajarilishi quyidagicha:

«m» mantiqiy ifodaning qiymati hisoblanadi. Agar «m» mantiqiy ifoda rost qiymatga ega bo'lsa, «l» operatori bajariladi va bu operator «m» mantiqiy ifodaning qiymati yolg'on bo'lgungacha qayta-qayta bajariladi.

Agar «m» mantiqiy ifodaning qiymati birinchi tekshirishdayoq yolg'on bo'lsa, «l» operatori biron marta ham bajarilmaydi va boshqaruv while operatoridan keyingi operatorga uzatiladi.

Agar «m» mantiqiy ifoda rost bo'lib, kompyuter «l» operatorini bajarish davomida, biror sababga ko'ra takrorlanishdan chiqish talab etilsa, unda break operatori orqali amalga oshiriladi.

1-misol. $y = \sqrt{2x-1}$ funksiyaning qiymatini hisoblash dasturi tuzilsin. X o'zgaruvchisining qiymatlari a dan b gacha h qadam bilan o'zgarsin.

Dasturning konsol rejimidagi ko'rinishi quyidagicha:

```
//-----
```

```
#include <math.h>
```

```
#include<iostream.h>
```

```
#include<conio.h>
```

```
#include<stdio.h>
```

```
#include <vcl.h>
```

```
#pragma hdrstop
```

```
//-----
```

```
int main(int argc, char* argv[])
```

```
{
```

```
float a, b, h, y, x;
```

```
cout<<«a, b, h ning qiymatlarini kiriting» <<endl;
```

```
cin>>a>>b>>h;
```

```
x=a;
```

```
while (x<=b)
```

```
{
```

```
y=sqrt(2*x-1);
```

```
cout<<«x= »<<x<<« y= »<<y<< endl;
```

```
x=x+h;
```

```
}
```

```
getch();
```

```
return 0;
```

```
}
```

```
//-----
```

Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



13.1-rasm. Natija oynasi.

2-misol. $y=10^6 \cdot \ln(x+1)$ funksiyaning qiymatini hisoblash dasturi tuzilsin. X o'zgaruvchisining qiymatlari a dan b gacha h qadam bilan o'zgarsin. Dastur vizual muhitda amalga oshirilsin.

Ifodani vizual muhitda dasturlash uchun 3 ta Label, 3 ta Edit, 1 ta Button va 1 ta Memo komponentlari kerak bo'ladi.

Forma darchasiga o'rnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

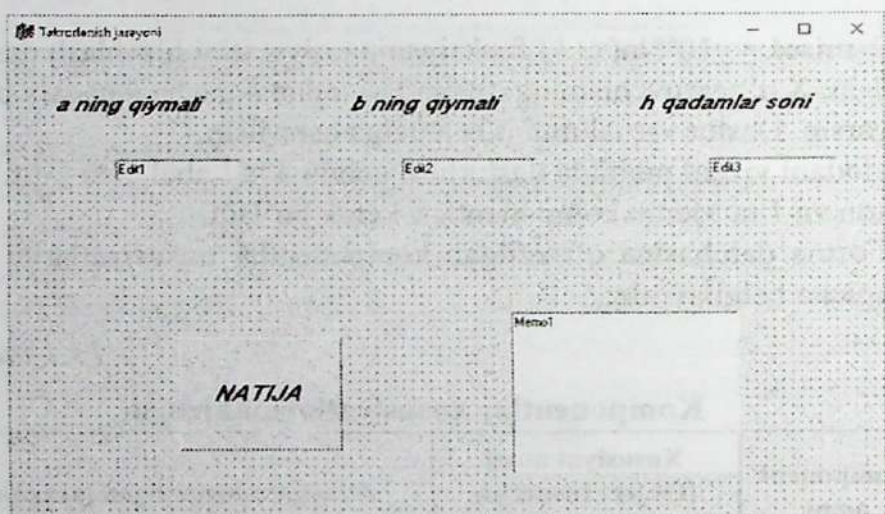
13.1-jadval

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Takrorlanish jarayoni» so'zi kiritiladi.
Label1	Caption (Properties)	«a ning boshlang'ich qiymati» so'zi kiritiladi.
Label2	Caption (Properties)	«b ning so'nggi qiymati» so'zi kiritiladi.
Label3	Caption (Properties)	«h qadamlar soni» so'zi kiritiladi.

Edit1	Text (Properties)	«Edit1» so'zi o'chirib tashlanadi.
Edit2	Text (Properties)	«Edit2» so'zi o'chirib tashlanadi.
Edit3	Text (Properties)	«Edit3» so'zi o'chirib tashlanadi.
Button1	Caption (Properties)	«Natija» so'zi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.
Memo1	Lines (Properties)	«Memo1» so'zi o'chirib tashlanadi.

Mavjud komponentlar xususiyatlari kiritilgandan so'ng dastur loyihasi quyidagi ko'rinishga keladi:



13.2-rasm. Forma oynasida dastur loyihasining tuzilishi.

Dastur loyihasi tayyor bo'lganidan so'ng quyidagi dastur matni kiritiladi:

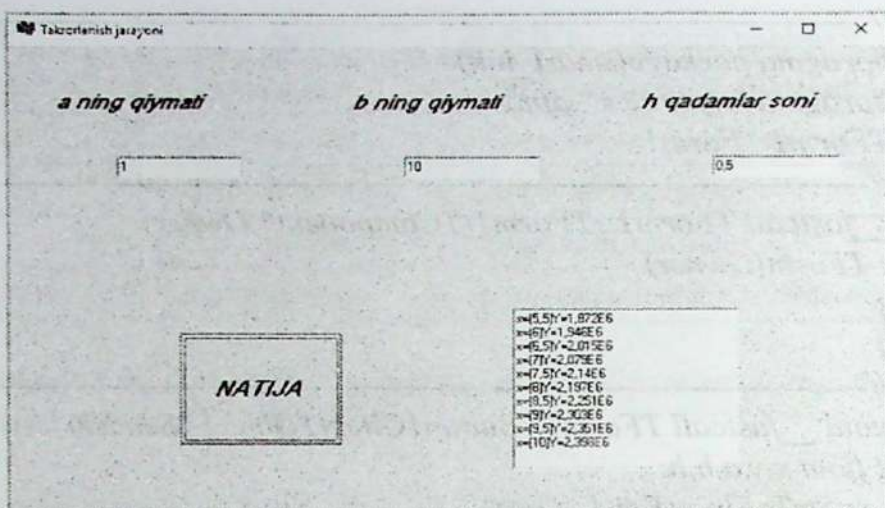
```
//-----
#include <math.h>
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
```

```

//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{ float x,y,a,b,h;
  a=StrToFloat(Edit1->Text);
  b=StrToFloat(Edit2->Text);
  h=StrToFloat(Edit3->Text);
  x=a;
  while (a<=b)
  {
    y=pow(10,6)*log(x+1);
    Memo1->Lines->Add(«x=(«+FloatToStr(a)+»)Y=«+FloatTo-
StrF(y, ffFixed,4,4));
    x=x+h;
    a=a+h;
  }
}
//-----

```

Dastur matni kiritib bo‘lingandan so‘ng F9 tugmasi bosiladi va natijada quyidagi dastur ko‘rinishi paydo bo‘ladi:



13.3-rasm. Natija oynasi.

13.3. Do...while operatorining qo'llanilishi

Sharti keyin tekshiriladigan takrorlanish jarayoni ham takrorlanish soni oldindan ma'lum bo'lmagan hollarda ishlatiladi. Bunday jarayonda biror buyruq yoki buyruqlar tizimi berilgan shart bajarilgunga qadar takror-takror bajariladi. Bu jarayonning oldin sharti tekshiriladigan takrorlanish jarayonidan farqi shundaki, bunda takrorlanishning tanasi hech bo'lmaganda bir marta bo'lsa ham bajariladi, chunki takrorlanishdan chiqish sharti takrorlanishning tanasi bajarilgandan keyin tekshiriladi.

Bunday takrorlanish jarayonini dasturlashda maxsus «do while» operatoridan foydalaniladi.

Ushbu operatorning umumiy ko'rinishi quyidagicha:

<Sharti keyin tekshiriladigan takrorlanish operatori> = do <operatorlar guruhi> while <mantiqiy ifoda>;

yoki do M while (N);

bu yerda: do – bajarmoq, while – «toki» ma'nosini anglatuvchi xizmatchi so'zlar; M – takrorlanish tanasi, uning tarkibida bitta operator yoki operatorlar guruhi bo'lishi mumkin; N – mantiqiy ifoda.

Operator quyidagi tartibda bajariladi:

– takrorlanish tanasiga kirgan operatorlar birin-ketin bajariladi. So‘ng M mantiqiy ifodaning yolg‘on qiymati topiladi, ya‘ni shart tekshiriladi. Agar bu shart bajarilmasa (M ning qiymati False (yolg‘on) bo‘lsa) boshqaruv takrorlashdan tashqariga, while so‘zidan keyingi operatorga uzatiladi. Aks holda, takrorlanish davom etadi.

Takrorlanish jarayonining bunday ko‘rinishi yuqorida keltirilgan takrorlanishdan shu bilan farq qiladiki, bunda takrorlanish tanasi hech bo‘lmaganda bir marta bajariladi.

1-misol. $y=(1+x)*e^{-x}$ funksiyaning qiymatini hisoblash dasturi tuzilsin. X o‘zgaruvchisining qiymatlari a dan b gacha h qadam bilan o‘zgarsin.

Dasturning konsol rejimidagi ko‘rinishi quyidagicha:

```
#include <vcl.h>
```

```
#include <iostream.h>
```

```
#include <math.h>
```

```
#include <conio.h>
```

```
#pragma hdrstop
```

```
//-----
```

```
#pragma argsused
```

```
int main(in
```

```
argc, char* argv[])
```

```
{
```

```
float a,b,x,y,h;
```

```
cout<<«a=«;
```

```
cin>>a;
```

```
cout<<«b=«;
```

```
cin>>b;
```

```
cout<<«h=«;
```

```
cin>>h;
```

```
x=a;
```

```
do
```

```
{
```

```
y=(1+x)*exp(-x);
```



```

cout << «x=» << x << endl;
cout << «y=» << y << endl;
x=x+h;
}
while(x<=b);
getch();
return 0;
}

//-----

```

Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



13.4-rasm. Natija oynasi.

2-misol. $y=x^2+\sin(x)$ funksiyaning qiymatini hisoblash dasturi tuzilsin. X o'zgaruvchisining qiymatlari a dan b gacha h qadam bilan o'zgarsin. Dastur vizual muhitda amalga oshirilsin.

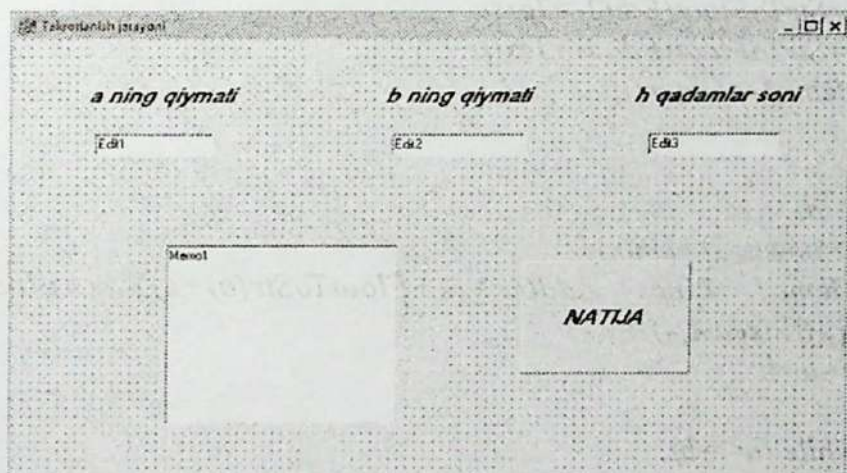
Ifodani vizual muhitda dasturlash uchun 3 ta Label, 3 ta Edit, 1 ta Button va 1 ta Memo komponentlari kerak bo'ladi.

Komponentlar xususiyatlarini quyidagicha belgilaymiz:

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Takrorlanish jarayoni» soʻzi kiritiladi.
Label1	Caption (Properties)	«a ning qiymati» soʻzi kiritiladi.
Label2	Caption (Properties)	«b ning qiymati» soʻzi kiritiladi.
Label3	Caption (Properties)	«h qadamlar soni» soʻzi kiritiladi.
Edit1	Text (Properties)	«Edit1» soʻzi oʻchirib tashlanadi.
Edit2	Text (Properties)	«Edit2» soʻzi oʻchirib tashlanadi.
Edit3	Text (Properties)	«Edit3» soʻzi oʻchirib tashlanadi.
Button1	Caption (Properties)	«Natija» soʻzi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.

Mavjud komponentlar xususiyatlari kiritilgandan soʻng dastur loyihasi quyidagi koʻrinishga keladi:

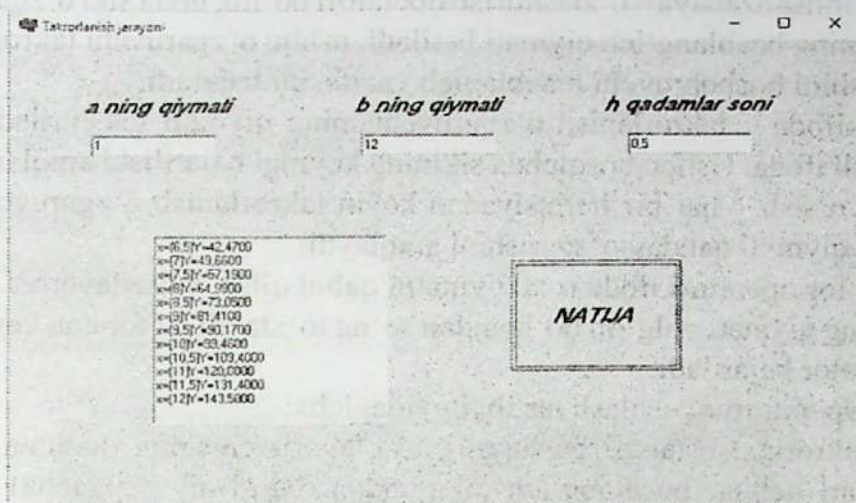


13.5-rasm. Forma oynasida dastur loyihasining tuzilishi.

Dastur loyihasi tayyor bo'lganidan so'ng quyidagi dastur matni kiritiladi:

```
//-----
#include <math.h>
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
float a,b,h,y,x;
a=StrToFloat(Edit1->Text);
b=StrToFloat(Edit2->Text);
h=StrToFloat(Edit3->Text);
Memo1->Clear();
do
{
x=a;
y=pow(x,2)+sin(x);
Memo1->Lines->Add(«x=(«+FloatToStr(a)+»)Y=«+FloatTo-
StrF(y,ffFixed,4,4));
a=a+h;
}
while (a<=b);
}
//-----
```

Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



13.6-rasm. Natija oynasi.

13.4. For operatorining qo'llanilishi

Yuqorida keltirilgan takrorlanish operatorlarini odatda takrorlanish soni noma'lum bo'lgan hollarda ishlatish maqsadga muvofiqdir. Takrorlanish jarayonining takrorlanish soni uning bajarilishidan oldin ma'lum bo'lsa, parametrli takrorlanish operatoridan foydalanish qulay hisoblanadi.

Bu operatorning dasturdagi umumiy yozilish konstruksiyasi quyidagi ko'rinishga ega: <parametrli takrorlanish operatori> = for (<boshqaruvchi o'zgaruvchining boshlang'ich qiymati, takrorlanish sharti, boshqaruvchi o'zgaruvchining o'zgarishi>) {<operatorlar guruhi>;

```
for (initsializatsiya; ifoda; o'sish)
```

```
{
```

```
// operatorlar ketma-ketligi
```

```
// ...
```


}

Bu yerda:

- Initsalizatsiya – o‘zlashtirish operatori bo‘lib, unda sikl o‘zgaruvchisining boshlang‘ich qiymati beriladi, ushbu o‘zgaruvchi takrorlanish ishini boshqaruvchi hisoblagich vazifasini bajaradi;

- ifoda – takrorlanish o‘zgaruvchisining qiymati tekshiriladigan shartli ifoda. Ushbu bosqichda siklning keyingi bajarilishi aniqlanadi;

- o‘sish – har bir iteratsiyadan keyin takrorlanish o‘zgaruvchisining qiymati qanday o‘zgarishini aniqlaydi;

- for operatori ifoda rost qiymatni qabul qilsa, bajarilaveradi. Ifodaning qiymati yolg‘on bo‘lgandan so‘ng to‘xtaydi va for dan keyingi operator bajariladi.

Operatorning ishlash tartibi quyidagicha:

takrorlanish tanasi boshqaruvchi o‘zgaruvchisining hamma qiymatlari uchun (boshlang‘ich qiymatidan oxirgi qiymatigacha) qayta-qayta takrorlanadi. Masalan, bu operatorlarni quyidagi ko‘rinishlarda yozish mumkin:

1) for (x=b; x<=a; x--) y=n;

2) for (x=a; x<=b; x++) y=n;

3) for (int i=1, s=0; i<=30; i++) s += i;

4) for (x=y+d; x<=z*k; x++) y=b;

1-misol. $S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$ ifodani hisoblovchi dastur tuzilsin, bu yerda n – natural son. Dastur konsol rejimda bajarilsin.

Dasturning konsol rejimidagi ko‘rinishi quyidagicha:

```
#include <math.h>
```

```
#include <conio.h>
```

```
#include <stdio.h>
```

```
#include <iostream.h>
```

```
#include <vcl.h>
```

```
#pragma hdrstop
```

```
//-----
```

```
#pragma argsused
```

```
int main(int argc, char* argv[])
```

```

{ int n,i;
float s = 0;
s=5;
for (i=1; i<=n; i++){
s=s+1.0/i;
cout<<"s="<<s<<endl;
}
getch();
return 0;
}
//-----

```



13.7-rasm. Natija oynasi.

2-misol. $y = \cos(2x+1)$ funksiyaning qiymatini hisoblash dasturi tuzilsin. X o'zgaruvchisining qiymatlari a dan b gacha 1 qadam bilan o'zgarsin. Dastur vizual muhitda amalga oshirilsin.

Ifodani vizual muhitda dasturlash uchun 2 ta Label, 2 ta Edit, 1 ta Button va 1 Memo komponentlari kerak bo'ladi.

