

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

QARSHI MUHANDISLIK-IQTISODIYOT INSTITUTI

“Axborot texnologiyalari” kafedrası

**“AXBOROT TEXNOLOGIYALARI VA JARAYONLARNI
MATEMATIK MODELLASHTIRISH”**

fanidan

MA'RUZALAR MATNI



Qarshi – 2022

“Axborot texnologiyalari va jarayonlarni matematik modellashtirish” fanidan ma’ruzalar matni 5310600-Transport vositalari muhandisligi, 60112400- Professional ta’lim (QXM) bakalavriat ta’lim yo’nalishi va barcha texnika bakalavriat ta’lim yo’nalishi talabalari uchun mo’ljallangan.

Tuzuvchi:	G.B.Jamalova “Axborot texnologiyalari” kafedrasida o’qituvchisi
------------------	---

Taqrizchilar:

PhD B.Suropov QarMII “Axborot texnologiyalari” kafedrasida mudiri.

A.R.Mallayev QarMII “Texnologik jarayonlarni avtomatlashtirish va boshqarish” kafedrasida dotsenti.

Ma’ruzalar matni «Axborot texnologiyalari» kafedrasining 20__ yil “__” __dagi __-sonli, “Muhandislik texnikasi” fakulteti o’quv-uslubiy komissiyasining 20__ yil “__” __dagi __-sonli, institut o’quv-uslubiy Kengashining 20__ yil “__” __dagi __-sonli yig’ilishlarida ko’rib chiqilib, chop etishga tavsiya etildi.

SO'Z BOSHI

Ushbu ma'ruza matnlari barcha texnika ta'lim yo'nalishlari talabalari uchun mo'ljallangan bo'lib, unda texnik masalalarni qo'yish va ularni kompyuter yordamida yechish, "Axborot texnologiyalari va jarayonlarni matematik modellashtirish"ning ijtimoiy sohalar, iqtisodiyot va boshqaruvdagi roli, AKTni rivojlantirish dasturi va undagi vazifalar, axborot tizimlari, axborot tizimlarining rivojlanish bosqichlari, kompyuter tizimlari va ularning arxitekturasi, shu jumladan zamonaviy kompyuter tizimlarining hozirgi holati, axborot jarayonlarining dasturiy ta'minoti, dasturiy ta'minotning turkumlanishi, elektron hujjatlar, elektron hujjatlarni yaratishga mo'ljallangan amaliy dasturlar kabi asosiy masalalar ko'rib chiqiladi. Talabalar bu kursni tugallagandan so'ng, ishlab chiqarish sohalardagi dasturiy-texnik vositalar, Windows, Word, Excel, Access, Power Point va shunga o'xshash dasturlarda ishlay olish ko'nikmalarini hosil qiladilar. Talabalarning olgan bilim va ko'nikmalari keyingi, yuqori kurslarda o'tiladigan boshqa fanlarni o'rganish uchun asos bo'lib xizmat qiladi.

Talabalar ushbu kursdan to'laonli saboq olishlari va bu fanni yaxshi o'zlashtirishlari uchun ular maktab dasturi miqiyosida kompyuter haqidagi bilimlarga ega bo'lishlari lozim. Ya'ni, dasturlarni kompyuterga kiritish va ularni sozlash, kompyuterning qo'shimcha qurilmalarining ishini tushunish kabi bilimlarni o'zlashtirib olishlari lozim.

Ushbu fanni o'qitishda ilg'or pedagogik texnologiyalar va uning xilma-xil usullari keng miqyosda qo'llaniladi. Talabalar ma'ruza darslarida olgan bilim va ko'nikmalarini amaliy mashg'ulotlar va laboratoriya darslarida qo'llashni o'rganadilar.

Maruza №1:

Mavzu: Axborot texnologiyalari va jarayonlarni matematik modellashtirish faniga kirish

REJA:

1. “Axborot texnologiyalari va jarayonlarni matematik modellashtirish” faniga kirish;
2. “Axborot texnologiyalari va jarayonlarni matematik modellashtirish ” fanining mohiyati, maqsadi va vazifalari;
3. “Axborot texnologiyalari va jarayonlarni matematik modellashtirish ” fanining hozirgi zamondagi o‘rni va rivojlanish istiqbollari.

Tayanch so‘zlar: Komponent, elektron to‘lov, algoritmik vositalar, dasturiy vositalar, texnik vositalar, kommunikatsiya, globallashtirish, elektron tijorat, elektron raqamli imzo, axborot tizimi, axborotlashtirish, konvergentsiya, elektron hujjat

Axborot texnologiyalari fanini o‘rganishdan maqsad quyidagilar haqida mukammal bilimga ega bo‘lishdan iborat:

- informatika va axborot texnologiyalari fanining taraqqiyoti;
- axborotlar va ularning tasnifi;
- axborotlarning kompyuterda tasvirlanishi;
- kompyuter avlodlari;
- kompyuter qurilmalari va ularning ishlash prinsiplari;
- dasturlash haqida ma’lumotlar.

Fanning asosiy vazifasi – milliy iqtisodiyotda iqtisodiy va texnologik jarayonlarni tashkil qilish va boshqarish masalalarini hal qilishda informatika fanining yutuqlaridan, ya’ni matematik modellashtirish, algoritmlar tuzish, axborot ta’minotini tayyorlash, dasturlardan foydalanish va masalalarni kompyuterda yechishni o‘rganishdan iborat.

Informatikaning asosiy uchta yo‘nalishi mavjud.