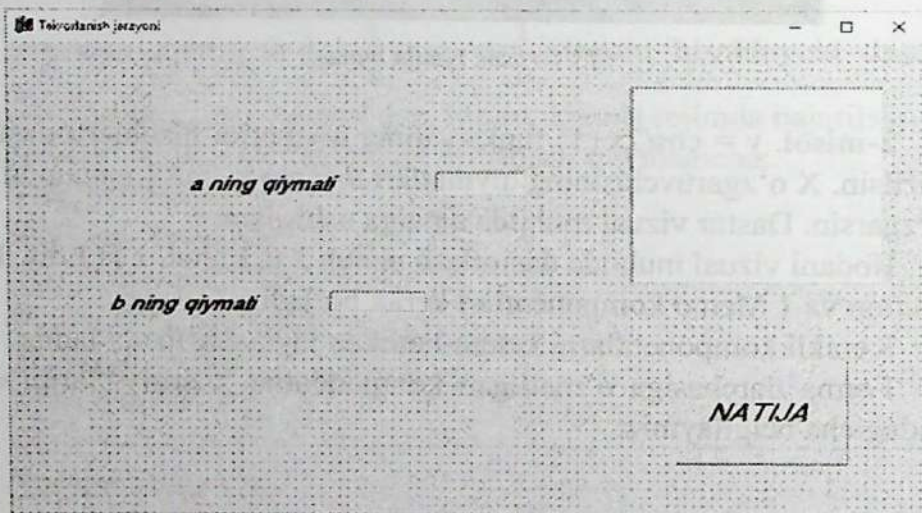
Kerakli komponentlarni ketma-ketlikda joylashtirib chiqamiz.

Forma darchasiga o'rnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Takrorlanish jarayoni» soʻzi kiritiladi.
Label1	Caption (Properties)	«a ning qiymati» soʻzi kiritiladi.
Label2	Caption (Properties)	«b ning qiymati» soʻzi kiritiladi.
Edit1	Text (Properties)	«Edit1» soʻzi oʻchirib tashlanadi.
Edit2	Text (Properties)	«Edit2» soʻzi oʻchirib tashlanadi.
Button1	Caption (Properties)	«Natija» soʻzi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.
Memo1	Lines (Properties)	«Memo1» soʻzi oʻchirib tashlanadi.

Mavjud komponentlar xususiyatlari kiritilgandan soʻng dastur loyihasi quyidagi koʻrinishga keladi:



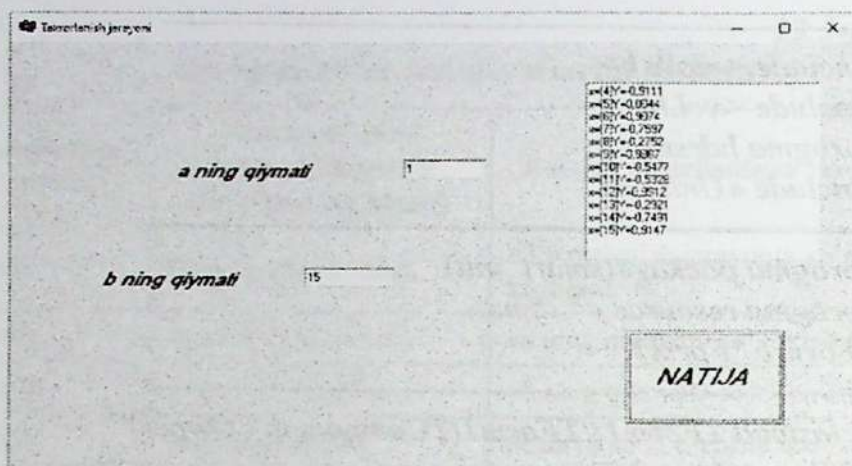
13.8-rasm. Forma oynasida dastur loyihasining tuzilishi.

```

//-----
#include <math.h>
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
float x,y;
int a,b;
a=StrToFloat(Edit1->Text);
b=StrToFloat(Edit2->Text);
Memo1->Clear();
for (x=a; x<=b; x++)
{
y = cos(2*x+1);
Memo1->Lines->Add(«x=(«+FloatToStr(x)+»)Y=«+ FloatTo-
StrF(y,ffFixed, 4,4));
}
}
//-----

```

Dastur matni kiritib bo‘lingandan so‘ng F9 tugmasi bosiladi va natijada quyidagi dastur ko‘rinishi paydo bo‘ladi:



13.9-rasm. Natija oynasi.

Nazorat savollari

1. Takrorlanuvchi jarayon deb nimaga aytiladi?
2. Takrorlanuvchi jarayonning necha turi mavjud?
3. Parametrli takrorlash jarayoni qanday tashkil etiladi?
4. Avval shartli takrorlash jarayoni qanday tashkil etiladi?
5. So'ng shartli takrorlash jarayoni qanday tashkil etiladi?

14-BOB. GRAFIK MODULINING IMKONIYATLARI VA ULARDAN FOYDALANISH

14.1. Tasvirlar bilan ishlash

Borland C ++ Builder 6 ma'lum bir komponentlar va sinflarga ega bo'lib, nafaqat grafik primitivlarni chizish, balki tasvirlar, shriftlar va grafik fayllar bilan ishlash imkonini beradi. Borland C ++ Builder 6 umumiy prinsiplarga asoslangan holda Windows operatsion tizimida grafik obyektlarni yaratadi. Windows operatsion tizimi dasturchiga C ++ Builderning to'liq grafik funksiyalaridan foydalanish va grafik qurilma interfeysidan ochish imkonini beradi.

C ++ Builder 6 muhitida grafika bilan bog'liq uch turdagi obyektlar mavjud:

1) **Kanva** – grafik tasvirni yaratishda ishlatilishi mumkin bo'lgan dastur oynasi, komponent, printer va boshqalar bilan ishlash imkonini beradi. Kanva mustaqil obyekt emas, u har doim boshqa grafik obyektga tegishli bo'ladi;

2) **Grafika** – har qanday fayl yoki manbaning rastarli tasvirini ko'rsatadi (tasvir obrazi, ikonka yoki metafile). C ++ Builder TGraphic asosiy sinfidan olingan TBitmap, TIcon va TMetafile obyekt sinflarini aniqlaydi. Albatta, siz o'zingizning grafik obyektlar sinflarini e'lon qilishingiz mumkin. TGraphic sizning ilovangizdagi barcha turdagi grafikalar uchun minimal standart interfeysni taqdim etadi;

3) **Rasm** (TPicture) – bu grafik obyektlarning har qanday sinflarini o'z ichiga oladigan grafik komponent. TPicture klasi tasvir obrazi, ikonka, metafile yoki boshqa foydalanuvchi tomonidan belgilanadigan grafik turini o'z ichiga oladi va dastur rasmi obyekt orqali barcha komponent obyektlarini standartlashtiradi. Tasvirni boshqarish komponentlarining ko'pchiligi TPicture obyekt turidagi Picture xususiyatiga

tiga ega, bu esa har xil turdagi grafik tasvirlarni taqdim etish imkonini beradi.

Borland C ++ Builder dasturida grafik :: TBitmap sifatida belgilangan TBitmap klassi, grafik tasvirlarni yaratishga imkon beradigan xususiyatlar va usullarga ega. Ularning atributlarini boshqarish, xotirada o'qish va diskda saqlash vazifalarini bajaradi.

TIcon va TMetafile sinflari ham bazadan olingan TGraphic sinfiga mansub. Grafikalar bilan ishlashning xususiyatlari va usullariga ega, lekin ularning TBitmapdan farqi, ular Canvas xususiyatiga ega emas, bu formatlari *.ico va *.wmf (* .emf) tasvirlarni yaratish va ishlatish xususiyatlari bilan chegaralanadi.

TPicture klassi TImage sinfiga inkapsulatsiya qilingan. TImage TPicturening asosiy komponenti sifatida, ikonkalar bilan ishlash qobiliyatiga ega (bmp fayllari va metafayllari). Ba'zida boshqa qo'shimcha formatdagi fayllar (*.jpeg, *.jpg, *.gif) bilan ishlashi ham mumkin.

14.2. Dasturning asosiy grafik funksiyalari

Tasvirlarni hosil qilish uchun quyidagi protsedura va funksiyalar ishlatiladi:

putpixel (x, y, color) – x va y koordinatadagi nuqtani color rangda chizish;

getpixel (x, y) – x va y koordinatadagi nuqtaning rangini aniqlaydi;

line (x1, y1, x2, y2) – x1 va y1 koordinatadagi nuqtadan x2 va y2 koordinatadagi nuqttagacha kesma chizish;

circle (x, y, r) – markazi x va y koordinatada va radiusi R bo'lgan aylana chizish;

rectangle (x1, y1, x2, y2) – yuqori chap nuqtasi x1 va y1 koordinatada, o'ng pastki nuqtasi x2 va y2 koordinatada bo'lgan to'g'ri to'rtburchakni chizish;

setbkcolor (color) – orqa fonga rang berish;

setcolor (color) – chizish rangini o'rnatish (rangli qalam); bu yerda color – rang nomeri yoki nomi. Agar rang nomi yoziladigan bo'lsa, uni katta harflarda yoziladi.

bar (x1, y1, x2, y2) – joriy rang va chiziqlar yordamida ichi bo‘yalgan to‘g‘ri to‘rtburchak chizish;

fillellipse (x, y, xr, yr) – markazi x va y da, xr kenglikda va yr ba-landlikda ichi bo‘yalgan rangli ellips chizadi;

setfillstyle (style, color) – bo‘yash usul va rangini o‘rnatish. Bu yerda style – o‘zgarmas kattalik bo‘lib, u quyidagicha bo‘lishi mumkin:

- 0 – sohani fon rangi bilan to‘ldirish;
- 1 – sohani rang bilan uzluksiz to‘ldirish;
- 2 – qalin gorizontaal chiziqlar;
- 3 – ingichka og‘ma chiziqlar;
- 4 – yo‘g‘on og‘ma chiziqlar;
- 5 – yo‘g‘on og‘ma chiziqlar (boshqa turi);
- 6 – og‘ma yo‘llar;
- 7 – to‘rtburchakli chiziqlar;
- 8 – og‘ma to‘rtburchaklar;
- 9 – zich og‘ma shtrixlar;
- 10 – siyrak nuqtalar (u yer-bu yerda);
- 11 – zich nuqtalar bilan.

getmaxx – joriy rejim va drayverlar uchun nuqtalar sonini aniqlash; **getmaxy** – joriy rejim va drayverlar uchun vertikal nuqtalar soni. Bu protsedura yordamida kompyuterning o‘zi ekrandagi maksimal nuqtalar sonini aniqlaydi.

linerel (x, y) – x va y koordinatali nuqtadan joriy nuqttagacha kesma chizish;

lineto (x, y) – joriy nuqtadan x va y koordinatali nuqttagacha kesma chizish.

bar3D (x1, y1, x2, y2, h, top) – parallelepiped chizadi. Bu yerda h – parallelepipedning uzunligi; top – yuqori qismini chizish uchun kerak. Agar topon bo‘lsa – tomi bor, agar topoff bo‘lsa – tomi yo‘q.

arc (x, y, a, b, r) – yoy chizish uchun. Bu yerda x va y – markazning koordinatalari, a – bosh burchak, b – oxirgi burchak, r – yoy radiusi. Burchaklar gradusda qabul qilinadi.

ellipse (x, y, a, b, xr, yr) – xuddi shu tartibda ellips yoyini chizadi.

drawpoly (n, p) – ko'pburchak chizish uchun. Bu yerda n – ko'pburchakning uchlari soni; p – ko'pburchak uchlarining koordinatalari.

floodfill (x, y, color) – joriy rang va usuldan foydalangan holda chegaralangan sohani bo'yash. Bu yerda x va y – shu sohaga tegishli bo'lgan biror nuqta koordinatasi. Avval rang turi, keyin chizmalar ko'rsatiladi. Masalan:

setcolor (4); {qizil rangli qalam, chegara rangi}

setfillstyle (1, 2); {1-tur bilan yashil rang bilan bo'yash}

circle (50, 50, 35); {radiusi 35 bo'lgan aylana chizish}

floodfill (50, 50, 4); {aylana ichiga rang bilan to'ldirish, bo'yaladigan chegara rangi rangli qalam bilan bir xil bo'lishi kerak}

setlinestyle (s, a, b) – turli turdagi chiziqlarni chizish uchun; bu yerda s- style (tur) nomeri; a – foydalanuvchi stilini yaratishi mumkin bo'lgan parametr, odatda a=1 deb olinadi; b – chiziqning qalinligini ko'rsatadigan parameter.

0 – oddiy chiziq;

1 – mayda punktir chiziq;

2 – qalin va uzunchoq punktir chiziq;

3 – yupqa va uzunchoq punktir chiziq;

4 – siyrak nuqtali chiziq.

Dastur grafik tasvirlarni Forma va Image komponentiga chizish imkonini beradi. Obyektlar Canvas xususiyati orqali beriladi. Chizmani (to'g'ri chiziq, to'rtburchak, aylana va h.k.) chizish uchun Canvas xususiyatiga komponentni qo'shib qo'yish kerak bo'ladi. Umumiy ko'rinishi quyidagicha bo'ladi: bu yerda:

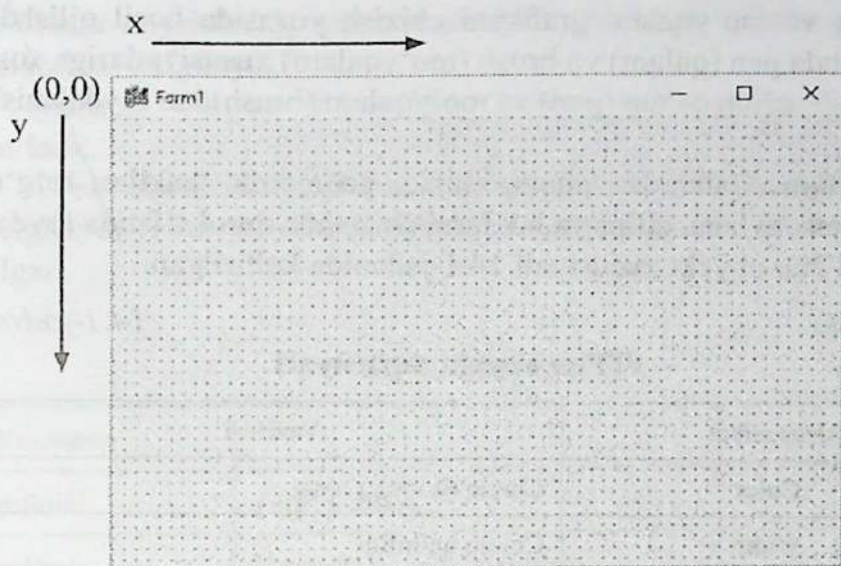
– chizma chiziladigan joy;

– chizilayotgan grafik tasvir.

Masalan:

Form1.Canvas.Rectangle (10,10,100,100);

C++da koordinatalar sistemasi quyidagi ko'rinishda belgilanadi (14.1-rasm).



14.1-rasm. Koordinatalar nuqtalarini shakllantirish.

14.3. Grafik tasvirlar yaratishda mo'yqalam bilan ishlash

Borland C++ Builderning grafik imkoniyatlari ham qalam va mo'yqalamdan foydalanish imkoniyatlarini yaratadi. Qalamdan chiziq va kontur chizishda, mo'yqalamdan esa kontur bilan chegaralangan yuzani bo'yash uchun foydalaniladi.

Ranglar kodi

Kod	Nomi	Kod	Nomi
0	BLACK	8	DARKGRAY
1	BLUE	9	LIGHTBLUE
2	GREEN	10	LIGHTGREEN
3	CYAN	11	LIGHTCYAN
4	RED	12	LIGHTRED
5	MAGENTA	13	LIGHTMAGENTA
6	BROWN	14	YELLOW
7	LIGHTGRAY	15	WHITE

14.2-rasm. Ranglarning raqamlarda ifodalanishi.

Qalam va mo'yqalam grafikani chizish yuzasida hosil qilishda mos ravishda pen (qalam) va brush (mo'yqalam) xususiyatlariga xosdir. Chizish uchun qalam (pen) va mo'yqalam (brush)dan foydalanish mumkin.

1. Qalam. Qalamdan nuqta, chiziq, geometrik shakllar: to'g'ri to'rtburchak, aylana, ellips va h.k.larni chizishda qurol sifatida foydalaniladi. TPen obyekt xususiyati 14.1-jadvalda keltirilgan.

14.1-jadval

TPen obyekt xususiyati

Xususiyat	Vazifasi
Color	Chiziq (kontur) rangi
Width	Chiziq qalinligi
Style	Chiziq ko'rinishi
Mode	Tasvirlash rejimi

TPen obyektining Color xususiyati chizuvchi qalam rangini belgilaydi. Quyidagi 14.2-jadvalda PenColor xususiyatlari keltirilgan:

14.2-jadval

PenColor xususiyati

Konstanta	Rang	Konstanta	Rang
clBlack	qora	clSilver	kumushrang
clMaroon	kashtanrang	clRed	qizil
clGreen	yashil	clLime	och yashil
clFuchsia	pushti	clBlue	ko'k
clNavy	to'q ko'k	clYellow	sariq
clPurple	atirgulrang	clMedGray	o'rta kulrang
clGray	kulrang	clWhite	oq

TPen obyektining width xususiyati chizuvchi qalam qalinligini (pikselda) belgilaydi.

Masalan, Canvas->Pen->Width2 chiziq qalinligi 2 pikselga teng bo'ladi.

TPen obyektining style xususiyati chiziluvchi chiziqning turini belgilaydi. TPen obyektining style xususiyatlari 14.3-jadvalda keltirilgan.

14.3-jadval

Style xususiyatlari

Konstanta		Ko'rinishi
psSolid	To'g'ri chiziq	
psDash	Uzun shtrixli punktir chiziq	
psDot	Qisqa shtrixli punktir chiziq	
psDashDot	Uzun-qisqa shtrixli punktir chiziq	
PsDashDotDot	Bir uzun va ikki qisqa shtrixli punktir chiziq	
PsClear	Ko'rinmas chiziq	

2. Mo'yqalam. Mo'yqalamdan (Brush) yopiq sohalarni to'ldirish uchun foydalaniladi, masalan, geometrik shakllarni bo'yash va h.k. Mo'yqalam obyekt sifatida quyidagi ikki xususiyatni o'z ichiga oladi:

- Color – bo'yaluvchi soha rangi;
- Style – to'ldiruvchi soha ko'rinishi.

Masalan, konturning ichki sohasi bo'yalishi yoki shtrixlanishi mumkin. Color xususiyati sifatida barcha o'zgarmaslaridan foydalanish mumkin.

Style xususiyatlari 14.4-jadvalda keltirilgan.

Style xususiyati

Konstanta	Bo'yash turi
bsSolid	yaxlit bo'yash
bsClear	soha bo'yalmaydi
bsHorizontal	gorizontal shtrixlash
bsVertical	vertikal shtrixlash
bsFDiagonal	oldinga egilish bilan diagonal shtrixlash
bsBDiagonal	orqaga egilish bilan diagonal shtrixlash
bsCross	gorizontal-vertikal shtrixlash, to'r ko'rinishida
bsDiagCross	diagonal shtrixlash, to'r ko'rinishida

1-misol. Mo'yqalamning barcha xususiyatlarini ko'rsatuvchi dastur tuzilsin. Bunda misolni vizual muhitda dasturlash uchun 1 ta Button va 1 Label komponentlari kerak bo'ladi.