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

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

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

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

Texnologiyalar. "Texnologiya" grekcha so'z bo'lib mohirlik, ustalik, biror ishni uddalay olishni anglatadi. Bu ma'lum bir jarayonga nisbatan qo'llanilgan. Jarayon deganda esa maqsadga erishishga yo'naltirilgan xatti-harakatlar majmui tushunilgan. Ushbu jarayon inson tomonidan tanlangan strategiya bilan belgilanadi va turli xildagi vositalar, usullar yordamida amalga oshiriladi.

Umumiy hollarda texnologiya deganda, mahsulotni ishlab chiqarish jarayonida amalga oshiriladigan xomashyo, material yoki yarim tayyor mahsulot shakli, xususiyati, holatining o'zgarishi, uni qayta ishlash, tayyorlash usullarining majmui tushuniladi. Bu biror bir ishni yuqori darajada uddalash deganidir.

Axborot texnologiyalari va jarayonlarni matematik modellashtirish to'g'risida gap ketganda, material sifati ham, mahsulot sifatida ham axborot ishtirok etadi. Biroq bu ob'ekt, jarayon yoki hodisa to'g'risidagi sifat jihatidan yangi ma'lumot bo'ladi. Texnologiya xodimning axborot bilan ishlash usuli va uslubi hamda texnik vositalar orqali namoyon bo'ladi.

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

Moddiy ishlab chiqarish texnologiyasining maqsadi – inson yoki tizimning ehtiyojini qondiruvchi mahsulot ishlab chiqarish sanaladi.

Axborot texnologiyasining maqsadi esa – axborot ishlab chiqarish bo'lib, uni tahlil etish va uning asosida biror bir harakatga qo'l urish uchun tegishli qaror qabul qilish.

Axborotlarni yig'ish, uzatish, to'plash, saqlash, taqdim etish va foydalanish uslublari va usullari tizimi axborot texnologiyasi deb yuritiladi.

Axborot texnologiyasi avtomatlashgan va an'anaviy (qog'oz) ko'rinishga amalga oshiriladi. Avtomatlashtirish hajmi va texnik vositalardan foydalanish turi aniq bir texnologiyaning mohiyatiga bog'liq.

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

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

1-jadval.

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

1. Axborot texnologiyasining asosiy tavsifi.

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

Zamonaviy axborot texnologiyasi (ZAT) (komp'yuter axborot texnologiyasi) – shaxsiy komp'yuter va telekommunikatsiya vositalaridan foydalanuvchi uchun qulay "interfeys"li axborot texnologiyasidir. Ma'lum bir turdagi komp'yuter uchun mo'ljallangan bir yoki bir necha o'zaro bog'liq dasturiy mahsulotlar zamonaviy Axborot texnologiyalari va jarayonlarni matematik modellashtirish ning vositasi sanaladi. ZAT ning asosiy tavsifi 2-jadvalda keltirilgan.

2-jadval

Zamonaviy Axborot texnologiyalari va jarayonlarni matematik modellashtirish ning asosiy tavsifi

Uslubiyot	Asosiy belgisi	Natija
Axborotni qayta ishlashning asosiy yangi vositasi	Boshqaruv texnologiyasiga “joylashish”	Kommunikatsiyaning yangi texnologiyasi
YAXlit texnologik tizimlar	Mutaxassislar va menejerlar vazifasining integratsiyalashuvi	Axborotni qayta ishlash bo'yicha yangi texnologiya
Maqsadga qaratilgan holda axborotni yaratish, uzatish, saqlash va aks ettirish	Ijtimoiy muhit qonunchiligini hisobga olish	Boshqaruv qarorlar qabul qilishning yangi texnologiyasi

3. Axborot texnologiyasining asosiy elementlari.

ZAT ning asosiy elementlari quyidagicha:

- ma'lum bir vaqt davomida axborotni kiritish va qayta o'zgartirish;
- tasvirni kiritish va unga ishlov berish;
- signal axboroti paydo bo'lgan erda uni qayta ishlash;
- og'zaki axborotni qayta ishlash;
- foydalanuvchining SHK bilan faol muloqoti;
- turli axborot tizimlarida mashinali modellashtirish;
- axborot almashuvining tarmoq texnologiyasi (dialog yuritish, video va telekommunikatsiya, elektron pocha, vidioteka va x.z.);
- taqsimlangan tarmoq tizimlarida ma'lumotlarni mul'tiprotsessor asosida qayta ishlash;
- mahalliy, mintaqaviy va xalqaro tarmoqlar bo'yicha axborotni tezkor tarqatish.

Axborot texnologiyasining bazaviy texnologiyasi bo'lib quyidagilar hisoblanadi:

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

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

4. Axborot texnologiyalari va jarayonlarni matematik modellashtirish taraqqiy etishining asosiy bosqichlari va rivojlanish tendentsiyasi.

XIX asrning ikkinchi yarmigachi “qo'l” axborot texnologiyasi rivojlangan bo'lib, uning asosini pero, siyohdon va buxgalteriya daftari tashkil etgan. Kommunikatsiya (aloqa) paket (rasiy xujjatlar solingan konvert) yuborish orqali amalga oshirilar edi.

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

XX asrning 40 - 60–yillarida “elektr” texnologiyasi paydo bo'lib, u echib almashtiriladigan elementlarga ega elektr yozuv mashinkalari, oddiy qog'ozdan foydalanuvchi nusxa ko'chirish mashinasi, portativ diktafonlardan iborat edi.