Forma darchasiga o'rnatilgan komponentlarning xususiyatlarini quyidagicha belgilaymiz:

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Qalam va mo'yqalam xususiyatlari» so'zi kiritiladi.
Label1	Caption (Properties)	«Mo'yqalam xususiyati turlari» so'zi kiritiladi.
Button1	Caption (Properties)	«Natija» so'zi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.

Mavjud komponentlar xususiyatlari kiritilgandan so'ng dastur loyihasi 14.3-rasm ko'rinishga keladi.



14.3-rasm. Forma oynasida dastur loyihasining tuzilishi.

```
//-----
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
}
```



```

Canvas->Pen->Width=8;
Canvas->Brush-> Style=bsClear;
Canvas->Pen->Color=clRed;
Canvas->Rectangle(10,130,90,220);
//-----
Canvas->Brush-> Style=bsSolid;
Canvas->Pen->Color=clMedGray;
Canvas->Brush-> Color=clMedGray;
Canvas->Rectangle(110,130,190,220);
//-----
Canvas->Brush-> Style=bsHorizontal;
Canvas->Pen->Color=clBlack;
Canvas->Brush-> Color=clBlack;
Canvas->Rectangle(210,130,290,220);
Canvas->Brush-> Style=bsVertical;
Canvas->Pen->Color=clBlue;
Canvas->Brush-> Color=clBlue;
Canvas->Rectangle(310,130,390,220);
//-----
Canvas->Brush-> Style=bsFDiagonal;
Canvas->Pen->Color=clMaroon;
Canvas->Brush-> Color=clMaroon;
Canvas->Rectangle(410,130,490,220);
//-----
Canvas->Brush-> Style=bsBDiagonal;
Canvas->Pen->Color=clNavy;
Canvas->Brush-> Color=clNavy;
Canvas->Rectangle(510,130,590,220);
//-----
Canvas->Brush-> Style=bsCross;
Canvas->Pen->Color=clPurple;
Canvas->Brush-> Color=clPurple;
Canvas->Rectangle(610,130,690,220);
//-----

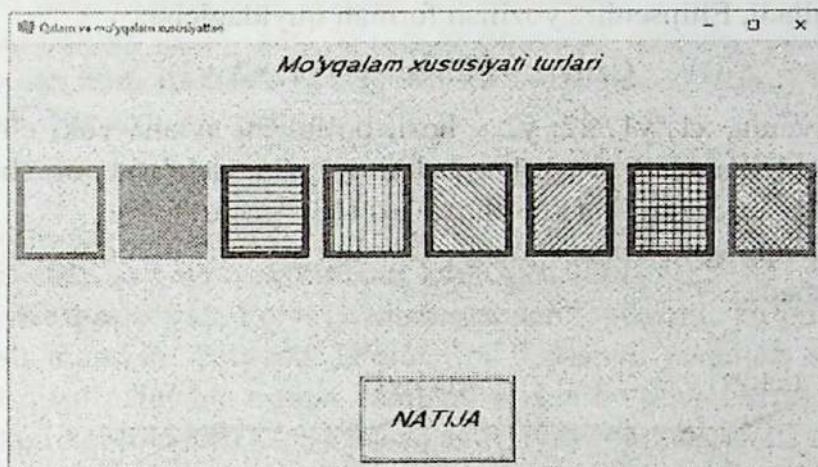
```

```

Canvas->Brush-> Style=bsDiagCross;
Canvas->Pen->Color=clFuchsia;
Canvas->Brush-> Color=clFuchsia;
Canvas->Rectangle(710,130,790,220);
}
//-----

```

Dastur matnida mo'yqalamning xususiyatlari har xil ranglarda ko'rsatilgan. Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



14.4-rasm. Natija oynasi.

14.4. Oddiy geometrik figuralar chizish

Grafik elementlarni (to'g'ri chiziq, aylana, to'g'ri to'rtburchak va h.k.) obyekt yuzasida hosil qilish uchun Canvas xususiyatidan foydalaniladi.

Borland C++ Builder dasturida quyidagi grafik elementlarni hosil qilish mumkin:

1. To'g'ri chiziq. Borland C++ da to'g'ri chiziq hosil qilish uchun LineTo dan foydalaniladi. Uning yozilishi quyidagicha:

```
Canvas->LineTo (x, y);
```


LineTo to'g'ri chiziqli qalam (ko'rsatkich) turgan koordinatadan boshlab x , y nuqtagacha chizadi. Shuning uchun chiziqli boshlang'ich nuqtasini kerakli joyga o'rnatib olish lozim bo'ladi. Bunda biz MoveTo ga murojaat qilamiz:

Canvas->MoveTo (x0, y0);

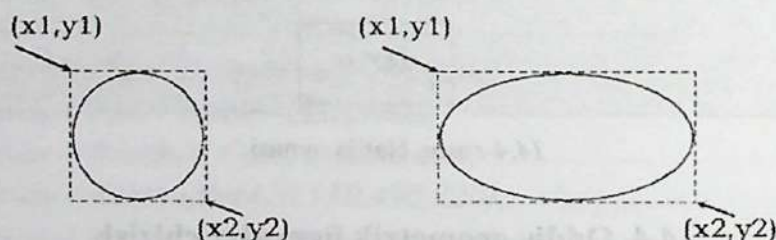
Chiziqli ko'rinishi (rangi, qalinligi va turi) Pen obyekt bilan ifodalanadi.

2. Aylana va ellips. Ellipse uslubi ellips va aylana chizish uchun qo'llaniladi. Ellipsning yozilish formati quyidagicha:

Canvas->Ellipse (x1, y1, x2, y2);

bu yerda, $x1$, $y1$, $x2$, $y2$ – hosil bo'luvchi aylana yoki ellipsga tashqi chizilgan to'g'ri to'rtburchakning mos ravishda yuqori chap va quyi o'ng nuqtalarining koordinatalari (13.5-rasm).

Chiziqli ko'rinishi (rangi, qalinligi va turi) Pen obyekt bilan ifodalanadi.



13.5-rasm. Aylana va ellipsni qiyamat berish.

3. Yoy. Yoy hosil qilish uchun Arc uslubidan foydalaniladi. Uning yozilish formati quyidagicha:

Canvas->Arc (x1, y1, x2, y2, x3, y3, x4, y4);

bu yerda, $x1$, $y1$, $x2$, $y2$ – hosil bo'luvchi yoyning davom ettirib hosil qilinuvchi ellips (aylana)ga tashqi chizilgan to'g'ri to'rtburchakning mos koordinatalari; $x3$, $y3$ – yoyning boshlang'ich nuqtasi; $x4$, $y4$ – yoyning tugash nuqtasi.

Shuni aytib o'tish lozimki, yoy soat strelkasi yo'nalishiga qarama-qarshi yo'nalishda chiziladi.

Chiziqning ko'rinishi (rangi, qalinligi va turi) Pen obyekti bilan ifodalanadi.

4. To'g'ri to'rtburchak. To'g'ri to'rtburchak hosil qilishda Rectangle uslubidan foydalaniladi. Uning yozilish formati quyidagicha:

Canvas->Rectangle (x1, y1, x2, y2);

bu yerda: x1, y1, x2, y2 – to'g'ri to'rtburchakning mos ravishda yuqori chap va quyi o'ng burchak koordinatalari.

RoundRect uslubi ham to'g'ri to'rtburchak chizadi, faqat Rectangle dan farqi shundaki, uning burchaklari yumaloq (silliq) shaklda bo'ladi. Yozilish formati:

Canvas->RoundRect (x1, y1, x2, y2, x3, y3)

bu yerda: x1, y1, x2, y2 – to'g'ri to'rtburchakning mos ravishda yuqori chap va quyi o'ng burchak koordinatalari;

5. Ko'pburchak. Polygon xususiyatidan foydalanib ko'pburchak chizish mumkin. Polygon TPoint tipli massivni parametr sifatida qabul qiladi. Har bir massiv elementi o'zida ko'pburchakning bitta burchagi koordinatasi (x, y) ni saqlaydi. Polygon xususiyati esa shu nuqtalarni ketma-ket to'g'ri chiziqlar bilan tutashtirib chiqadi. Yozilish formati quyidagicha:

Canvas->Polygon (p, n);

bu yerda: p – TPoint tipli massivlar majmuasi bo'lib, ko'pburchakning koordinatalarini o'z ichiga oladi. n – ko'pburchakning burchaklar soni.

6. Sektor. Ellips yoki aylana sektorini hosil qilishda Pie uslubidan foydalaniladi. Pie uslubining umumiy yozilish formati:

Canvas->Pie (x1, y1, x2, y2, x3, y3, x4, y4)

bu yerda: x1, y1, x2, y2 – hosil bo'luvchi sektorni davom ettirib hosil qilinuvchi ellips (aylana)ga tashqi chizilgan to'g'ri to'rtburchak-

ning mos koordinatalari; x_3 , y_3 – sektorning boshlang‘ich nuqtasi; x_4 , y_4 – sektorning tugash nuqtasi.

7. Matn hosil qilish. Grafik obyekt sirtida matnni hosil qilish uchun TextOut usulidan foydalaniladi. TextOut ning yozilish formati quyidagicha:

Canvas->TextOut(x, y, Text);

bu yerda: x , y – matn boshlanuvchi koordinata; Text – hosil bo‘luvchi belgi kattalikdagi matn yoki satrli o‘zgaruvchi.

Hosil bo‘luvchi matn belgilari Canvas obyektiga muvofiq keluvchi Font xususiyati orqali ifodalanadi.

14.6-jadval

Font obyektining xususiyatlari

Xususiyat	Aniqlanishi
Name	Foydalaniluvchi shrift. Qiymat sifatida shrift nomi yoziladi. (Masalan, Times New Roman)
Size	Punktlarda ifodalaniluvchi shrift o‘lchami. Punkt – poligrafiyada qo‘llaniluvchi o‘lchov birligi bo‘lib, u taxminan 1/
Style	Belgini yozish usuli, quyidagicha bo‘lishi mumkin: oddiy, qalin, kursiv, ostiga chizilgan, ustiga chizilgan. Bular quyidagi konstantalar yordamida amalga oshiriladi: <i>fsBold</i> (qalin), <i>fsItalic</i> (kursiv), <i>fsUnderline</i> (ostiga chizilgan), <i>fsStrikeOut</i> (ustiga chizilgan).
Color	Belgi rangi. Qiymat sifatida <i>TColor</i> konstantalaridan foydalanish mumkin.

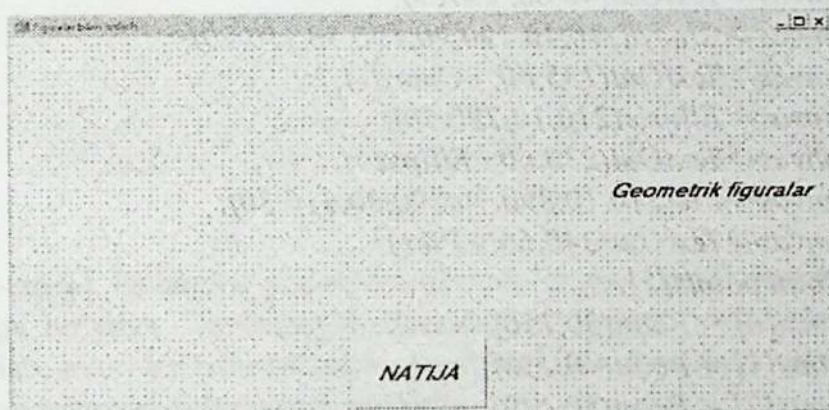
1-misol. Yuqorida keltirilgan usullardan foydalanib grafik obyektlar hosil qilinsin. Misolni vizual muhitda dasturlash uchun 1 ta Label, 1 ta Button komponentlari kerak bo‘ladi.

Forma darchasiga o‘rnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Figuralar bilan ishlash» soʻzi kiritiladi.
Label1	Caption (Properties)	«Figuralar» soʻzi kiritiladi.
Button1	Caption (Properties)	«Natija» soʻzi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.

Mavjud komponentlar xususiyatlari kiritilgandan soʻng dastur loyihasi quyidagi koʻrinishga keladi:



14.6-rasm. Forma oynasida dastur loyihasining tuzilishi.

Dastur loyihasi tayyor boʻlganidan soʻng quyidagi dastur matni kiritiladi:

```
//-----
#include <math.h>
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
```



```

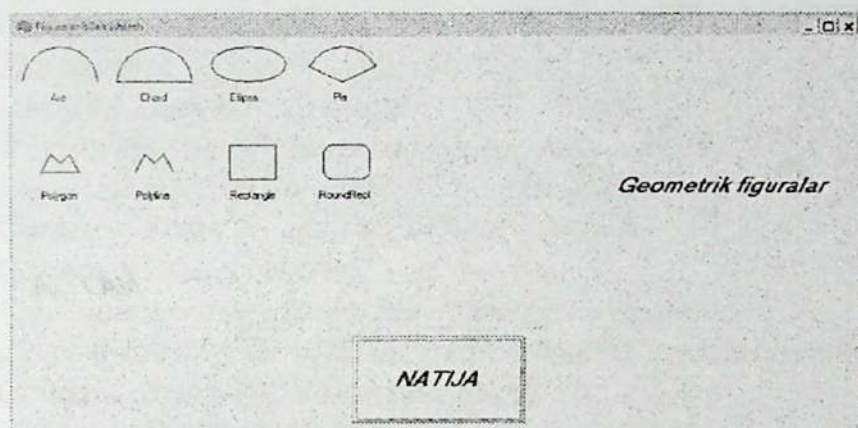
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
Canvas->Font->Style << fsBold;
Canvas->Arc(10,10,90,90,90,50,10,50);
Canvas->TextOut(40,60, «Arc»);
Canvas->Chord(110,10,190,90,190,50,110,50);
Canvas->TextOut(135,60, «Chord»);
Canvas->Ellipse(210,10,290,50);
Canvas->TextOut(230,60, «Ellipse»);
Canvas->Pie(310,10,390,90,390,30,310,30);
Canvas->TextOut(340,60, «Pie»);
TPoint points[5];
points[0] = Point(30,150);
points[1] = Point(40,130);
points[2] = Point(50,140);
points[3] = Point(60,130);
points[4] = Point(70,150);
Canvas->Polygon(points,4);
Canvas->TextOut(30,170, «Polygon»);
points[0].x += 100;
points[1].x += 100;
points[2].x += 100;
points[3].x += 100;
points[4].x += 100;
Canvas->Polyline(points,4);
}

```

```

Canvas->TextOut(130,170, «Polyline»);
Canvas->Rectangle(230,120,280,160);
Canvas->TextOut(230,170, «Rectangle»);
Canvas->RoundRect(330,120,380,160,20,20);
Canvas->TextOut(325,170, «RoundRect»);
}
//-----

```



14.7-rasm. Dastur natijasi.

2-misol. Oynaning yuqori qismini oq rangga, pastki qismini ko'k rangga bo'yash. Oynaning markazida, to'ldirilgan joylar chegarasi bo'ylab, matn kiritish. Misolni vizual muhitda dasturlash uchun 1 ta Button komponentlari kerak bo'ladi.

Forma darchasiga o'rnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

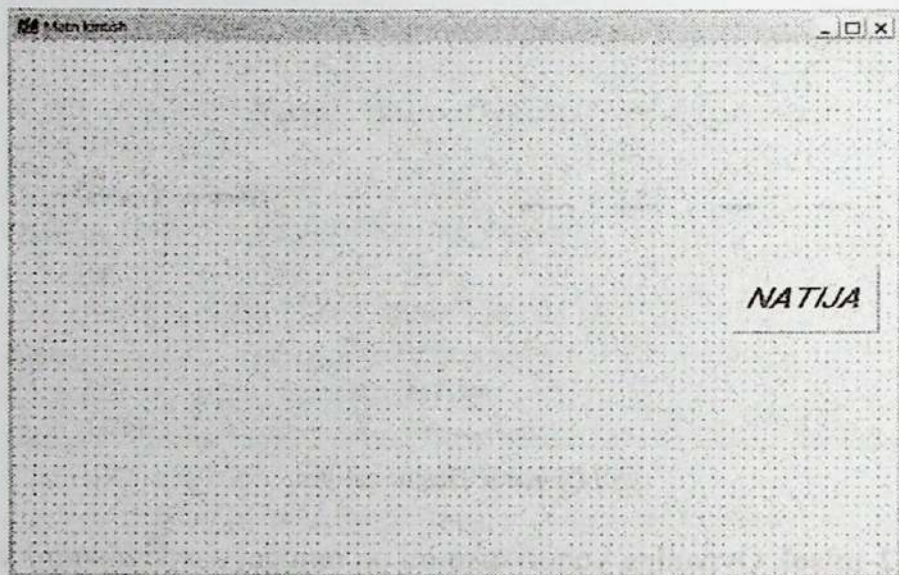
14.8-jadval

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Matn kiritish» so'zi kiritiladi.

Button1	Caption (Properties)	«Natija» so'zi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.

Mavjud komponentlar xususiyatlari kiritilgandan so'ng dastur loyihasi quyidagi ko'rinishga keladi:



14.8-rasm. Forma oynasida dastur loyihasining tuzilishi.

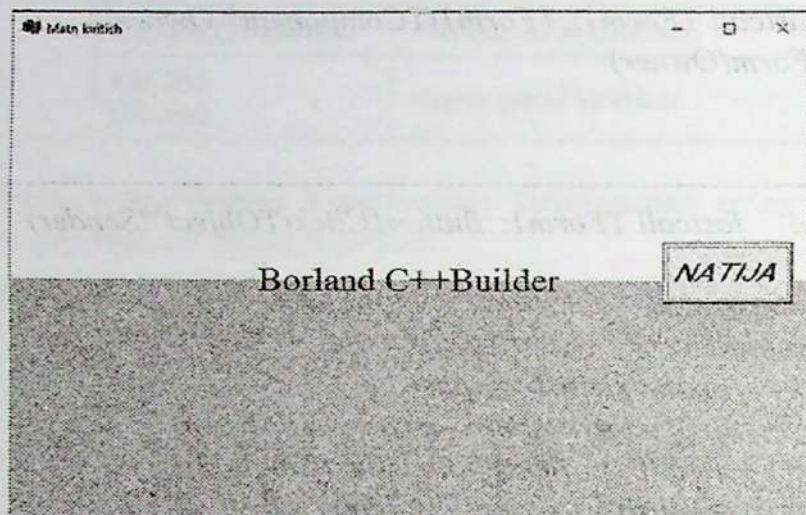
Dastur loyihasi tayyor bo'lganidan so'ng quyidagi dastur matni kiritiladi:

```
//-----
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
```

```

__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
AnsiString ms = «Borland C++ Builder»;
TRect aRect;
int x,y; // matn kiritish nuqtasi
// oynaning yuqori qismini oq rangga bo'yash
aRect = Rect(0,0,ClientWidth,ClientHeight/2);
Canvas->Brush->Color = clWhite;
Canvas->FillRect(aRect);
// oynaning pastki qismini moviy rangga bo'yash
aRect = Rect(0,ClientHeight/2,ClientWidth,ClientHeight);
Canvas->Brush->Color = clSkyBlue;
Canvas->FillRect(aRect);
Canvas->Font->Name = «Times New Roman»;
Canvas->Font->Size = 24;
// matnni oyna markaziga joylash
x = (ClientWidth - Canvas->TextWidth(ms)) /2;
y = ClientHeight/2 - Canvas->TextHeight(ms) /2;
Canvas->Brush->Style = bsClear; // ?matnni kiritish sohaci
// bo'yash kerak emas
Canvas->Font->Color = clBlack;
Canvas->TextOut(x,y,ms); // matn kiritish
}
//-----

```

14.9-rasm. Natija oynasi.