60-yillarning ikkinchi yarmidan esa “elektron (yoki komp'yuter)” texnologiyasi yuzaga kela boshladi va axborotni shaklini emas, mazmunini o'zgartirishga urg'u berila boshlandi.

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

Xorijiy mutaxassislar Axborot texnologiyalari va jarayonlarni matematik modellashtirish rivojlanishiing beshta asosiy tendentsiyasini ajratib ko'rsatadi:

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

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

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

4. Globalashtirish. Tashkilot yo'ldosh aloqa va Internet tarmog'ida foydalanib Axborot texnologiyalari va jarayonlarni matematik modellash tirish yordamida hohlagan joyda va hohlagan paytda ish olib borish mumkin.

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

O'zbekiston Respublikasining axborotlashtirish dasturi.

O'zbekistonda axborot texnologiyasi rivojlanishini tezlashtirish va zamonaviy darajaga ko'tarish uchun Respublika Vazirlar Mahkamasi tomonidan bir qator qarorlar qabul qilingan. 1994 yilda Vazirlar Mahkamasi O'zbekiston Respublikasini axborotlashtirish konsepsiyasini qabul qilgan. Konsepsiya asosida "O'zbekiston Respublikasini axborotlashtirish Dasturi" ishlab chiqilgan.

Dastur uch maqsadni o'z ichiga oladi:

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

Milliy axborot hisoblash tarmog'i davlat aloqa tizimi negizida ishlaydigan va yagona qoidalarga rioya qilish asosida qurilgan davlat va idoraviy xususiyatga ega bo'lgan axborot hisoblash tarmoqlarini mujassamlashgan ochiq tizimidir.

EHM ni matematik va dasturiy ta'minlash axborotlashtirish masalalarining matematik modeli, hisoblash algoritmi va echim olish dasturini yaratishdan iboratdir.

Dasturning shaxsiy kompyuter qismi informatika industriyasini yaratish va rivojlantirishdan iborat.

1. Respublikada axborot-kommunikatsiya texnologiyalari sohasidagi Qonunlar.

- Axborotlashtirish to'g'risidagi qonuni (2003y.)
- Elektron raqamli imzo to'g'risidagi qonun (2003y.)
- Elektron hujjat aylanishi to'g'risidagi qonun (2004y.)
- Elektron tijorat to'g'risidagi qonun (2004y.)
- Elektron to'lovlar to'g'risidagi qonun (2005y.)

1.O'zbekiston Respublikasining axborotlashtirish to'g'risidagi qonuni 2003 yilda qabul qilingan. Ushbu qonun 23 ta moddadan iborat bo'lib, uning maqsadi axborotlashtirish, axborot resurslari va axborot tizimlaridan foydalanish sohasidagi munosabatlarni tartibga solishdan iborat.

Ushbu Qonunda quyidagi asosiy tushunchalar qo'llaniladi:

axborotlashtirish - yuridik va jismoniy shaxslarning axborotga bo'lgan ehtiyojlarini qondirish uchun axborot resurslari, Axborot texnologiyalari va jarayonlarni matematik modellashtirish hamda axborot tizimlaridan foydalangan holda sharoit yaratishning tashkiliy ijtimoiy-iqtisodiy va ilmiy-texnikaviy jarayoni;

axborot resursi – axborot tizimi tarkibidagi elektron shakldagi axborot, ma'lumotlar banki, ma'lumotlar bazasi;

axborot resurslarining yoki axborot tizimlarining mulkdori – axborotresurslarigayoki axborot tizimlariga egalik qiluvchi, ulardan foydalanuvchi va ularni tasarruf etuvchi yuridik yoki jismoniy shaxs;

axborot resurslarining yoki axborot tizimlarining egasi – qonun bilan yoki axborot resurslarining, axborot tizimlarining mulkdori tomonidan belgilangan huquqlar doirasida axborot resurslariga yohud axborot tizimlariga egalik qiluvchi, ulardan foydalanuvchi va ularni tasarruf etuvchi yuridik yoki jismoniy shaxs;

axborot texnologiyasi – axborotni to'plash, saqlash, izlash, unga ishlov berish va uni tarqatish uchun foydalaniladigan jami uslublar, qurilmalar, usullar va jarayonlar;

axborot tizimi – axborotni to'plash, saqlash, izlash, unga ishlov berish hamda undan foydalanish imkonini beradigan, tashkiliy jihatdan tartibga solingan jami axborot resurslari, Axborot texnologiyalari va jarayonlarni matematik modellashtirish va aloqa vositalari.

Axborotlashtirish sohasidagi davlat siyosati axborot resurslari, Axborot texnologiyalari va jarayonlarni matematik modellashtirish va axborot tizimlarini rivojlantirish hamda takomillashtirishning zamonaviy jahon tamoyillarini hisobga olgan holda milliy axborot tizimini yaratishga qaratilgan.

Axborotlashtirish sohasidagi davlat siyosatining asosiy yo'nalishlari quyidagilardan iborat:

har kimning axborotni erkin olish va tarqatishga doir konstitutsiyaviy huquqlarini amalga oshirish, axborot resurslaridan erkin foydalanilishini ta'minlash;

davlat organlarining axborot tizimlari, tarmoq va hududiy axborot tizimlari, shuningdek yuridik hamda jismoniy shaxslarning axborot tizimlari asosida O'zbekiston Respublikasining yagona axborot makonini yaratish;

xalqaro axborot tarmoqlari va Internet jahon axborot tarmog'idan erkin foydalanish uchun sharoit yaratish;

davlat axborot resurslarini shakllantirish, axborot tizimlarini yaratish hamda rivojlantirish, ularning bir-biriga mosligini va o'zaro aloqada ishlashini ta'minlash;

Axborot texnologiyalari va jarayonlarni matematik modellashtirish ning

zamonaviy vositalari ishlab chiqarilishini tashkiletish;

axborot resurslari, xizmatlari va Axborot texnologiyalari va jarayonlarni matematik modellashtirish bozorini shakllantirishga ko'maklashish;

dasturiy mahsulotlar ishlab chiqarish rivojlantirilishini rag'batlantirish;

tadbirkorlikni qo'llab-quvvatlash va rag'batlantirish, investitsiyalarni jalbetish uchun qulay sharoit yaratish;

kadrlar tayyorlash va ularning malakasini oshirish, ilmiy tadqiqotlarni rag'batlantirish.

2.O'zbekiston Respublikasining elektron raqamli imzo to'g'risidagi qonuni 2003 yilda qabul qilingan. Ushbu qonun 22 ta moddadan iborat bo'lib, uning maqsadi elektron raqamli imzodan foydalanish sohasidagi munosabatlarni tartibga solishdan iborat.

Ushbu Qonunda quyidagi asosiy tushunchalar qo'llaniladi:

elektron raqamli imzo - elektron hujjatdagi mazkur elektron hujjat axborotini elektron raqamli imzoning yopiq kalitidan foydalangan holda maxsus o'zgartirish natijasida hosil qilingan hamda elektron raqamli imzoning ochiq kaliti yordamida elektron hujjatdagi axborotda xatolik yo'qligini aniqlash va elektron raqamli imzo yopiq kalitining egasini identifikatsiya qilish imkoniyatini beradigan imzo;

elektron raqamli imzoning yopiq kaliti - elektron raqamli imzo vositalaridan foydalangan holda xosil qilingan, faqat imzo qo'yuvchi shaxsning o'ziga ma'lum bo'lgan va elektron hujjatda elektron raqamli imzoni yaratish uchun mo'ljallangan belgilar ketma-ketligi;

elektron raqamli imzoning ochiq kaliti - elektron raqamli imzo vositalaridan foydalangan holda hosil qilingan, elektron raqamli imzoning yopiq kalitiga mos keluvchi, axborot tizimining har qanday foydalanuvchisi foydalana oladigan va elektron hujjatdagi elektron raqamli imzoning haqiqiylikini tasdiqlash uchun mo'ljallangan belgilar ketma-ketligi;

elektron raqamli imzoning haqiqiylikini tasdiqlash - elektron raqamli imzoning elektron raqamli imzo yopiq kalitining egasiga tegishliligi va elektron hujjatdagi axborotda xatolik yo'qligi tekshirilgandagi ijobiy natija;

elektron hujjat - elektron shaklda qayd etilgan, elektron raqamli imzo bilan tasdiqlangan hamda elektron hujjatning uni identifikatsiya qilish imkonini beradigan boshqa rekvizitlariga ega bo'lgan axborot.

3.O'zbekiston Respublikasining elektron hujjat aylanishi to'g'risidagi qonuni 2004 yilda qabul qilingan. Ushbu qonun 19 ta moddadan iborat bo'lib, uning maqsadi elektron hujjat aylanishi sohasidagi munosabatlarni tartibga

solishdan iborat.

Elektron hujjat aylanishi sohasidagi davlat siyosati elektron hujjat aylanishi keng qo'llanilishini ta'minlashga, elektron hujjat aylanishi ishtirokchilarining huquqlari va qonuniy manfaatlarini himoya qilishga, elektron hujjatdan foydalanish standartlari, normalari va qoidalarini ishlab chiqishga qaratilgan.

Ushbu qonunda quyidagi asosiy tushunchalar qo'llaniladi:

Elektron hujjat - Elektron shaklda qayd etilgan, elektron raqamli imzo bilan tasdiqlangan va elektron hujjatning uni identifikatsiya qilish imkoniyatini beradigan boshqa rekvizitlariga ega bo'lgan axborot elektron hujjatdir.

Elektron hujjat texnika vositalaridan va axborot tizimlari xizmatlaridan hamda Axborot texnologiyalari va jarayonlarni matematik modellashtirish dan foydalanilgan holda yaratiladi, ishlov beriladi va saqlanadi.

Elektron hujjat elektron hujjat aylanishi ishtirokchilarining mazkur hujjatni idrok etish imkoniyatini inobatga olgan holda yaratilishi kerak.

Elektron hujjatning rekvizitlari - Elektron hujjatning majburiy rekvizitlari quyidagilardan iborat:

elektron raqamli imzo;

elektron hujjatni jo'natuvchi yuridik shaxsning nomi yoki elektron hujjatni jo'natuvchi jismoniy shaxsning familiyasi, ismi, otasining ismi;

elektron hujjatni jo'natuvchining pochta va elektron manzili;

hujjat yaratilgan sana.

Qonun hujjatlarida yoki elektron hujjat aylanishi ishtirokchilarining kelishuvi bilan elektron hujjatning boshqa rekvizitlari ham belgilanishi mumkin.

4.O'zbekiston Respublikasining elektron tijorat to'g'risidagi qonuni 2004 yilda qabul qilingan. Ushbu qonun 14 ta moddadan iborat bo'lib, uning maqsadi elektron tijorat sohasidagi munosabatlarni tartibga solishdan iborat.