3-misol. Olimpiya aylanalarini chizish. Misolni vizual muhitda dasturlash uchun 1 ta Button komponenti kerak bo'ladi.

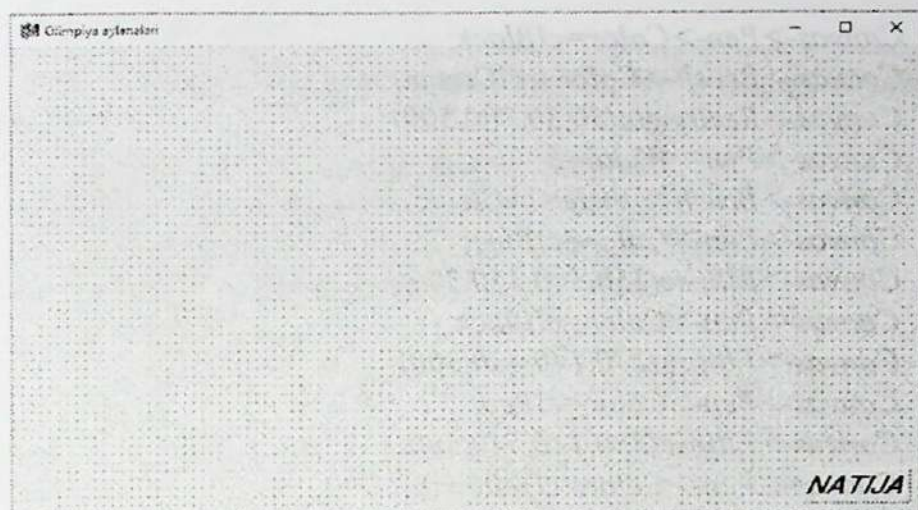
Forma darchasiga o'rnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

14.9-jadval

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Olimpiya aylanalarini» so'zi kiritiladi.
Button1	Caption (Properties)	«Natija» so'zi kiritiladi.
	OnClick (Events)	Dastur matni kiritiladi.

Mavjud komponentlar xususiyatlari kiritilgandan so'ng dastur loyihasi quyidagi ko'rinishga keladi:



14.10-rasm. Dastur loyihasining ko'rinishi.

Dastur loyihasi tayyor bo'lganidan so'ng quyidagi dastur matni kiritiladi:

```
//-----
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
Canvas->Pen->Width=1;
```

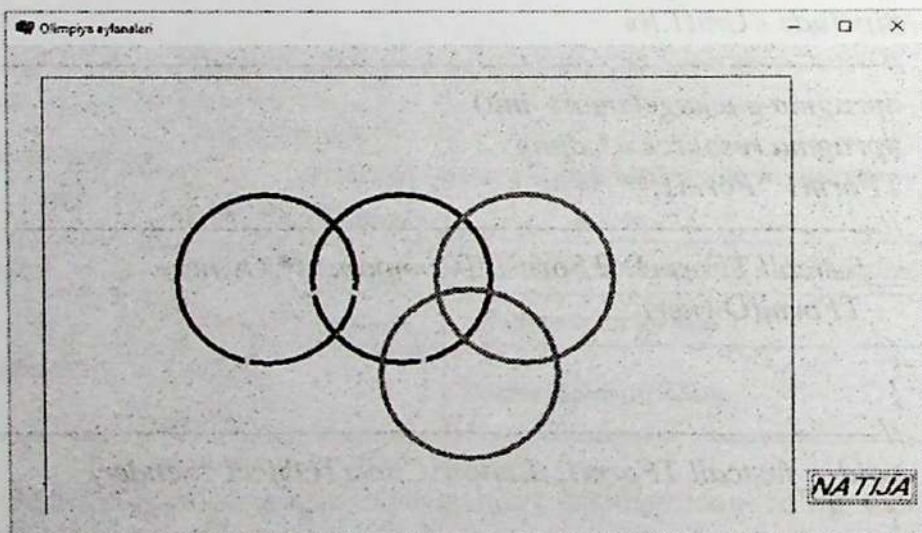


```

Canvas->Pen->Color=clBlack;
Canvas->Brush->Color =clCream;
Canvas->Rectangle(30,30,700,500);
Canvas->Pen->Width=5;
Canvas->Brush-> Style=bsClear;
Canvas->Pen->Color=clBlue;
Canvas->Ellipse(150,140,310,300);
Canvas->Pen->Color=clBlack;
Canvas->Ellipse(270,140,430,300);
Canvas->Pen->Color=clRed;
Canvas->Ellipse(380,140,540,300);
Canvas->Pen->Color=clYellow;
Canvas->Ellipse(210,230,370,390);
Canvas->Pen->Color=clGreen;
Canvas->Ellipse(330,230,490,390);
}
//-----

```

Dastur matni kiritib bo'lingandan so'ng F9 tugmasi bosiladi va natijada quyidagi dastur ko'rinishi paydo bo'ladi:



14.11-rasm. Natija oynasi.

Nazorat savollari

1. C ++ Builder 6 da grafika bilan bog'liq necha turdagi obyektlar mavjud?
2. TPicture obyekti qanday vazifani bajaradi?
3. Grafik adapterlar haqida ma'lumot bering.
4. Koordinatalar nuqtalari qanday shakllantiriladi?
5. Style obyekti qanday xususiyatlarga ega?

15-BOB. TEXNIK TIZIMLARDA INTEGRALLASHGAN MUHITDA MA'LUMOTLAR BAZASINI YARATISH VA QAYTA ISHLASH VOSITALARI

15.1. Ma'lumotlar bazasining texnologiyalari

Ma'lumotlar bazasi (MB) – bu o'zaro bog'langan va tartiblangan ma'lumotlar majmuasi bo'lib, u ko'rilayotgan obyektlarning xususiyatini, holatini va obyektlar o'rtasidagi munosabatni ma'lum sohada tavsiflaydi.

Ma'lumotlarni saqlash, uzatish va qayta ishlash uchun ma'lumotlar bazasini yaratish, so'ngra undan keng foydalanish bugungi kunda dolzarb bo'lib qolmoqda. Katta hajmdagi ma'lumotlarni boshqarish, ulardan kerakli ma'lumotlarni so'rov orqali istalgan ko'rinishda chiqarib olish, ma'lumotlarning zaxira nusxalarini olish, katta hajmdagi ma'lumotlarni siqish, qulay interfeysda ma'lumotlar bazasi ustidan nazorat o'rnatish, ma'lumotlar asosida hisobotlar hosil qilish va bulardan boshqa ma'lumotlar ustida juda katta ko'lamdagi ishlarni amalga oshiradigan dasturiy komplekslar mavjud. Bunday dasturlar majmuyi ma'lumotlar bazasini boshqarish tizimlari deb yuritiladi.

Ma'lumotlar bazasini boshqarish tizimlari (MBBT) – bu dasturiy ta'minot bo'lib, tashkilot ma'lumotlarini markazlashtirish hamda bu orqali ma'lumotlarni samarali boshqarish va amaliy dasturiy vositalar yordamida saqlanayotgan ma'lumotlardan foydalanish imkonini beradi. MBBT amaliy dasturiy vositalar va fayllarning fizik ma'lumotlari o'rtasida interfeys vazifasini bajaradi. An'anaviy ma'lumotlar fayllari tizimidan foydalanib, dasturchi dasturda ishlatiladigan har bir ma'lumotlarning elementi hajmini va formatini belgilashi, so'ngra ular joylashgan maydonni kompyuterda belgilashi kerak bo'ladi.

Ma'lumotlar bazasi texnologiyasi an'anaviy fayllarni tashkil etishning ko'pgina muammolarini kamaytiradi. Ma'lumotlar bazasini

ta'riflaydigan bo'lsak, bu ma'lumotlar majmuyi bo'lib, u ma'lumotlarni samarali nazorat qilish va ma'lumotlarni markazlashtirish orqali bir nechta ilovalarni boshqarish uchun tashkil etiladi. Bunda har bir ilova uchun ma'lumotlarni alohida fayllarda saqlash o'rniga foydalanuvchi barcha ma'lumotlarni markazlashtirilgan joyda saqlaydi.

Vizual ma'lumotlar bazasi komponentlarini quvvatlaydigan BDE (Borland Database Engine) ilova va ma'lumotlar bazasi o'rtasida interfeys vazifasini bajaradi.

MBBT – bu ko'plab foydalanuvchilar tomonidan MBni yaratish, unga qo'shimcha ma'lumotlarni kiritish va MBni birgalikda ishlatish uchun zarur bo'lgan dasturlar majmuyidir. MBBTning tarkibidagi asosiy komponenti – bu ma'lumotlardir.

Ma'lumotlar bazasini boshqarish tizimi umumiy tushuncha bo'lib, uning tarkibiga ma'lumotlar bazasi ham kiradi. Misol uchun, mashina chiqaradigan zavod MBBT hisoblanadi, mashinalar esa ma'lumotlar bazasidir. MBBT umumiy bir dastur bo'lib, ma'lumotlar bazasini boshqaradi va uning bir tekis ishlashini ta'minlaydi.

MBBT ortiqcha ma'lumotlarni kamaytirib, qayta takrorlanuvchi ma'lumotlarni minimum darajagacha kamaytirish imkonini beradi. MBBT dasturlar va ma'lumotlarni ajratadi, buning natijasida ma'lumotlardan mustaqil foydalanish imkoni bo'ladi. Ma'lumotlardan foydalanish imkoniyati oshiriladi, dasturni ishlab chiqish va qo'llab-quvvatlash uchun sarf-xarajatlar kamayadi, shuningdek foydalanuvchilar va dasturchilar ma'lumotlar bazasida ma'lumotlar uchun maxsus so'rovlarini bajarishlari mumkin bo'ladi. MBBT tashkilotga ma'lumotlarni markazlashtirilgan holda boshqarish va axborot xavfsizligini oshirish imkonini beradi.

MBBTga misol qilib: Oracle MBBT, MySQL MBBT, MS SQL Server MBBT, MS Access MBBT kabilarni olish mumkin, bular ichida ma'lumotlar bazasini yaratsa bo'ladi.

Texnik qismi tashqi qo'shimcha xotiradan iborat bo'lsa, dastur qismi esa MB bilan foydalanuvchi o'rtasidagi muloqotni tashkil qilishni amalga oshiradi. MBning tuzilishi o'rganilayotgan obyektning ma'lumotlari ko'rinishi, ma'nosi, tuzilishi va hajmiga bog'liq bo'ladi.

Hozirgi kunda kompyuter uchun, shuningdek, server kompyuterlar uchun MBBTning eng mashhur turidan biri relatsion model hisoblanadi. Relatsion modellarda obyektlar va ularning o'zaro aloqalari ikki o'lchovli jadval ko'rinishida tasvirlanadi. Ma'lumotlarning bunday ko'rinishida tasvirlanishi obyektlarning o'zaro aloqalarining yaqqol tasvirlanishiga asos bo'ladi.

Misol sifatida Access dasturida yaratilgan 15.1-jadvalni keltirish mumkin.

15.1-jadval

№	Familiyasi	Ismi	Tug'ilgan sanasi	Guruhi	Turar joyi
1	Anvarov	Olim	22.05.20001	113-21	Mirzo Ulug'bek tumani, 20-uy
2	Kobulov	Farhod	12.02.2001	118-21	Ibn Sino ko'chasi, 14-uy
3	Aminov	San'at	14.05.1999	110-21	Olmazor tumani, 15-uy
4	Tolipov	Jasur	15.03.1999	22-21	Beruniy ko'chasi, 22-uy

15.2. Integrallashgan muhitda ma'lumotlar bazasini yaratish

C++ Builder dagi ma'lumotlar bazasi – bu yozuvlar to'plamidan iborat obyekt, ularning har biri, o'z navbatida, maydonlardan va joriy yozuvlardan tashkil topadi. Ma'lumotlar to'plami haqiqiy jadval yoki so'rov natijasi, jadvalning bir qismi bo'lishi yoki bir nechta jadvallarni birlashtirishi mumkin.

Ushbu bo'limda Borland C++ Builder 6 muhitida ma'lumotlar bazasi bilan ishlash asoslarini ko'rib o'tamiz. Borland C++ Builder 6 dasturidan foydalanib, bitta foydalanuvchiga asoslangan ma'lumotlar bazasini, shuningdek serverlar bilan ishlaydigan ilovalarni yaratish mumkin, bunga misol qilib Oracle, Sybase, Informix, Interbase, MS SQL Server, DB2 va ODBC MBBTlarni keltirish mumkin.

Borland C ++ Builder 6 dasturining ma'lumotlar bazalarini ishlatadigan ilovalar yaratish bilan bog'liq bo'lgan imkoniyatlari juda keng.

Borland C ++ Builder 6 dasturida ma'lumotlar bazasi bilan ishlovchi juda ko'p komponentlar mavjud. Ba'zi bir komponentlar bilan tanishib chiqamiz:

-  – TTable obyekti ma'lumotlar omboridagi mavjud jadval bilan muloqot o'rnatish uchun xizmat qiladi. TTable ixtiyoriy tipdagi (FoxPro, ODBC, SQL va hokazo) ma'lumotlar bazasining har bir yozuviga va maydoniga to'g'ridan to'g'ri murojaat qila oladi. Bu komponent, shuningdek, alohida hisobotlar bilan ham muloqot o'rnatadi.

TTable obyektidan foydalanishdan oldin unga ma'lumotlar bazasini ulash kerak, ya'ni shu komponentning DatabaseName xususiyatida chiqadigan ro'yxatdan kerakli jadvalni tanlash va TableName xususiyatidagi ro'yxatdan kerakli jadval nomini tanlash kerak. TTable obyektining bir necha xususiyatlarini ko'rib chiqamiz:

- Active xususiyati ikkita qiymat qabul qiladi, agar «true» qiymat bo'lsa, jadval faol hisoblanadi, «false» qiymat bo'lsa, jadval yopiqligini bildiradi.

- DatabaseName – kerak bo'lgan jadval joylashgan katalog nomi yoki masofada joylashgan ma'lumotlar bazasining tashqi xotiraga yozilgan qo'shimcha nomi (alias). «Alias» qo'shimcha dastur BDE konfiguratsiyasi yordamida yoki Database/Explore bosh menyusida joylashgan SQL Explorer yordamida o'rnatiladi. Bu xususiyat faqat jadval yopilgan (uning «active» xususiyati «false») bo'lsa, o'zgarishi mumkin, masalan:

Table1->Active = false;

Table1->DatabaseName = «BCDEMOS»

Table1->Active = true;

- TableName – jadvalning nomi.

- Exclusive – agar bu xususiyat «true» bo'lsa, ushbu joriy dastur tomonidan ochilgan bo'lsa, boshqa hech bir foydalanuvchi jadvalni

ocholmaydi. Agar bu xususiyat «false» bo'lsa (asl qiymati), boshqa foydalanuvchilar ushbu jadvalni ochishlari mumkin bo'ladi.

- **IndexName** – jadval uchun ikkilamchi indeksni aniqlaydi. Jadval ochiq holatda bo'lganda bu xususiyatni o'zgartirib bo'lmaydi.

- **MasterFields** – boshqa jadvalga havola yaratish uchun maydon nomini belgilaydi.

- **MasterSource** – **TDataSource** komponentining nomi bo'lib, **TTable** komponenti bog'langan jadvaldan ma'lumotlarni olish imkonini beradi.

- **Fields** – **TField** obyektlar massivi bo'lib, ushbu xususiyatdan maydonning tartib raqamiga murojaat etish mumkin. Misol uchun:

`Edit1->Text=Table1->Fields[2]->AsString;`

-  **TADOTable** obyekti ham xuddi **TTable** obyekti kabi ma'lumotlar bazasidagi biror jadvalga bog'lanish va unga murojaat qilish uchun xizmat qiladi. Bu obyekt asosan **MSAccess** ma'lumotlar bazasini boshqarish tizimida yaratilgan ma'lumotlar bazasi bilan ishlashga mo'ljallangan. Bu obyekt **TADOconnection** obyekti bilan birga qo'llanilib, **TADOConnection** ma'lumotlar bazasiga ulanadi. Shundan so'ng bir yoki bir nechta **TADOTable** obyektlari **Connection** xususiyati yordamida **TADOConnection**ga ulanadi va **TableName** xususiyati yordamida kerakli jadval bilan bog'lanadi. Obyektni faollashtirish uchun «Active» xususiyati qiymatini «true»ga o'tkazish kerak. Bu obyekt yordamida ma'lumotlar bazasidan ma'lumotlarni biror filtr yordamida ajratib olish mumkin.

TADOTable obyektining bir necha xususiyatlarini ko'rib chiqamiz:

- **Active** xususiyati ikkita qiymat qabul qiladi, agar «true» qiymat bo'lsa, jadval faol hisoblanadi, «false» qiymat bo'lsa, jadval yopiqligini bildiradi.

- **ReadOnly** xususiyati agar «true» qiymat qabul qilsa, jadval «Faqat o'qish uchun» rejimi faol bo'ladi. Jadval ochiq bo'lganda **ReadOnly** xususiyatini o'zgartirib bo'lmaydi.

- **TableName** – jadvalning nomi.

- **Connection** – jadvaldagi ma'lumotlar bazasi bilan bog'lanish uchun xizmat qiladi.

• Filter – jadvaldagi ma'lumotlar bazasini saralash imkonini beradi. Misol uchun, saralashni amalga oshirish uchun quyidagi dastur kodini kiritish mumkin:

```
{ADOTable1->Filtered = false;  
ADOTable1->Filter = «Name like 'C*'»;  
ADOTable1->Filtered = true;  
}
```

• Name – komponentning nomi.

TDataSource obyektini bevosita TTable yoki TAdoTable-ga bog'lanib, ma'lumotlar bazasidagi yozuvlarni tahrirlash, ularga murojaat qilish imkonini beradi. Buning uchun komponentning DataSet xususiyatidagi ro'yxatdan kerakli Table elementi tanlanadi va shu orqali ikki obyekt bir-biriga bog'lanadi. TDataSource obyektini bitta ma'lumotlar bazasidagi bitta jadvalga ulana oladi.