Ushbu qonunda quyidagi asosiy tushunchalar qo'llaniladi:

Elektron tijorat - Axborot tizimlaridan foydalangan holda amalga oshiriladigan, tovarlarni sotish, ishlarni bajarish va xizmatlar ko'rsatishga doir tadbirkorlik faoliyati elektron tijoratdir.

Elektron tijorat ishtirokchilari - Elektron tijoratni amalga oshiruvchi yuridik va jismoniy shaxslar, shuningdek tegishli tovarlarning (ishlarning, xizmatlarning) xaridorlari bo'lgan yuridik va jismoniy shaxslar elektron tijorat ishtirokchilaridir.

Elektron tijoratda axborot vositachilari ham ishtirok etishlari mumkin. Elektron

hujjat aylanishi bilan bog'liq xizmatlar ko'rsatuvchi yuridik va jismoniy shaxslar axborot vositachilaridir.

5. O'zbekiston Respublikasining elektron to'lovlar to'g'risidagi qonuni 2005 yilda qabul qilingan. Ushbu qonun 23 ta moddadan iborat bo'lib, uning maqsadi elektron to'lovlar sohasidagi munosabatlarni tartibga solishdan iborat.

Ushbu qonunda qo'llaniladigan asosiy tushunchalar:

Elektron to'lov - Texnika vositalaridan, Axborot texnologiyalari va jarayonlarni matematik modellash tirish dan va axborot tizimlari xizmatlaridan foydalangan holda elektron to'lov hujjatlari vositasida naqd pulsiz hisob-kitoblarni amalga oshirish elektron to'lovdir. Elektron to'lov to'lov tizimining sub'ektlari o'rtasida tuzilgan shartnomalar asosida to'lov tizimining belgilangan qoidalariga muvofiq amalga oshiriladi.

To'lov tizimi va uning turlari - Elektron to'lovlar amalga oshirilayotganda to'lov tizimining sub'ektlari o'rtasida yuzaga keladigan munosabatlar majmui to'lov tizimidir.

To'lov tizimining turlari:

Banklararo to'lov tizimi;

Bankning ichki to'lov tizimi;

CHakana to'lovlar tizimlaridan iborat.

2. O'zbekiston Respublikasi Prezidentining farmonlari.

1. Telekommunikatsiyalar sohasida boshqaruvni takomillashtirishga doir chora-tadbirlar to'g'risida (2000 y.)

2. Kompyuterlashtirishni yanada rivojlantirish va axborot-kommunikatsiya texnologiyalarini joriy etish to'g'risida (2002 y.)

3. Axborot-kommunikatsiya texnologiyalarini yanada rivojlantirishga oid qo'shimcha chora-tadbirlar to'g'risida (2005 y.)

4. O'zbekiston Respublikasining jamoat ta'lim axborot tarmog'ini tashkil etish to'g'risida (2005 y.)

5. Respublika aholisini axborot-kutubxona bilan ta'minlashni tashkil etish to'g'risida (2006 y.)

4. O'zbekiston Respublikasi Prezidenti va O'zbekiston Respublikasi Vazirlar mahkamasining qarorlari va sohaga oid me'yoriy hujjatlari

1. Kompyuterlashtirishni yanada rivojlantirish va axborot-kommunikatsiya texnologiyalarini joriy etish chora-tadbirlari to'g'risida (2002y.)

2.Axborotlashtirish sohasida normativ-huquqiy bazani takomillashtirish to‘g‘risida (2005 y.)

3.Davlat va xo‘jalik boshqaruvi, Mahalliy davlat hokimiyati organlarining axborot-kommunikatsiya texnologiyalaridan foydalangan holda yuridik va jismoniy shaxslar bilan o‘zaro aloqadorligini yanada takomillashtirish chora-tadbirlari to‘g‘risida (2007 y.)

4.Internet tarmog‘ida O‘zbekiston Respublikasining Hukumat portalini yanada rivojlantirish chora-tadbirlari to‘g‘risida (2007y.)

5.Internet tarmog‘ida O‘zbekiston Respublikasining Hukumat portaliga axborotlarni taqdim etish va joylashtirish tartibi to‘g‘risida (2009y.)

6.Zamonaviy axborot-kommunikatsiya texnologiyalarini yanada joriy etish va rivojlantirish chora-tadbirlari to‘g‘risida (2012 y.)

7.Interaktiv davlat xizmatlari ko‘rsatishni hisobga olgan holda Internet tarmog‘ida O‘zbekiston Respublikasining hukumat portali faoliyatini yanada takomillashtirish chora-tadbirlari to‘g‘risida (2012 y.)

8.O‘zbekiston Respublikasi Oliy va o‘rta maxsus ta’lim vazirligi huzurida ta’lim muassasalarida elektron ta’limni joriy etish markazini tashkil etish to‘g‘risida (2012 y.)

9.O‘zbekiston Respublikasi Aloqa, axborotlashtirish va telekommunikatsiya texnologiyalari davlat qo‘mitasi huzuridagi “Elektron xukumat” tizimini rivojlantirish markazi hamda axborot xavfsizligini ta’minlash markazi faoliyatini tashkil etish chora-tadbirlari to‘g‘risida (2013 y.)

8.Mamlakatimizning dasturiy ta’minot vositalari ishlab chiqaruvchilarini rag‘batlantirishni yanada kuchaytirish chora-tadbirlari to‘g‘risida (2013 y.)