Yuqoridagi uchala obyekt ham dastur bajarilish vaqtida ko'rinishmaydigan obyekt bo'lib, Formalar Dizayneri ko'rinishida ularni formaga tashlaganda o'lchamlarni o'zgartirib bo'lmaydi. Ularni ma'lumotlar omboriga Formalar Dizayneri rejimida ham, dasturiy yo'l bilan dastur bajarilish vaqtida ham bog'lash mumkin. Buning uchun quyidagicha kodlar yoziladi:

```
{  
Table1->DatabaseName= «DBDEMOS»;  
Table1->TableName= «animals.dbf»;  
Table1->Active=True;  
DataSource1->DataSet=Table1;  
DBGrid1->DataSource = DataSource1;  
}
```

TDBgrid obyektini ma'lumotlar bazasidagi hisobotlar, jadvallar va so'rovlardagi ma'lumotlarni jadval ko'rinishida namoyish etish uchun qo'llanadi. Bu obyekt yordamida ma'lumotlar bazasidagi yozuvlarni namoyish qilish, tahrirlash va o'zgartirish mumkin. Kiritilgan o'zgartirishlar joriy yozuv ustida boradi va bu o'zgarishlar faqat

boshqa yozuvga o'tganda yoki dastur yopilganida saqlab qolinadi. TDBgrid obyektini bevosita DataSource xususiyati yordamida TDataSource obyektga bog'lanadi va shu orqali ma'lumotlarni namoyish etadi.

 TDBNavigator (QDBCtrls) obyektini dasturda TDBgrid yoki TDBedit komponentlari orqali ma'lumotlar bazasi yozuvlariga murojaat qilinayotgan vaqtda qo'llaniladi. TDBNavigator foydalanuvchiga ma'lumotlar bazasidagi yozuvlarni tahrirlash yoki ko'rib chiqishda foydalaniladi. Foydalanuvchi TDBNavigator tugmalardan birini bosganda shu tugma bilan bog'langan amal dasturda bajariladi.

Quyidagi 15.2-jadvalda TDBNavigator obyektining tugmalari va ular bajaradigan amallar ko'rsatib o'tilgan.

15.2-jadval

TDBNavigator obyektining tugmalari

Tugma	Bajaradigan amali
 First	Ma'lumotlar bazasidagi dastlabki yozuvni faollashtirish. U faqat joriy yozuv dastlabki yozuv bo'lmagandagina faol bo'ladi.
 Prior	Ma'lumotlar bazasida joriy yozuvdan oldingi yozuvni faollashtirish. U faqat joriy yozuv dastlabki yozuv bo'lmagandagina faol bo'ladi.
 Next	Ma'lumotlar bazasida joriy yozuvdan keyingi yozuvni faollashtirish. U faqat joriy yozuv oxirgi yozuv bo'lmagandagina faol bo'ladi.
 Last	Ma'lumotlar omboridagi oxirgi yozuvni faollashtirish. U faqat joriy yozuv oxirgi yozuv bo'lmagandagina faol bo'ladi.
 Insert	Jadvalga ma'lumotlarni kiritish uchun yangi satr qo'shish. Bunda satrning ixtiyoriy maydoniga ma'lumot kiritilganda o'zgarishlar saqlanadi.
 Delete	Joriy yozuvni o'chirish. Bunda yozuvni o'chirish haqida so'rov chiqariladi va o'chirilgan yozuv qayta tiklanmaydi.

 Edit	Joriy yozuvni o'zgartirish, tahrirlash mumkin bo'lgan holatga o'tkazish.
 Post	Kiritilgan o'zgarishlarni xotirada saqlash. Bunda joriy maydonning oldingi ma'lumotlari o'rniga kiritilgan o'zgarishlar saqlanadi.
 Cancel	Joriy yozuvga kiritilgan o'zgarishlarni bekor qilish. Bu amaldan joriy yozuvni almashtirguncha foydalanish mumkin.
 Refresh	Kiritilgan ma'lumotlarni yangilash.

15.3. Ma'lumotlar bazasini qayta ishlash

Yuqorida ko'rib chiqilgan komponentlar asosida texnik tizimlarda ma'lumotlar bazasini qayta ishlash jarayonini ko'rib chiqamiz.

1-misol. MS Access dasturida yaratilgan ma'lumotlar bazasidan foydalanib, forma darchasida guruhining ma'lumotlar bazasini qayta ishlash dasturi ishlab chiqilsin.

Misolni vizual muhitda dasturlash uchun 1 ta ADOConnection, 1 ta ADOTable, 1 ta DataSource, 1 ta DBNavigator, 1 DbGrid, 3 ta Label, 2 ta Edit va 2 BitBtn komponentlari kerak bo'ladi. Dastur 2 ta Form darchasida amalga oshiriladi.

Formal darchasiga o'rnatilgan komponentlar xususiyatlarini quyidagicha belgilaymiz:

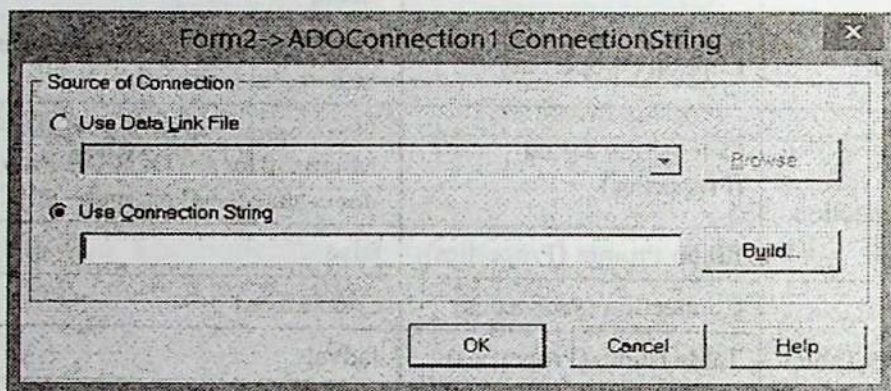
15.3-jadval

Komponentlar xususiyatlarini kiritish

Komponent nomi	Xususiyat nomi (Object Inspector darchasining holati)	Amalga oshiriladigan jarayon
Form1	Caption (Properties)	«Texnik tizimlarda ma'lumotlar bazasi» so'zi kiritiladi.
Label1	Caption (Properties)	«Foydalanuvchi loginini kiriting» so'zi kiritiladi.

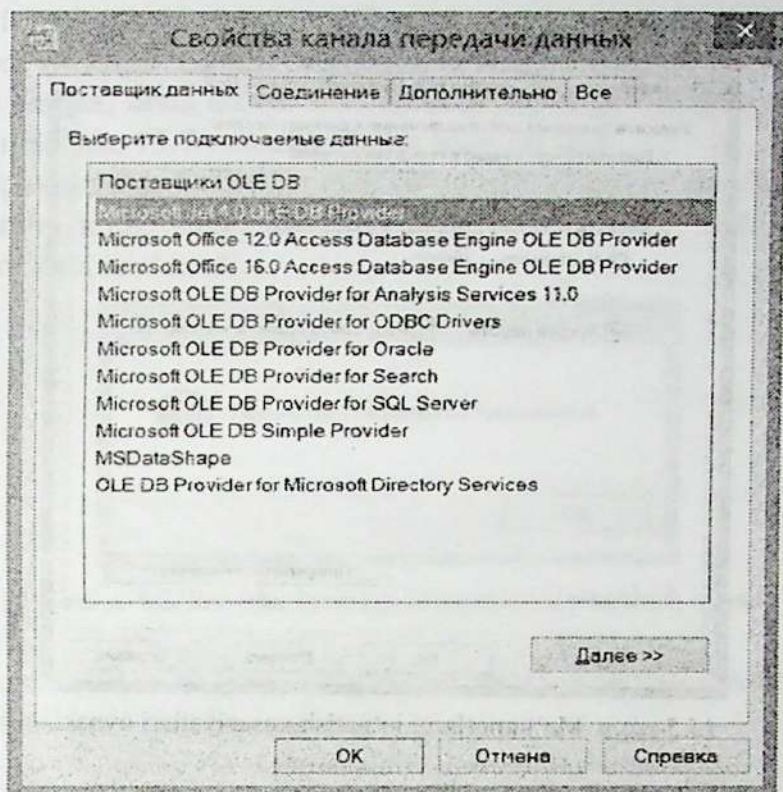
Kerakli komponentlar joylashtirilgandan so'ng komponentlar ustida quyidagi amallarni bajaramiz:

AdoConnection obyektining ConnectionString xususiyatiga kiriladi va mavjud ma'lumotlar bazasi bilan bog'lanadi. Buning uchun Microsoft Access dasturida «Baza.mdb» nomi bilan jadval yaratamiz. Faylni loyiha yaratilgan joriy katalogga saqlaymiz. Faylga bog'lash quyidagicha amalga oshiriladi (15.3-rasm):



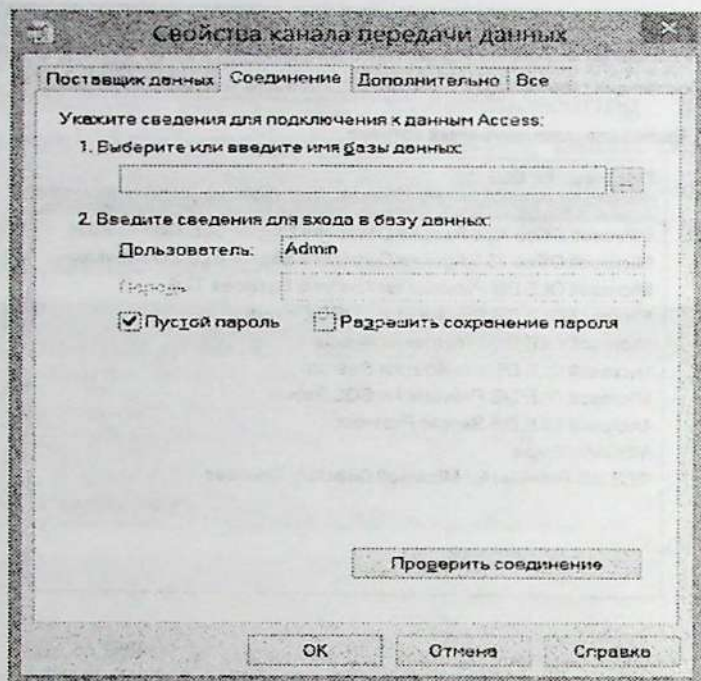
15.3-rasm. ConnectionString xususiyati oynasi.

Hosil bo'lgan darchadan «Build» tugmasini bosamiz. Natijada bog'lanish imkonini beruvchi ro'yxat paydo bo'ladi. Ro'yxatdan MS Access dasturi bilan ishlovchi Microsoft Jet OLEDB 4.0 Provider ni tanlaymiz (15.4-rasm):



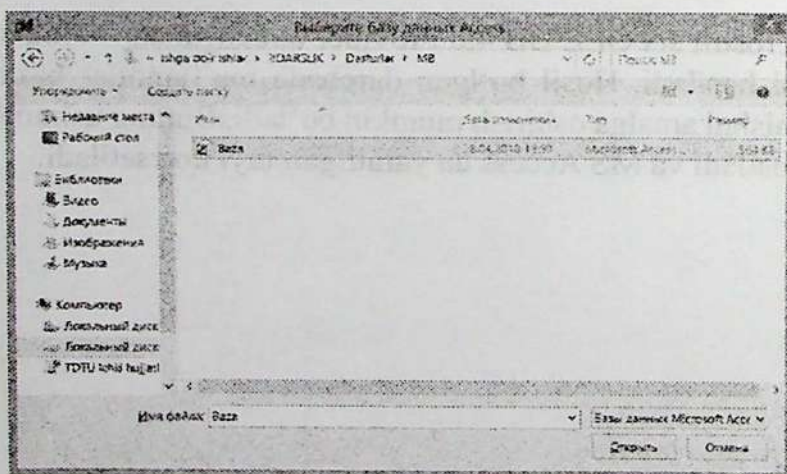
15.4-rasm. Microsoft Jet OLE DB 4.0 Provider ni tanlash oynasi.

Microsoft Jet OLE DB 4.0 Provider tanlangandan so'ng «Далее» tugmasi bosiladi. Hosil bo'lgan darchada ma'lumotlar bazasi bilan bog'lanishni amalga oshirish mumkin bo'ladi. Buning uchun [...] tugmasi bosiladi va MS Access da yaratilgan fayl ko'rsatiladi.



15.5-*расм.* Ма'lumotlarni jo'natish xususiyatlari oynasi.

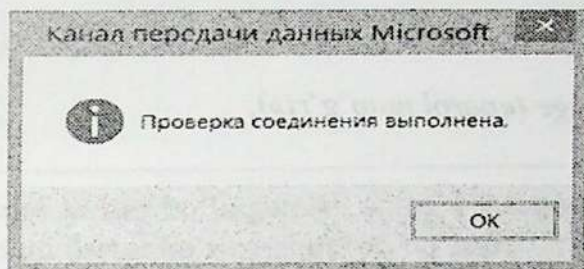
Ма'lumotlar bazasi joylashgan faylni tanlash uchun Windows operatsion tizimida standart darchadan foydalaniladi.



15.6-*расм.* Faylni ochish oynasi.

Fayl tanlangandan so'ng «Проверить подключение» tugmasi bilan tekshiriladi. Agarda «Проверка подключения выполнена» natijasi chiqsa, unda dastur ma'lumotlar bazasiga ulangan sanaladi. Ulangan bo'lsa, 15.7-rasmdagi holat hosil bo'ladi.

Barcha amallar bajarilganidan so'ng «Ok» tugmasi bosiladi. Qolgan barcha komponentlarning xususiyatlari 15.3 va 15.4-jadvalda berilgani kabi sozlanadi.



15.7-rasm. Ma'lumotlar bazasiga ulanganligini tekshirish oynasi.

Dastur loyihasi tayyor bo'lganidan so'ng quyidagi dastur matni kiritiladi (Form1 darchasi uchun):

```
//-----
#include <vcl.h>
#pragma hdrstop
#include «Unit1.h»
#include «Unit2.h»
//-----
#pragma package(smart_init)
#pragma resource «*.dfm»
TForm1 *Form1;
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner)
{
}
//-----
```



```
void __fastcall TForm1::BitBtn1Click(TObject *Sender)
```

```
{  
if (Edit1->Text == «MK»)  
{  
if (Edit2->Text == «12345»)  
{  
Form2->ShowModal();  
}  
} else  
{  
ShowMessage («parol noto 'g'ri»);  
}  
}  
}  
}  
}  
}
```

```
//-----
```

```
Form2 darchasi uchun dastur matni:
```

```
//-----
```

```
#include <vcl.h>  
#pragma hdrstop  
#include «Unit2.h»
```

```
//-----
```

```
#pragma package(smart_init)  
#pragma resource «*.dfm»  
TForm2 *Form2;
```

```
//-----
```

```
__fastcall TForm2::TForm2(TComponent* Owner)  
: TForm(Owner)
```

```
{  
}  
}
```

```
//-----
```

```
void __fastcall TForm2::DBGrid1TitleClick(TColumn *Column)  
{
```

```

if (ADOTable1->Active)
if ((ADOTable1->Sort.Pos(Column->FieldName) > 0) && (AD-
OTable1->Sort.Pos(«ASC») > 0))
{
ADOTable1->Sort = Column->FieldName + « DESC»;
}
else
{
ADOTable1->Sort = Column->FieldName + « ASC»;
}
}
//-----

```

Dastur matni kiritib bo‘lingandan so‘ng F9 tugmasi bosiladi va natijada quyidagi dastur ko‘rinishi paydo bo‘ladi:

15.8-rasm. Natija oynasi.

Foydalanuvchi logini va parol to‘g‘ri kiritilgandan so‘ng tizimga kirish imkoniyati paydo bo‘ladi.

№	FAMILIYA	ISM	SHARFI	DATA	SEX	YOIL	TAQILAR KOD
1	Sag'ratov	Mirzozir	Vartsovich	08.11.1960	Erkak		10000971 TTAT, AT
2	Kadirov	Mir-Xusan	Mirpulatovich	22.05.1985	Erkak		5062971 TTAT, AT
3	Alkbarova	Shohida	Azafavna	12.02.1974	Ayol		5062971 TTAT, AT
4	Karimova	Nazima	Aybekovna	15.01.1982	Ayol		5062971 TTAT, AT
5	Toshmurova	Nadira	Zakirovna	14.02.1970	Ayol		5062971 TTAT, AT
6	Karimova	Dilbar	Karimovna	16.07.1940	Ayol		3062971 TTAT, AT
7	Toshmurova	Shahnoza	Sabirovna	04.01.1955	Ayol		3094000 TTAT, AT
8	Kurbanova	Kabira	Erkinovna	03.09.1963	Ayol		3062971 TTAT, AT
9	Toltaganov	Zakirjon	Yulubayevich	27.09.1985	Erkak		3062971 TTAT, AT
10	Elmuradov	Tasummalik	Dilshod o'g'li	02.02.1989	Erkak		3062971 TTAT, AT

15.9-rasm. Natija oynasi.

Nazorat savollari

1. Ma'lumotlar bazasi deb nimaga aytiladi?
2. C++ Builder 6 da ma'lumotlar bazasi qaysi model asosida ishlaydi?
3. MBda asosan qaysi komponentlar ishlatiladi?
4. Faylni ochish oynasi qanday ketma-ketlikda bajariladi?
5. Foydalanuvchining logini nima uchun kiritiladi?

GLOSSARIY

O'zbek tilida	Rus tilida	Ingliz tilida
A		
Axborot – lotincha informatio so'zidan kelib chiqqan bo'lib, bi- ror-bir dalil, voqea, hodisani bayon qilish tushunchasini bildiradi.	Информация – сведения об объек- тах и явлениях окружающей среды, их параметрах, свойствах и состо- янии, которые воспринимают ин- формационные системы в процессе жизнедеятельности и работы.	Information – information about objects and phenomena of the envi- ronment, their parameters, properties and condition, which are perceived by information systems in the process of life and work.
Axborot texnologiyalari – ilmiy va muhandislik bilimlari hamda usullari va vositalaridan foydalanish va ularni yig'ish, uzatish, saqlash va qayta ish- lash axborot tizimi majmuasidir.	Информационная технология – система научных и инженерных знаний, а также методов и средств, которая используется для создания, сбора, передачи, хранения и обра- ботки информации в предметной области.	Information technology – is specific to the Information Technology (IT) industry, or to the skills necessary for performing information technolo- gy jobs. IT training includes courses related to the application, design, de- velopment, implementation, support or management of computer-based information systems.
Avtomatlashtirilgan axborot tex- nologiyalari – axborot technologi- yasi bo'lib, ma'lumotlarni yig'ish, saqlash, uzatish, qayta ishlashda ish- latiladigan usullar va hisoblash tex- nikasi vositalari va aloqa tizimidan iborat bo'ladi.	Автоматизированная информа- ционная технология – информа- ционная технология, в которой для передачи, сбора, хранения и обра- ботки данных используются мето- ды и средства вычислительной tex- ники и систем связи.	Automated Information System – is a combination of computer hardware and computer software, data, and/or telecommunications that performs functions such as collecting, process- ing, storing, transmitting and display- ing information.

Автоном експерт тизимлари – o'ziga xos «ekspert» vazifalari uchun foydalanuvchi bilan maslahat rejimida ishlaydi.	Автономные экспертные системы – работает в режиме консультации с пользователем под конкретные «экспертные» задачи.	Autonomous expert systems – work in consultation with the user for specific «expert» tasks.
Antivirus – viruslarni aniqlovchi yoki aniqlovchi va yo'q qiluvchi dastur. Agar virus yo'q qilinmasa, zararlangan dastur yo'q qilinadi. Yana – viruslardan himoyalashga zararlangan dasturiy modullar va tizimli ma-konlarni aniqlashga, hamda zararlangan obyektlarning dastlabki holatini tiklashga mo'ljallangan dastur.	Антивирус – программа, обнаруживающая или удаляющая вирусы. Если вирус удалить не удается, то зараженная программа уничтожается. Еще программа, предназначенная для защиты от вирусов, обнаружения зараженных программных модулей и системных областей, а также восстановления исходного состояния зараженных объектов.	Antivirus – the program that detect or detect and remove viruses. If virus remove not possible, then the infected program is destroyed. Another program, designed to protect against viruses, detecting infected software modules and system areas as well as restore the original state of infected object.
Arxiv – arxivator dastur yordamida zichlashtirilgan ma'lumotlar va dasturlar majmuyi.	Архив – совокупность данных или программ, сжатых программой архиватором.	Archive – a set of data or programs that are compressed by the archiver program.
Atribut – alomat, tarkibida xarakteristikalaridan biri bo'lgan ma'lumotlar tavsifi: ismi, turi, uzunligi, soni, ifodalanish shakli, sanoq tizimi.	Атрибут – признак, описатель данных, содержащий одну из характеристик данного: имя, тип, длину, количество, форму представления, систему счисления.	Attribute – sign, the description of the data comprising one of the characteristics of the: name, type, length, quantity, representation, number system.

Аутентификация – odatda tizim resurslaridan foydalanishga ruxsat etish xususida qaror qabul qilish uchun foydalanuvchining, qurilmaning yoki tizimning boshqa tashkil etuvchisining identifikatsiyasini tekshirish; saqlanuvchi va uzatuvchi ma'lumotlarning ruxsatsiz modifikatsiyalanganligini aniqlash uchun tekshirish.	Аутентификация – проверка идентификации пользователя, устройства или другого компонента в системе, обычно для принятия решения о разрешении доступа к ресурсам системы; проверка целостности хранящихся или передающихся данных для обнаружения их несанкционированной модификации.	Authentication – checking the identification of user, device, or other component in the system, typically for decision-making about access to system resources; check the integrity of stored or transmitted data to detect unauthorized modification.
Ахборотни хавфсизлиги – axborot, axborot zaxiralari va axborot tizimlarida talab qilingan ehtimollikda axborot chiqib ketishidan, o'g'irlanishidan, yo'qotilishidan himoya ta'minlanadi.	Безопасность информации – состояние информации, информационных ресурсов и информационных систем, при котором с требуемой вероятностью обеспечивается защита информации от утечки, хищения, утраты и т.д.	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.
Ахборот himoyasi – axborotni soxtalashtirish, buzish, o'g'irlash, qonuniy tarqatishni bartaraf etish harakatlari va vositalari.	Защита информации – действия и средства по предотвращению утечки, хищения, искажения или подделки информации.	Information protection – actions and means to prevent leakage, theft, distortion or falsification of information.

Avtomatlashtirilgan axborot tizimi – tashkiliy-texnik tizim bo'lib, avtomatlashtirilgan axborot texnologiyalarini boshqarish jarayonida va ilmiy-muhandislik ishlarida axborot-tahlil ta'minoti maqsadida ishlatiladi.	Автоматизированная информационно-техническая система – организационно-техническая система, использующая автоматизированные информационные технологии в целях информационно-аналитического обеспечения научно-инженерных работ и процессов управления.	Automated information system – 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.
Avtomatlashtirilgan ma'lumotlar banki – ma'lumotlar bazalarini boshqarish tizimlarining yig'indisi va ularning boshqaruvi ostidagi konkret ma'lumotlar bazasidir.	Автоматизированный банк данных – совокупность систем, управляемых базами данных и конкретной базы (баз) данных, находящейся (находящихся) под ее управлением.	Automated Data Bank - a set of database management system and a specific database (database) data located (are) under its control.
Avtomatik nazorat – apparat vositalari yordamida avtomatik bajariluvchi nazorat.	Автоматический контроль – контроль, выполняемый автоматически аппаратными средствами.	Automatic control – the control performed automatically by the help of hardware.
Avtorizatsiya – tizimda foydalanuvchiga, uning ijobiy autentifikatsiyasiga asosan, ma'lum foydalanish huquqlarini taqdim etish.	Авторизация – представление пользователю определенных прав доступа на основе положительного результата его аутентификации в системе.	Authorization – granting the user certain access rights based on the positive result of authentication in the system

В

Bilim – dasturda foydalaniladigan intellektual axborot.	Знание – это интеллектуальная информация, используемая в программе.	Knowledge – intellectual information used in the program.
Bilimlar – har qanday axborotlar, shu jumladan tizim vaqtning ushbu paytida vazifani yechish-yechmasligidan qat'i nazar tizimda saqlanayotgan aniq dalillar.	Знания – это любая информация, включая конкретные свидетельства, хранящиеся в системе, независимо от того, выполняет ли система задачу в данный момент или нет.	Knowledges – any information, including clear evidence stored in the system, whether or not the system is performing the task at this time.
Bilimlar bazalari – muammoli sohalarning, shuningdek, faktlar oraliq'idagi mantiqiy bog'liqlikning bayon etilishi.	Базы знаний – это описания проблемных областей, а также логическая связь между фактами.	Knowledge bases – a description of the problem areas, as well as the logical connection between the facts.
Bashoratlovchi ekspert tizimlari – o'tnash va bugunning voqealariga asoslanib kelajak ssenariysini oldindan aytib beruvchi tizim.	Прогностические экспертные системы – система, которая предсказывает будущий сценарий на основе прошлых и настоящих событий.	Predictive expert systems – a system that predicts the future scenario based on past and present events.
Bayt – axborot o'lchov birligi, o'zida 8 bit axborotni saqlaydi.	Байт – единица измерения памяти, равняется 8 битам.	Byte – is a unit of memory, equal to 8 bits.
Bit – axborotning eng kichik o'lchov birligi.	Бит – самая малая единица измерения информации.	Bit – is the smallest unit of information measurement.

Bir necha xavfsizlik darajalari – turli xavfsizlik domenlarining resurslari (ayniqsa, saqlanuvchi ma'lumotlar) orasida taqsimlashni tashkil etish va madadlash uchun ishonchli axborot tizimi imkoniyati.	Несколько уровней безопасности – способность доверительной информационной системы, содержать и поддерживать распределение между ресурсами (особенно хранимые данные) различных доменов безопасности.	Multiple security levels – capability of an information system that is trusted to contain, and maintain separation between, resources (particularly stored data) of different security domains.
Brauzer – WWW da ishlash uchun mijoz dasturi.	Браузер – клиентская программа для работы в WWW.	Browser – a client program in WWW.
Blok-sxema – algoritmni tavsiflash uchun grafik yozuv. Dasturiy komponentlarni ishlash chiqishda va ishlashini mantiqiy tahlil qilish jarayonida dasturchilar tomonidan qo'llaniladi.	Блок-схема – графическая нотация для описания алгоритмов. Используется программистами в процессе разработки и анализа логики работы программных компонентов.	Flowchart – graphical notation for the description of algorithms. Used by programmers in the process of developing and analyzing the logic of the operation of software components.
D		
Diagnostika – biror-bir tizimdagi nosozlikni topish jarayoni.	Диагностика – это процесс поиска неисправности в какой-либо системе.	Diagnostics – the process of finding a fault in some system.
Dinamik ekspert tizimi – kelib tushayotgan ma'lumotlarning uzluksiz interpretatsiyasiga ega haqiqiy vaqt rejimidagi obyektlar datchiklar bilan birgalikda ishlaydi.	Динамическая экспертная система – объекты реального времени с непрерывной интерпретацией поступающих данных работают совместно с датчиками.	Dynamic expert system – real-time objects with continuous interpretation of incoming data work in conjunction with sensors.

Dispatcher – bu chiqarish mexanizmining bir qismi, u predmetli bilimlardan qoidalarni qachon va qanday tartibda qo'llashini hal qiladi.	Диспетчер – является частью механизма вывода, который решает, когда и в каком порядке применять правила из предметных знаний.	Dispatcher – it is part of the deduction mechanism that decides when and in what order to apply the rules from the subject knowledge.
Dasturiy interfeys – hisoblash tizimi doirasida qurilma va dasturlar o'zaro ta'sirini ta'minlovchi vositalar yig'indisi.	Программный интерфейс – интерфейс между разными видами программного обеспечения.	Program interface – is the interface between different types of software.
Dasturiy ta'minot – axborotlarni qayta ishlashda dasturlash jarayonini avtomatlashtiradi.	Программное обеспечение – комплекс прикладных программ, с помощью которых на данном рабочем месте выполняются конкретные работы.	Software – a set of application programs, through which specific work is performed in this workplace.
Diagramma – ma'lumotlarni tahlil qilish va solishtirish uchun foydalaniladigan jadval ma'lumotlarining grafik ko'rinishda tasvirlanishi.	Диаграмма – это графическое представление табличных данных, используемых для анализа и сравнения данных.	Diagram – any kind of graphical representation of data in a spreadsheet.
Domen – internetning tashkiliy birligi bo'lib, identifikatsion tugunlarga yoki qarindosh guruhlar tuguniga xizmat qiladi. Katta domenlar turli soha qiziqishi yoki mas'ulligini ifodalovchi domenostiga bo'linadi.	Домен – организационная единица в Интернете, служащая для идентификации узла или группы родственных узлов. Крупные домены могут подразделяться на поддомены, отражающие различные области интересов или ответственности.	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.

Drayver – amaliyot tizimining imkoniyatlarini oshirish hamda kiritish-chiqarish qurilmalarini boshqarish uchun mo'ljallangan dastur.	Драйвер – программа, обеспечивающая взаимодействие компьютера с внешним устройством.	Driver – is a program that allows the computer to communicate with an external device.
Е		
Ekspert – bu aniq predmetli sohada samarali qarorlarni to'ra oluvchi mutaxassis.	Эксперт – это специалист, который может найти эффективные решения в определенной предметной области.	Ekspert – it is an expert who can find effective solutions in a specific subject area.
Ekspert tizimi – mutaxassislarning aniq predmetli sohalaridagi bilimlarini jamlovchi va bu empirik tajribani kamroq malakali foydalanuvchilarga maslahat berish uchun nusxa ko'paytiruvchi murakkab dasturiy majmualardir.	Экспертные системы – это сложные программные пакеты, которые объединяют знания экспертов в конкретных предметных областях и воспроизводят этот эмпирический опыт для консультирования менее опытных пользователей.	Expert system – are complex software packages that consolidate the knowledge of experts in specific subject areas and replicate this empirical experience to advise less skilled users.
Elektron jadvallar – jadval ko'rinishidagi ko'p miqdordagi ma'lumotlarni qayta ishlashda turli hisob operatsiyalarini bajarish sifatini hamda hisoblashlar darajasini oshirish uchun ishlatiladigan jadval muharriri deb ataladigan maxsus dasturlar pakeiti.	Электронные таблицы – программа, обрабатывающая таблицы, состоящие из строк и граф, на пересечении которых располагаются клетки. В клетках содержится числовая информация, формулы или текст. Значение в числовой клетке таблицы либо записано, либо расчитано по формуле.	Spreadsheet – is a program that processes tables, consisting of rows and graph, at the intersection of which the cells are located. The cells contain numerical information, formulas or text. The value in the numerical cell of the table is either written, or calculated by the formula. In the formula, there may be calls to other cells.

E-mail – elektron pochta. Elektron pochta maxsus dastur bo'lib, uning yordamida internet tarmog'i orqali dunyoning ixtiyoriy joyidagi elektron manzilga xat, hujjat, ya'ni ixtiyoriy ma'lumotni tezda jo'natish va qabul qilib olish mumkin.	E-mail – система пересылки сообщений между пользователями вычислительных систем, в которой компьютер берёт на себя все функции по хранению и пересылке сообщений.	E-mail – system for sending messages between users of computer systems, in which the computer takes care of all the functions of storing and sending messages.
Ekspert tizimi – ekspert bilimlarini tor doiradagi masalalarni samarali yechishda foydalanuvchi kompyuter dasturi.	Экспертная система – система искусственного интеллекта, включающая знания об определенной слабо структурированной и трудно формализуемой узкой предметной области и способная предлагать и объяснять пользователю разумные решения.	Expert system – is an artificial intelligence system that includes knowledge of a certain weakly structured and difficultly formalized narrow subject area and is able to offer and explain reasonable solutions to the user.
F Freyim (inglizcha Frame – qolip yoki ramka) – bu qandaydir andazaviy vaziyat yoki abstrakt obrazni taqdim etish uchun mo'ljallangan bilimlarining tuzilishi.	Фрейм – это структура знаний, предназначенная для представления стандартной ситуации или абстрактного изображения.	Frame (English Frame - mold or frame) – it is the structure of knowledge designed to present some standard situation or abstract image.

Fayl – nomiga, kengaytmaga va hajmiga ega bo'lgan obyekt bo'lib, tashqi xotirada saqlanadi.	Файл – именованная совокупность любых данных, размещенная на внешнем запоминающем устройстве и хранимая, пересылаемая и обрабатываемая как единое целое.	File – a named collection of any data stored on an external storage device and stored, forwarded and processed as a whole.
Fayl atributi – faylni belgilovchi xarakteristika: ismi, o'ichami, tuzilishi (turi), foydalanish usuli, yozilish uzunligi, yozilish xili va h.k.	Атрибут файла – характеристика, определяющая файл: имя, размер, организация (тип), метод доступа, длина записи, тип записи и др.	File attribute – characteristic that determines the file name, size, organization (type), method of access, record length, record type, etc.
Fayllarni jo'natish – butun faylni yoki uning qismini ochiq tizimlar orasida ko'chirish tartibi.	Пересылка файлов – процедура перемещения содержимого всего файла или его части между открытыми системами.	File transfer – the procedure for moving the contents of the whole file or its part between open systems.
Forma – elektron blank tarzida ifodalanib, unda ma'lumotlar kiritiladigan maydon mavjud, shu maydonlarga kerakli ma'lumotlar joylashtiriladi va jadval shu tariqa hosil qilinadi.	Форма – это специальное средство для ввода данных, предоставляющее конечному пользователю возможность заполнения только тех полей базы данных, к которым у него есть право доступа.	Form – is a special tool for data entry, which allows the end user to fill only those fields of the database to which he has the right of access.
Foydalanuvchi interfeys – foydalanuvchining dasturiy yoki kompyuter bilan o'zaro ta'siridagi dasturiy va apparat vositalari.	Пользовательский интерфейс – интерфейс между пользователем и программно-аппаратными средствами компьютера.	User interface – is the interface between the user and the firmware of the computer.

G	Gipermatn – interaktiv muhit turini tasvirlash tushunchasi boʻlib, murojaatlarga oʻtish imkoniyatini bajaradi. Soʻzlar, iboralar yoki rasmlarga oʻrnatilgan murojaatlarni (URL manzil formati), foydalanuvchi (murojat ushida sichqonchani chap tugmasini bosib) tanlashi matn yoki rasm va darhol tegishli maʼlumotlarni va multimedia materiallarini olib chiqish imkonini beradi.	Гипертекст – понятие, описывающее тип интерактивной среды с возможностями выполнения переходов по ссылкам. Ссылки (адреса формата URL), 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.	Hyperlink – a concept that describes the type of interactive environment with the ability to perform referrals. Links (URLs of the URL format) embedded in words, phrases or pictures allow the user to select (set the pointer and click the left mouse button) the text or drawing and immediately display the related information and multimedia materials.
	Gipermurojaat – bir yoki turli kompyuterlarda joylashgan, hujjat ichidagi va tashqarisidagi turli axborot komponentlari orasidagi aloqa.	Гиперссылка – элемент документа для связи между различными компонентами информации внутри самого документа, в других документах, в том числе и размещенных на различных компьютерах.	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.

I

Interfeys (ma'lumotlarni ishlash tizimida) – professor tomonidan taqdim etiladigan muayyan xizmatlar jamlanmasi.	Интерфейс – это часть системы базы знаний, которая обеспечивает достаточно высокий уровень языка, близкий к профессиональному языку профильных специалистов, которые работают с базой знаний и механизмом принятия решений.	Interface – it is part of a knowledge base system that provides a sufficiently high level of language that is close to the professional language of subject matter professionals who work with the knowledge base and decision-making mechanism.
Interpreter – ekspert tizimining bir qismi bo'lib, bazadagi bilimlarni ma'lum bir tartibda qayta ishlaydi.	Интерпретатор – это часть экспертной системы, которая обрабатывает знания в базе данных в определенном порядке.	Interpreter – is part of the expert system and processes the knowledge in the database in a certain order.
Internet – kompyuterlar orasida axborot almashishini ta'minlaydigan, o'zaro bog'langan kompyuterlar tarmog'i bo'lib, ochiq jahon axborot tizimini tashkil qiladi.	Интернет – открытая мировая информационная система, состоящая из взаимосвязанных компьютерных сетей, обеспечивающая доступ к удаленной информации и обмен информацией между компьютерами.	Internet – the global communication network that allows almost all computers worldwide to connect and exchange information.
Internet provayder – foydalanuvchilarga internetga kirishni ta'minlovchi tashkilot.	Интернет-провайдер – организация, предоставляющая пользователям доступ к Интернету.	An Internet service provider – is a company that provides customers with Internet access.

К		
Katalog – fayllar nomi, hajmi, yozilgan vaqti kabi ma'lumotlarni saqlovchi diskdagi maxsus joy.	Каталог – оглавление файлов. Его можно просматривать, переименовывать зарегистрированные в нём файлы, переносить их содержимое на новое место и удалять.	Catalog – table of contents of files. It can be viewed, renamed files registered in it, move their contents to a new location and delete them.
Keys-technologie – o'quv jarayonida ishlatiladigan ixtiyoriy o'quv-uslubiy materiallar texnologiyasi va talabaga mustaqil ishlarni berish va qabul qilish to'plami.	Кейс-технология – технология организации 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).	Case study – is an account of an activity, 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.