Nazorat savollari

1. Axborot texnologiyalari va jarayonlarni matematik modellashtirish fanining maqsadi nimalardan iborat ?

2. Axborot texnologiyalari va jarayonlarni matematik modellashtirish fanining vazifalari nimalardan iborat ?

3. Axborot texnologiyalari va jarayonlarni matematik modellashtirish qanday belgilar asosida tavsiflanadi?

4. Axborot mahsulotlari tarkibiga nimalar kiradi?

5. Axborot texnologiyasi deganda nimani tushuniladi?

6. Avtomatlashtirish qachon zarur bo‘ladi?

7. Avtomatlashtirilgan Axborot texnologiyalari va jarayonlarni matematik modellashtirish nima uchun xizmat qiladi?

8. Axborot texnologiyalari va jarayonlarni matematik modellashtirish qanday belgilar asosida tavsiflanadi?

9. Zamonaviy Axborot texnologiyalari va jarayonlarni matematik modellashtirish ning asosiy elementlarini keltiring.

10. Axborot texnologiyalari va jarayonlarni matematik modellashtirish ning taraqqiy etish bosqichlarini aytib bering.

11. Avtomatlashtirilgan Axborot texnologiyalari va jarayonlarni matematik modellashtirish ning rivojlanib borish tendentsiyalari nimalar bilan izohlanadi?

Ma'ruza №2

2- Ma'ruza: Intellektual boshqaruv tizimlari maqsadi, funktsiyalari va uslubi. Ekspert tizimlari.

Reja:

1. Ta'lim jarayonida intellektual tizimlarni qo'llash
2. Ekspert tizimlari.

Ta'lim jarayonida talabaning kasbiy hamda individual rivojlanishi jarayonini tashkil etishga nisbatan yangicha yondashuv sifatida ixtisoslik fanlarini intellektual tizimlar asosida o'qitishning maqsadi, tarkibi va qo'llash texnologiyalarini ishlab chiqish hozirgi kunning dolzarb muamolaridan biri hisoblanadi.

O'qitishning intellektual tizimi sun'iy intellekt metodlari va vositalarining avtomatlashtirilgan o'qitish sohasida qo'llanishining amaliy natijasi bo'lib, ta'lim tizimlarining yangi avlodi hisoblanadi. Ta'lim jarayonida talabaning bilim ko'nikma va malakalarini shakllantirishning yuqori ko'rsatkichlariga erishish uchun, ixtisoslik fanlari o'qituvchisi uch asosiy tipdagi bilimlardan foydalanadi: o'qitilayotgan ixtisoslik fani bo'yicha bilimlar, o'qitish metodlari to'g'risidagi bilimlar va ta'lim oluvchi haqidagi bilimlar. O'qitishning an'anaviy avtomatlashtirilgan tizimlarida bu bilimlarning ko'pgina qismlari tanlangan o'qitish metodikasiga mos ravishda o'quv fanining alohida bo'limlariga qat'iy ravishda kiritilgan. O'qitishning intellektual tizimida zarur bo'lgan bilimlar ajratib olinadi va sun'iy intellektning turli metodlari va texnologiyalaridan foydalangan holda ko'rsatiladi. O'qitishning intellektual tizimida o'quv materialini taqdim etishda, bu bilimlardan foydalanib va talabaning psixofiziologik va intellektual imkoniyatlaridan kelib chiqqan holda, eng samarali o'qitish metodlari, usullari va sur'atlarini, fanning mazmuni, hajmi va topshiriqlarning murakkablik darajasini aniqlash va tartibga solish mumkin bo'ladi.

Ixtisoslik fanlarini o'qitishning intellektual tizimi bilimlar bazasi, boshqarish tizimchasi, o'qitish tizimchasi va muloqot tizimchasidan iborat. Ixtisoslik fanlarini o'qitishning intellektual tizimining asosi bo'lib bilimlar bazasi hisoblanadi. Har bir ixtisoslik fani bo'yicha bilimlar bazasi o'z navbatida, o'zaro bog'langan bir necha ma'lumotlar bazasidan iborat bo'ladi va har biri o'qitilayotgan fanning ma'lum bir qismlarini tashkil etadi: «Atamalar va asosiy tushunchalar», «Nazariya», «Amaliy-laboratoriya ishlari», «Mustaqil ishlar».

O'qitish metodlari haqidagi bilimlar «Topshiriqlar» ma'lumotlar bazasida yig'ilib boradi, bu bazada o'quv fanining barcha bo'limlari bo'yicha didaktik materiallar tasniflangan va tartibli tarzda, o'qitish metodlariga mos ravishda namoyon bo'ladi. Ta'lim oluvchi haqidagi bilimlar «O'qitish natijalari» ma'lumotlar bazasida yig'iladi. Bunda har bir talabaning individual kartasi tuzilib, unda o'qitilayotgan fan bo'yicha talaba ega bo'lgan bilimlar darajasi va sifati haqida, uning psixofiziologik rivojlanish xususiyatlari haqidagi ma'lumotlar aks ettiriladi.