Konfidensiallik – 1. Avtorizatsiya qilgan shaxs tomonidan olinishi yoki foydalanishi tashkilot uchun jiddiy zarar sababi bo'la olmaydigan ma'lumotlarning qandaydir sinfi. 2. Alohida shaxslar, modullar, jarayonlar ruxsatsiz aniqlanishi va foydalanishi mumkin bo'lmagan axborot xususiyati.	Конфиденциальность – 1. Некий класс данных, получение либо использование которых неавторизованными для этого лица не может стать причиной серьезного ущерба для организации. 2. Свойство информации, состоящее в том, что она не может быть обнаружена и сделана доступной без разрешения отдельным лицам, модулям или процессам.	Confidentiality – 1. Some class data, obtaining or the use of which by unauthorized persons could not cause serious damage to the organization. 2. The quality of information, consisting in that it cannot be detected and made available without the permission of individuals, modules or processes.
L		
Loyihalashtirish – oldindan belgilangan xususiyatlarga ega «obyektlar»ni loyihalash uchun mutaxassislarni tayyorlash.	Проектирование – это обучение специалистов проектированию «объектов» с заранее заданными характеристиками.	Design – training in the preparation of «objects» with predetermined characteristics.
M		
Mantiqiy modellar – bilimlarning alohida birliklari (dalillar) o'rtasidagi munosabatlar rasmiy nazariyaning sintaktik bilimlari yordamida aks ettiradi.	Логические модели – отношения между отдельными единицами знания (свидетельства) отражаются с помощью синтаксических знаний формальной теории.	Logical models – the relationship between the individual units of knowledge (evidence) is reflected using the syntactic knowledge of formal theory.
Mantiqiy xulosalar bloki – qoidalar-ni faktlar bilan qiyoslagan holda xulosalar mantiq'ini yuzaga keltirish.	Блок логических выводов – это формирование логики выводов путем сравнения правил с фактами.	A block of logical conclusions – o formulate the logic of conclusions by comparing the rules with the facts.

Ma'lumotlar – joriy vaqtda yetchi- layotgan vazifa haqidagi dastlabki, oralik va yakuniy (natijaviy) axborot- lar.	Данные – это начальная, промежу- точная и окончательная (итоговая) информация о решаемой задаче.	Information – initial, intermediate and final (final) information about the task currently being solved.
Monitoring – vaqtning haqiqiy qo'l- lashida ma'lumotlarni uzluksiz in- terpretatsiyalangan va u yoki bu o'l- chamlarning yo'l qo'yiladigan che- garadan tashqariga chiqishi haqida xabar berishdir.	Мониторинг – это отчет о факти- ческой интерпретации данных в приложении в реальном времени и отклонении того или иного параме- тра.	Monitoring – in the actual use of time, it is to report that the data has been continuously interpreted and that one or another dimension has gone beyond the allowable limit.
Ma'lumotlar bazasi – tatbiqiy dasturlarga bog'liq bo'lmagan hol- da ma'lumotlarni tavsiflashning, saqlashning va manipulyatsiyalash- ning umumiy prinsiplarini ko'zda tutuvchi, ma'lum qoidalar bo'yicha tashkil etilgan ma'lumotlar majmuvi. Predmet sohasining axborot modeli hisoblanadi.	База данных – совокупность дан- ных, организованных по опреде- ленным правилам, предусматрива- ющим общие принципы описания, хранения и манипулирования дан- ными, независимо от прикладных программ. Является информационной моделью предметной области.	Database – a collection of data or- ganized according to certain rules, providing general principles for de- scribing, storing and manipulating data independent of the application programs. An information domain model.
Ma'lumotlar modeli – ma'lumotlar bazasida ma'lumotlarni tashkil etish- ning mantiqiy ifodasi.	Модель данных – логическое представление организации дан- ных в базе данных.	Data model – a logical view of data organization in the database.
Ma'lumotlar avtorizatsiyasi – ma'lumot- larning ma'lumotlar bazasiga te- gishli darajasini aniqlash va belgilash.	Авторизация данных – определе- ние и установление степени при- ватности данных в базе данных.	Authorization data – a determination and an establishment of the level of privacy of the data in the database.

Ma'lumotlar bazasi ma'muri – ma'lumotlar bazasi xususida to'liq tasavvurga ega, undan foydalanish va rivoji uchun javobgar maxsus lavozimli shaxs (shaxslar guruhi). Ma'lumotlar banki ma'muriyati tarkibiga kiradi.	Администратор базы данных – специальное должностное лицо (группа лиц), имеющее(ая) полное представление о базе данных и отвечающее за ее ведение, использование и развитие. Входит в состав администрации банка данных.	Database administrator – a special official person (a group of people) that has a complete picture of the database system and responsible for its management, operation and development, which contains to the administration system of the data bank.
Ma'lumotlar banki – ma'lumotlar-ni markazlashgan saqlashning va kollektiv foydalanishning avtomatlashirilgan axborot tizimi. Bank tarkibiga bir yoki bir necha ma'lumotlar bazasi, ma'lumotlar bazasini boshqarish tizimi hamda so'rovlar va tatbiqiy dasturlar kutubxonasi kiradi. Yana – foydalanuvchilarning ma'lum guruhiga, ma'lum tematika bo'yicha ma'lumotlarni saqlash va qidirish xizmatlarini taqdim etish tizimi.	Банк данных – автоматизированная информационная система централизованного хранения и коллективного использования данных. В состав банка данных входят одна или несколько баз данных, справочник баз данных, система управления базами данных, а также библиотеки запросов и прикладных программ. Еще – система, предоставляющая услуги по хранению и поиску данных определенной группе пользователей по определенной тематике.	Databank – automated information system of centralized storage and sharing of data. In the composition of the data bank includes one or more databases, reference databases, the database management system and libraries of queries and applications. Yet, the system that provides services for the storage and retrieval of data in a certain group on a certain topic.
Multimedia – tovushli va video yozuvlarni kompyuter tizimi orqali bevosita boshqarish.	Мультимедиа – компьютерные системы с интегрированной поддержкой звукозаписей и видеозаписей.	Multimedia – is the field concerned with the computer-controlled integration of text, graphics, drawings, still and moving images (Video), animation, audio.

N		
Notarizatsiya – ma'lumotlarni, uning tarkibi, uzatuvchisi, uzatish vaqti va qabul qiluvchisi kabi xarakteristikalarining haqiqiylikini tasdiqlashni ta'minlovchi ishonchli uchinchilarning shaxs tomonidan qayd qilinishi.	Нотаризация – регистрация данных доверенным третьим лицом, которое обеспечивает последующее подтверждение правильности их характеристик, таких как содержание, отправитель, время и получатель.	Notarization – check data with a trusted third party that provides subsequent confirmation of the correctness of their characteristics such as content, sender, time and recipient.
Nul – uzatish yoki uzatishning xavfsizlik maqsadlari uchun deshiflashni sekinlashtirish yoki to'xtatish yoki shifrlangan guruhlar tugatilishi uchun shifrlangan xabarga qo'yiluvchi soxta harf, harf simvoli yoki kod guruhi.	Нуль – фиктивная буква, символ буквы или код группы, вставляемые в шифрованное сообщение, чтобы замедлить или предотвратить его расшифровку или для завершения шифрованной группы в целях передачи или обеспечения безопасности передачи.	Null – dummy letter, letter symbol, or code group inserted into an encrypted message to delay or prevent its decryption or to complete encrypted groups for transmission or transmission security purposes.
O		
Ochiq testlash – tashkilotning AT xodimi xabardorligi va roziligi bilan o'tkaziluvchi xavfsizlikni testlash.	Открытое тестирование – тестирование безопасности, проводимое с ведома и согласия ИТ-персонала организации.	Overt testing – security testing performed with the knowledge and consent of the organization's IT staff.

Ochiq saqlash – tasdiqlangan kon-teynerdan tashqaridagi milliy xavfsizlikka tegishli maxfiy axborotni saqlovchi har qanday joy.	Открытое хранение – любое место хранения секретной информации, касающейся национальной безопасности за пределами утвержденных контейнеров.	Open storage – any storage of classified national security information outside of approved containers.
Obyektga yo'naltirilgan ma'lumotlar bazasi – obyektga yo'naltirilgan ma'lumotlar modeliga asoslangan ma'lumotlar bazasi. Obyektga yo'naltirilgan modelning asosiy elementlari: sinflar, obyektlar, interfeyslar, atributlar (xususiyatlar), usullar va boshqalar.	Объектно-ориентированная база данных – база данных, основанная на объектно-ориентированной модели данных. Главными элементами объектно-ориентированной модели являются классы, объекты, интерфейсы, атрибуты (свойства), методы и т.п.	Object-oriented database – is a database based on an object-oriented data model. The main elements of the object-oriented model are classes, objects, interfaces, attributes (properties), methods, and so on.
Obyektga yo'naltirilgan dasturlash – dasturni obyektlar to'plami sifatida ko'rsatishga mo'ljallangan dasturlash metodologiyasi bo'lib, ularning har biri ma'lum bir sinfning namunasi sifatida ko'riladi. Sinflar merosning iyerarxiyasini tashkil qiladi.	Объектно-ориентированное программирование (ООП) – методология программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определенного класса, а классы образуют иерархию наследования.	Object-oriented programming (OOP) – programming methodology based on the representation of the program in the form of a collection of objects, each of which is an instance of a particular class, and the classes form a hierarchy of inheritance.

Р

Parol – talabgor o'zining haqiqiyligini tekshirish uchun xotirasida saqlagan va foydalanadigan maxfiy katta-lik. Yana – tizimdan, dasturdan yoki ma'lumotlardan foydalanishga ruxsat olish uchun kompyuter so'rovi bo'yicha kiritiladigan simvollarning noyob ketma-ketligi.	Пароль – условное слово или набор знаков, предназначенный для подтверждения личности или полномочий. Пароли часто используются для защиты информации от несанкционированного доступа. В большинстве вычислительных систем комбинация «имя пользователя пароль» используется для удостоверения пользователя.	Password – a secret that a claimant memorizes and uses to authenticate his or her identity. Also, a unique sequence of characters that must be entered but the request of the computer to get access to the system, program or data.
PowerFactory – energiya tizimlari rejimini tahlil qiluvchi zamonaviy yetakchi kompleks hisoblovchi dasturiy ta'minot.	Повер факторий – программное обеспечение, являющееся ведущим комплексным решением для планирование и анализа режимов современных энергосистем.	Power Factory – software which is leading complex solution for planning and analyzing modes of modern power systems
Passiv hujum – kriptotizimga yoki kriptografik protokolga hujum bo'lib, bunda dushman va/yoki buz'unchi uzatiluvchi shifrlangan axborotni kuzatadi va ishlatadi, ammo qonuniy foydalanuvchilar harakatiga ta'sir etmaydi.	Атака пассивная – атака на криптосистему или протокол криптографический, при которой противник и/или нарушитель наблюдает и использует передаваемые сообщения шифрованные, но не влияет на действия пользователей законных.	Passive attack – attack on a cryptosystem or a cryptographic protocol in which enemy and/or the offender observes and uses the transmitted messages are encrypted, but does not affect the user's actions legitimate.

Q		
Qaytarish – masofadagi axborot tizimi terminalini identifikatsiyalash va autentifikatsiyalashdagi jarayon bo'lib, uning yordamida host tizim terminalini uzadi va aloqani qaytadan tiklaydi.	Возвратить – процедура идентификации и аутентификации удаленного терминала информационной системы, посредством чего хост-система разъединяет терминал и восстанавливает контакт.	Call back – procedure for identifying and authenticating a remote information system terminal, whereby the host system disconnects the terminal and reestablishes contact.
Qoldiq – axborotni ishlash amallari tugallanganidan so'nggi, ammo joyini tozalash yoki qayta yozishdan oldingi xotirada qolgan ma'lumotlar.	Остаток – данные, остающиеся в памяти, когда операции по обработке данных закончена, но до момента размагничивания или переписи данных на это место.	Residue – data left in storage after information-processing operations are complete, but before degaussing or overwriting has taken place.
R		
Raqamli axborot – kompyuter tizimlarida ishlashga, saqlashga va almashishga mo'ljallangan ma'lumotlar ko'rinishida ifodalangan axborot.	Цифровая информация – информация, представленная в виде данных, предназначенных для обработки, хранения и обмена в системах компьютерных.	Digital information – the information presented in the form of data intended for processing, storage and sharing of computer systems.

Raqamli pullar – kompyuter tarmog'ida o'zaro hisob-kitob uchun ishlatiladigan va faqat elektron shaklda (elektron naqd pul, elektron tangga) mavjud shifrlangan yozuvlardan (elektron hamyondagi) iborat to'lov vositalari.	Цифровые деньги – платежные средства, представляющие собой шифрованные записи (в т.н. бумажнике электронном), используемые для взаиморасчетов в компьютерной сети и существующие исключительно в электронной форме (электронная наличность, монета электронная).	Digital money – means of payment representing the encrypted records (so-called e-wallet), used for settlements in a computer network, and existing solely in electronic form (electronic cash, electronic coin).
Raqamli imzo kaliti – elektron imzoni yaratishga mo'ljallangan simvollarning noyob ketma-ketligi.	Ключ цифровой подписи – уникальная последовательность символов, предназначенная для создания электронной подписи.	Signature creation key – a unique sequence of symbols intended for creation of electronic signature.
S		
Slot – bu obyekt xususiyatini bayon qilish atributi.	Слот – это атрибут, описывающий свойство объекта.	Slot – this is an object property statement attribute.
Statistik ekspert tizimlari – predmetli sohalarda ishlab chiqiladi, unda bilimlar bazasi va interpretatsiyalanayotgan ma'lumotlar vaqtda o'zgar-maydi hamda ular barqarordir.	Статистические экспертные системы – разрабатываются в предметных областях, где база знаний и интерпретируемые данные не меняются со временем и являются стабильными.	Statistical expert systems – are developed in subject areas where the knowledge base and the data being interpreted do not change over time and are stable.

Sun'iy intellekt – bu kompyuterda insonning fikrlash jarayonini taklif qiluvchi dasturiy vositalar tizimi; aqliy xatti-harakatlarga nisbatan kompyuter tizimining qobiliyati tushuniladi.	Искусственный интеллект – это система программного обеспечения, которая предлагает процесс мышления человека на компьютере; способность компьютерной системы реагировать на психическое поведение.	Artificial intelligence – it is a system of software tools that offers the process of human thinking on a computer; the ability of a computer system to respond to mental behavior.
Spam – so'ralmagan hajmiy xabarlarini asossiz yuboruvchi elektron no-maqbul pochta yoki elektron xabar almashish tizimlarining noto'g'ri ishlatilishi.	Спам – электронная нежелательная почта или неправильное обращение с электронными системами обмена сообщениями, когда без разбора посылаются незапрашиваемые объемные сообщения.	Spam – electronic junk mail or the abuse of electronic messaging systems to indiscriminately send unsolicited bulk messages.
Spamming – UNINETning turli guruhlariga katta sonli bir xil xabarlarini jo'natish. Ko'pincha arzon reklama shirkatini, piramidalarni tashkil etish yoki odamlar joniga tegish maqsadida ishlatiladi.	Спамминг – посылка большого числа одинаковых сообщений в различные группы UNINET. Часто используется для организации дешевой рекламной компании, пирамид или просто для надоедания людям.	Spamming – sending large numbers of identical messages to different groups UNINET. Often is used for cheap advertising campaign, the pyramids, or just for annoying people.
Т		
Tashxislovchi ekspert tizimlari – kuzatiladigan hodisalarning normal emasligi sabablarini topish xususiyatiga ega tizim.	Диагностические экспертные системы – это системы, которые имеют возможность находить причины отклонений в наблюдаемых явлениях.	Diagnostic expert systems – a system capable of finding the causes of abnormalities of observed phenomena.

Tizimni yaratish moduli – qoidalar to'plamini yaratish uchun xizmat qiladi.	Модуль создания системы – используется для создания набора правил.	System creation module – serves to create a set of rules.
Tushuntirish (izohlash) bloki – foydalanuvchining texnologiyada bilimlar bazasidan foydalanish ketma-ketligini aks ettirish.	Блок объяснения – представляет собой последовательность использования пользователем базы знаний в области технологий.	Explanatory block – reflect the sequence in which the user uses the knowledge base in technology.
Tarmoqlararo ekran – tarmoqlararo ekran yoki firewall tizimi deb ham ataluvchi tarmoqlararo himoyaning ixtisoslashtirilgan kompleksi.	Экран межсетевой – специализированный комплекс межсетевой защиты, называемый также брандмауэром или системой firewall.	Screen firewall – a specialized set of network security called a firewall, or a firewall system.
Tizim xavfsizligi – axborot tizimining ruxsatsiz foydalanishga urinishga qarshi tura olish xususiyati.	Безопасность системы – свойство информационной системы противостоять попыткам несанкционированного доступа.	System security – a property information system to resist unauthorized access attempts.
Virtual haqiqat – muloqotsiz axborot hamkorligining yangi texnologiyasi bo'lib, kompleks multimedia amaliyot vositalari yordamida real vaqt oralig'ida «dunyo ekrani»ga kirish illuziyasini ta'minlaydi. Bu foydalanuvchi tasavvuridagi batafsil mavhum xayoliy dunyodir.	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.	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.

Virtual kutubxona – global internet tarmog'iga joylashtirilgan o'quv-uslubiy va qo'shimcha adabiyot.	Виртуальная библиотека – учебно-методическая и дополнительная литература, размещенная в глобальной сети интернет.	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.
X		
Xulosa chiqarish mexanizmi – intellektning yangi dalillarni chiqarib olishga yordamlashadigan qismi.	Механизм рассуждения – это часть интеллекта, которая помогает извлекать новые доказательства.	Conclusion mechanism – the part of the intellect that helps to extract new evidence.
O'		
O'rgatuvchi ekspert tizimlari – foydalanuvchilarga berilgan sohada tashxis qo'yish va tahlil etish imkoniyatini beradigan tizim.	Образовательные экспертные системы – это системы, которые позволяют пользователям диагностировать и анализировать заданную область.	Teaching Expert Systems – a system that allows users to diagnose and analyze a given area.