Bilimlar bazasini boshqarish tizimchasi yangi bilimlarni kiritish va to'plash uchun mo'ljallangan. Birinchi bosqich tizimni o'rganishdan iborat bo'ladi, ixtisoslik fani bo'yicha barcha bilimlar yangi hisoblanadi va bu bilimlarni yig'ish tizim va ekspert o'rtasidagi muloqot (dialog) jarayonida amalga oshiriladi. Keyingi bosqichlarda yangi bilimlarni izlash avtomatik tarzda amalga oshiriladi: barcha qabul qilinayotgan axborotlar mavjud bo'lgan axborotlar bilan solishtiriladi, mavjud bo'lgan bilimlarga tayangan holda yangi axborotlar saralanadi, tahlil qilinadi va ularning tasnifi bir yoki bir necha variantda taqdim etiladi. Masalan, biror buyumga ishlov berishning yangi usuli topilganda, uni bajarish yo'llari, kerak bo'ladigan asbob-uskunalar va moslamalar, foydalaniladigan materiallar, qo'llaniladigan sohalar haqidagi ma'lumotlar o'rganish uchun taqdim etiladi.

O'qitish tizimchasi aniq maqsadga yo'naltirilgan ta'lim jarayonini tashkil etish va qo'llab-quvvatlashga mo'ljallangan hamda bu tizimcha uch moduldan iborat bo'ladi: o'qitishni boshqarish, topshiriqlarni shakllantirish, natijalarni tahlil qilish. O'qitishning har bir bosqichida, talabaning individual kartasini tahlil qilish natijalariga asoslangan holda o'qitishning maqsadi aniqlanadi, o'qitish vazifalari tanlanadi yoki ishlab chiqiladi. Agar talaba mustaqil ravishda ishlaydigan bo'lsa, uning harakatlari kuzatilib, xatoliklari, duch kelgan qiyinchiliklari qayd qilib boriladi, qachon o'quv jarayoniga aralashish, yordam ko'rsatish, talabaning individual kartasiga o'zgartirishlar kiritish lozimligi aniqlanadi. Yordam talab qilinganda, uni ko'rsatishning optimal shakllari tanlanadi (xatoni ko'rsatish, talabaning o'zi xatolarini tuzatishi yoki bilmaganlarini to'ldirishi uchun informatsion materiallar taklif etish va h.k.). Barcha qarorlar talabaning individual xususiyatlaridan kelib chiqqan holda qabul qilinishi bois, intellektual tizim ta'lim jarayonining talaba uchun optimal tarzda kechishini ta'minlaydi.

Muloqot tizimchasi ilm oluvchi bilan individual tarzda va tarmoq rejimida ishlashda, masofadan turib o'qitish jarayonida o'zaro muloqotni ta'minlash uchun xizmat qiladi. Bu tizim talabaning shaxsini aniqlashga, uning o'qitish tizimchasidan bilimlar bazasidan, ma'lumotlar va o'quv-axborot materiallaridan foydalana olishini ta'minlashga qaratilgan. Talaba va tizim o'rtasidagi muloqot tushunish uchun qulay bo'lgan shakllarda va sodda tilda amalga oshirilishi lozim.

Ixtisoslik fanlarini o'qitishning intellektual tizimini amalga oshirish jarayonining umumiy ko'rinishini quyidagi sxema tarzida ifodalash mumkin: foydalanuvchining shaxsi tasdiqlangandan keyin, uning tizimga kirish huquqi aniqlanadi, o'qitishni boshqarish tizimchasining moduli talabani individual kartasining holatini tahlil qiladi, navbatdagi o'qitishning maqsadi, metodikasi va vazifalarini belgilaydi. Topshiriqlarini shakllantirish moduli tanlangan metodika va bilimlar bazasi ma'lumotlariga asoslanib, o'qitish vazifasini ishlab chiqadi. Talaba qo'yilgan vazifani bajaradi (nazariy materialni o'rganish, amaliy ishni bajarish, test, savollarga javob yozish va h.k.), bunda tizim bilan muloqot ta'minlanib turiladi, ya'ni talabani ishi nazorat qilinib, kerak bo'lsa, yordam ko'rsatib turiladi. Natijalarni tahlil qilish moduli yo'l qo'yilgan xatoliklarni, sodir bo'lgan qiyinchiliklarni aniqlaydi va talabani individual kartasiga o'zgartirishlar kiritadi. Shu bilan o'qitish sikli yakunlanadi va tizim keyingi jarayon uchun tayyorlik holatiga o'tadi. Foydalanuvchining o'zi vazifani aniqlashi mumkin (ekranga nazariy materialning kerakli qismini yoki topshiriqni chiqarish, mavjud bo'lgan lug'atlar va manbalarga murojaat etish, individual kartani ko'rib chiqish va h.k.).

Ixtisoslik fanlarini o'qitishning intellektual tizimi talabani individual xususiyatlarini hisobga olgan holda, yopiq tizimda, avtomatlashtirilgan boshqaruv vositasida, mustaqil, individual, jamoaviy va masofaviy shakllarda shaxsning kamolotiga yo'naltirilgan, aniq maqsadlarga qaratilgan ta'lim jarayonini amalga oshirish imkonini beradi.

Ekspert tizimlari bugun bir necha mingdagi har xil dasturiy majmuallarni birlashtiradilar, ularni har xil mezonlar bo'yicha tasniflash mumkin (12.1-rasm).

Har bir mezon bo'yicha tasnifni ko'rib chiqamiz. Echilayotgan vazifa bo'yicha tasnif.

Ma'lumotlarning interpretatsiyasi. Bu ekspert tizimlar uchun an'anaviy vazifalarning biridir. Interpretatsiya ostida ma'lumotlarning ma'nosini aniqlash tushuniladi, uning natijalari kelishilgan va aniq bo'lishi kerak. Odatda ma'lumotlarning ko'p variantli tahlili ko'zda tutiladi.