1. *Nigmatov H.* Intellectual tizimlar. O'quv qo'llanma. – T.: Sano-standart, 2019.
2. *Швайко И.Г., Букача Л.Н., Шаповаленко В.А.* Графика в C++Builder. – Одесса: ОНАС, 2012.
3. *Малюк А.А.* Защита информации в информационном обществе: Учебное пособие для вузов. – М.: ГЛТ, 2015.
4. *Антамошин А.Н.* Интеллектуальные системы управления организационно-техническими системами. – М.: РиС, 2016.
5. *Быховский М.А.* Развитие телекоммуникаций. Учебное пособие. – М.: 2017.
6. *Крухмалев В.В., Гордиенко В.Н.* Цифровые системы передачи. – М.: Горячая линия-Телеком, 2018.
7. *Nigmatov H.* Intellectual tizimlar. O'quv qo'llanma. – T.: Sano-standart, 2019.
8. *Kunwoo Lee.* Principles of CAD/CAM/CAE: The Computer Aided Engineering Design Series. 5st Edition. Addison Wesley Longman, USA, 2015.
9. *Kadirov M.M.* Texnik tizimlarda axborot texnologiyalari. Darslik, 2-qism. – T.: O'zbekiston faylasuflari milliy jamiyati, 2019.
10. *Dadabayeva R.A., Nasridinova Sh.T., Shoaxmedova N.X., Ibragimova L.T., Ermatov Sh.T.* Axborot-kommunikatsiya texnologiyalari va tizimlari. O'quv qo'llanma. – T.: Sano-standart, 2017.
11. *Kenjabayev A.T., Ikromov M.M., Allanazarov A.Sh.* Axborot-kommunikatsiya texnologiyalari. O'quv qo'llanma. – T.: O'zbekiston faylasuflari milliy jamiyati, 2017.
12. КОМПАС-3D Версия 15. Руководство пользователя. М.: АСКОН, 2014.
13. КОМПАС-ЭЛЕКТРИК. Руководство пользователя. – М.: АСКОН, 2014.
14. *Dadajonov T., Muxitdinov M.* MATLAB asoslari. – T.: O'zFA «Fan» nashriyoti, 2008.
15. *Акулич М.* Искусственный интеллект и маркетинг. Цифровая книга. – М.: Издательские решения, 2018.
16. *Алексеев А.С.* Многоуровневая кибербезопасность интеллектуальных сетей. Вестник современных исследований. – М., 2019.

17. Зализный Д.И. Модель фотоэлемента для библиотеки SimPowerSys-пакет temsMATLAB/Simulink/. – СПб.: Питер, 2011.

18. Леонтьев В. Безопасность в сети Интернет. – М.: ОЛМА Медиа Групп, 2008.

19. Бузов Г.А. Защита информации ограниченного доступа от утечки по техническим каналам. – М.: ГЛТ, 2016.

20. Nazirov Sh.A., Qobulov R.V., Bobojonov M.R., Raxmanov Q.S. C va C++ tili. – T.: Voris-nashriyot, 2013.

Elektron resurslar

1. <http://sunny.ccas.ru/library.html>

2. <http://www.MathCad.com>

3. www.lex.uz

4. www.ZiyoNet.uz

5. www.uzenergy.uzpak.uz

6. <http://litmir.biz/rd/268495>

7. www.twirpx.com

MUNDARIJA

Kirish 3

1-BOB. TEXNIK TIZIMLARDA AXBOROT TEXNOLOGIYALARI FANINING PREDMETI VA VAZIFALARI

- 1.1. Axborot texnologiyalari va tizimlarining nazariy asoslari5
- 1.2. Axborot-kommunikatsiya texnologiyalarining asosiy vazifalari va
tarmoq sohalarda qo'llanilishi 9
- 1.3. Kompyuter aloqa tizimlar 11
- 1.4. Raqamli texnologiyalar tushunchasi va uning ahamiyati12
- 1.5. Raqamli iqtisodiyot tushunchasi va uning ahamiyati15

2-BOB. INTELEKTUAL TIZIMLARNING ASOSIY TUSHUNCHALARI

- 2.1. Intellektual tizimlarning maqsadi, funksiyalari va uslubiyati19
- 2.2. Ekspert tizimlari. Ekspert tizimlarining sinflanishi va struktura
tuzilishi 21
- 2.3. Ekspert tizimining afzalliklari va tizimni yaratish moduli23
- 2.4. Sun'iy intellektning asosiy tushunchalari 26
- 2.5. Sun'iy intellekt strukturasining tuzilishi 28

3-BOB. ZAMONAVIY AVTOMATLASHTIRILGAN LOYIHALASH TIZIMLARI VA ULARNING TEXNIK SOHALARDA QO'LLANILISHI

- 3.1. Loyihalash jarayonlari va bosqichlari 30
- 3.2. Avtomatlashtirilgan loyihalash tizimlarining strukturasi31
- 3.3. Loyihalash jarayonlaridagi CAE/CAD/CAM-tizimlari36
- 3.4. CALS-texnologiyalarni qo'llash sohalari 37

4-BOB. MATEMATIK MODELASHTIRISH, SONLI TAHLIL USULLARINI TIZIMLARDA AMALGA OSHIRISH

- 4.1. Matematik modelashtirish va ularning turlari41
- 4.2. Hisobli eksperiment. MathCAD dasturida statik modellarni
amalga oshirish 45

4.3. MathCAD dasturning ishchi sohasi va uskunalar paneli	47
4.4. MathCad tizimida ishlatiladigan ma'lumotlarning turlari	55
4.5. MathCad tizimida funksiyalarning ishlatilishi	56
4.6. MathCad tizimida masalalarni yechish usuli va vositalari	58

5-BOB. MATLAB DASTURLASHTIRISH TIZIMI

5.1. MATLAB tizimida statik va dinamik modellarni amalga oshirish	65
5.2. MATLAB dasturining boshqa dasturiy tizimlar bilan integratsiyalashuvi	67
5.3. Matritsalar bilan ishlash imkoniyatlari	68
5.4. MATLAB tizimining dialog rejimida ishlash	70
5.5. MATLAB tizimida matritsalar ustida amallar bajarish	76
5.6. Matritsalar ustida bajariladigan buyruqlar xususiyatlari	79
5.7. MATLAB tizimida matritsa elementlarini transponirlash	82
5.8. MATLAB dasturlashtirish tizimida grafiklarni yaratish va formatlash. Ikki o'lchovli grafika	83
5.9. Gistogrammalar. Polyar koordinatalarda grafika	85
5.10. MATLAB dasturlashtirish tizimida uch o'lchamli grafiklarni qurish va formatlash	90

6-BOB. BOB. IMITATSION MODELLASHTIRISH VA ULARNING TURLARI

6.1. Simulink tizimida imitatsion modellar yaratish	93
6.2. MATLAB tizimidagi Simulink kutubxonasining bloklari	95
6.3. Simulink paketining model oynasi	97
6.4. MATLAB tizimining Simulink paketida model yaratish	99
6.5. Power System Blockset paketida model yaratish	106
6.6. Power System Blockset paketi kutubxonasining bloklari	107

7-BOB. ZAMONAVIY AVTOMATLASHTIRILGAN LOYIHALASH TIZIMLARIDA GRAFIK MODELLASHTIRISH

7.1. «Компас-3Д» dasturida modellash tirish	119
7.2. «Компас электрик»da sxemalar yaratish	130
7.3. «Компас электрик»da kataloglar bilan ishlash	133

8-BOB. «BULUTLI TEXNOLOGIYA»LARNING TAVSIFI VA TUSHUNCHASI

- 8.1. «Bulutli texnologiya»larda SaaS, PaaS ba IaaS modellari138
- 8.2. Ma'lumotni saqlash va zaxiralash 144
- 8.3. Bulutli texnologiya xizmatlarining afzalliklari145

9-BOB. TEXNIK TIZIMLARDA AXBOROTNI HIMOYA QILISHNING ZAMONAVIY USULLARI

- 9.1. Texnik tizimlarda axborot xavfsizligini ta'minlash147
- 9.2. Axborot xavfsizligini yaratish vositalari 150
- 9.3. Axborotlarni texnik himoyalash 152
- 9.4. Ochiq va yopiq kalitlar bilan shifrlash tizimi153
- 9.5. Tarmoq va kompyuter viruslari, ularning turlari155

10-BOB. DASTURLASH TEXNOLOGIYALARI. OBYEKTGA YO'NALTIRILGAN DASTURLASH TILLARI

- 10.1. Dasturlash asoslari va uning imkoniyatlari161
- 10.2. Dasturlash tillari va ularning turlari 164
- 10.3. Obyektga yo'naltirilgan Borland C++ Builder 6 tizimi169
- 10.4. C++ Builder 6 dasturlash tilining asosiy konstruksiyalari173
- 10.5. Texnik tizimlardagi ma'lumotlarning toifalari strukturasi177
- 10.6. C++ dasturlash tilining standart kutbxonalari184

11-BOB. C++ BUILDER 6 DASTURLASH TIZIMIDA CHIZIQLI DASTURLARNING QO'LLANILISHI

- 11.1. Borland C++ builder 6 tizimidagi operatorlar tasnifi187
- 11.2. C++ tilida ma'lumotlarni kiritish va chiqarish189
- 11.3. Chiziqli hisoblash jarayonlarni dasturlash192
- 11.4. Forma ilovasida chiziqli jarayonlarni dasturlash195

12-BOB. C++ BUILDER 6 DASTURLASH TIZIMIDA MANTIQUIY DASTURLASH TEXNOLOGIYASI

- 12.1. C++ builder 6 dasturlash tizimida shartsiz o'tish operatori200
- 12.2. C++ builder 6 dasturlash tizimida shartli o'tish operatori201
- 12.3. C++ builder 6 dasturlash tizimida tanlash operatori208

13-BOB. C++ BUILDER 6 DASTURLASH TIZIMIDA TAKRORLANUVCHI STRUKTURALI DASTURLAR

- 13.1. Takrorlash operatorlari 216
- 13.2. While operatorining qo'llanilishi 217
- 13.3. Do...while operatorining qo'llanilishi 222
- 13.4. For operatorining qo'llanilishi 227

14-BOB. GRAFIK MODULINING IMKONIYATLARI VA ULARDAN FOYDALANISH

- 14.1. Tasvirlar bilan ishlash 233
- 14.2. Dasturning asosiy grafik funksiyalari 234
- 14.3. Grafik tasvirlar yaratishda mo'yqalam bilan ishlash237
- 14.4. Oddiy geometrik figuralar chizish 243

15-BOB. TEXNIK TIZIMLARDA INTEGRALLASHGAN MUHITDA MA'LUMOTLAR BAZASINI YARATISH VA QAYTA ISHLASH VOSITALARI

- 15.1. Ma'lumotlar bazasining texnologiyalari 256
- 15.2. Integrallashgan muhitda ma'lumotlar bazasini yaratish258
- 15.3. Ma'lumotlar bazasini qayta ishlash 263
- Glossariy 273
- Foydalanilgan adabiyotlar 297

Darslik

Shohida Azatovna Akbarova

TEXNIK TIZIMLARDA AXBOROT TEKNOLOGIYALARI

Muharrir
Texnik muharrir
Badiiy muharrir
Kompyuterda sahifalovchi

T. Mirzayev
R. Ahmedov
D. Mulla-Axunov
G. Ahmedova

Nash.lits. № AA 0049. 18.03.2020.

Bosishga 26.08.2022-yilda ruxsat etildi. Bichimi 60x84 1/16.

Ofset bosma. «Times New Roman» garniturasini.

Bosma tabog'i 19. Shartli b.t. 17,67.

Adadi 200 nusxa. Buyurtma № 6.

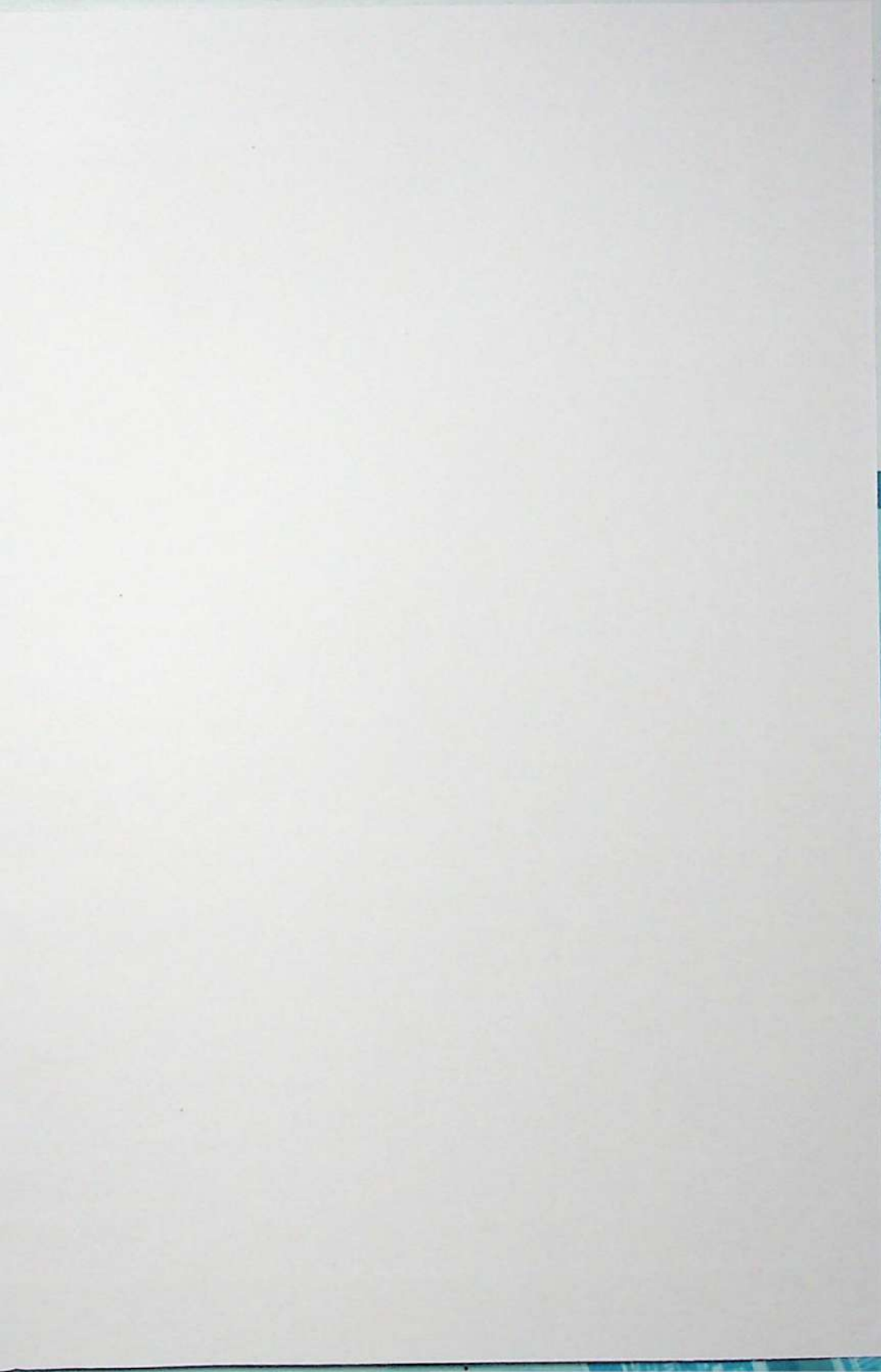
«Donishmand ziyosi» nashriyoti.

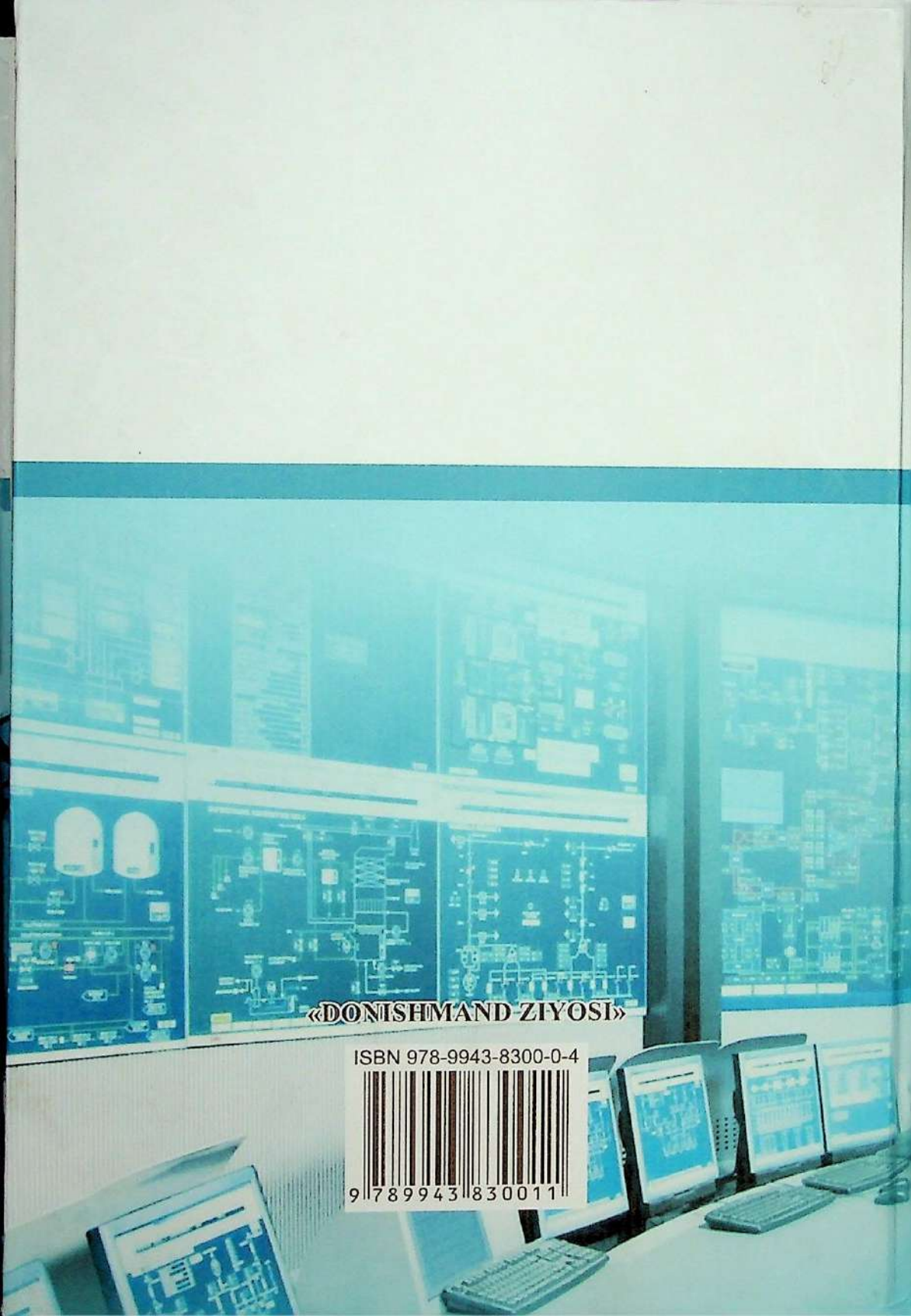
100011, Toshkent shahri, Navoiy ko'chasi, 30-uy.

«Donishmand ziyosi» MCHJ bosmaxonasida chop etildi.

100011, Toshkent shahri, Navoiy ko'chasi, 30-uy.

Telefon: (71) 244-40-91, 99-808-19-49, 99-388-38-83.





«DONISHMAND ZIYOSI»

ISBN 978-9943-8300-0-4



9 789943 830011