2.1-rasm. Ekspert tizimlarining tasnifi Har bir mezon bo'yicha tasnifni ko'rib chiqamiz. Yechilayotgan vazifa bo'yicha tasnif.

Ma'lumotlarning interpretatsiyasi. Bu ekspert tizimlar uchun an'anaviy vazifalarning biridir. Interpretatsiya ostida ma'lumotlarning ma'nosini aniqlash tushuniladi, uning natijalari kelishilgan va aniq bo'lishi kerak. Odatda ma'lumotlarning ko'p variantli tahlili ko'zda tutiladi.

Diagnostika (tashxis). Diagnostika ostida ba'zi bir tizimdagi nosozlikni topish tushuniladi. Buzoqlik - bu me'yordan chetga chiqishdir. Bunday talqin ham uskunalar va texnik vositalarning nosozligi va ham tirik organizmlarning kasalliklari va ehtimol bo'lgan tabiiy anomaliyalarni yagona nazariy nuqtai nazardan ko'rib chiqishga imkon beradi. Diagnostikalanayotgan tizimning vazifaviy tuzilishi (anatomiyasi)ni tushunish zaruriyati muhim o'ziga xoslik bo'ladi.

Monitoring. Monitoringning asosiy vazifasi vaqtning haqiqiy qo'llashida ma'lumotlarni uzluksiz interpretatsiyalangan va u yoki bu o'lchamlarni yo'l qo'yiladigan chegaradan tashqariga chiqishi haqida xabar berishdir. Asosiy muammolar tashvishli vaziyatni "o'tkazib yuborish" va "yolg'ondan" ishlab ketishning inversli vazifasidir. Bu muammolarni murakkabligi tashvishli vaziyatlar alomatlar (simptom)larini emirilib ketishi va vaqt kontekstini hisobga olishning zaruriyatidan iborat.

Loyihalashtirish. Loyihalashtirish oldindan belgilangan xususiyatlarga ega "ob'ektlar"ni tayyorlashga ixtisoslikni tayyorlashdan iboratdir. Ixtisoslik ostida zaruriy hujjatlar-rasm, izohli yozuv va x.k.ning butun majmuasi tushuniladi. Bu erda asosiy muammolar - ob'ekt "haqidagi" bilimlarning aniq tarkibiy bayonni olish va "iz"ning muammosidir. Samarali loyihalashtirish va yanada ko'proq darajada qayta loyihalashtirishni tashkil qilish uchun nafaqat loyiha qarorlarini o'zini balki

ularni qabul qilish sabablarini shakllantirish zarur. Shunday qilib loyihalashtirish vazifalarida tegishli ET doirasida bajariladigan ikkita asosiy jarayonlar: xulosalarni chiqarish jarayoni va izohlash jarayoni yaqindan bog'lanadi.

Bashoratlash. Loyihalashtirilayotgan tizimlar berilgan vaziyatlardan ehtimol bo'lgan oqibatlarni mantiqan keltirib chiqaradi. Loyihalashtirilayotgan tizimda odatda parametrik dinamik modeldan foydalaniladi, unda parametrlarning ma'nosi berilgan vaziyat ostiga to'g'rilanadi. Ushbu modeldan chiqarilayotgan natijalar ehtimollik baholarga ega bashoratlar uchun asosni tashkil qiladi.

Rejalashtirish. Rejalashtirish ostida ba'zi bir vazifalarni bajarishga qodir ob'ektlarga tegishli harakatlar rejalarini topish tuShuniladi. Bunday ETda rejalashtirilayotgan faoliyat natijalarini mantiqiy chiqarish uchun haqiqiy ob'ektlar xulqi modelidan foydalaniladi.

O'qitish. O'qitish tizimi qandaydir fanni EHM yordamida o'rganishda xatolarni diagnostikalaydi va to'g'ri qarorlarni aytib beradi. U gipotetik "o'quvchi" va uning o'ziga xos xatolari haqidagi bilimlarni jamlaydi, keyin ishlashda o'qiyotganlar bilimlaridagi kuchsiz joylarni diagnostikalash va ularni bartaraf qilish uchun tegishli vositalarni topishga qodir. Bundan tashqari ular bilimlarni uzatish maqsadida o'quvchining muvaffaqiyatlariga ko'ra o'quvchi bilan muloqot qilish hujjatini rejalashtiradilar. Umumiy holda, bilimlarga asoslangan barcha tizimlarni tahlil vazifasini echuvchi tizimlar va sintez vazifasini echuvchi tizimlarga bo'lish mumkin. Tahlil vazifalarining sintez vazifalaridan asosiy farqlari quyidagilardan iborat: agar tahlil vazifalarida ko'pchilik echimlarni sanash va tizimga kiritish mumkin bo'lsa, unda sintez vazifalarida echimlarning ko'pchiligi potentsial tarkibiy qismlarining echimlaridan yoki muammo ostida quriladi.

Ma'ruza 3.

Mavzu: Zamonaviy avtomatlashtirilgan loyihalash tizimlari

Reja:

1. Loyihalashning avtomatlashtirilgan tizimi va uning vazifalari.
2. Loyixalashni avtomatlashtirish jarayonining asosiy bosqichlari.
3. Loyihalashning avtomatlashtirilgan tizimi. Ularning turlari va modellari.

1. Loyihalashning avtomatlashtirilgan tizimi va uning vazifalari.

Texnik ob'yektlarni loyihalash – hali mavjud bo'lmagan ob'yekt obrazini yaratish, o'zgartirish va taqdim etish jarayonidir. Ob'yekt 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'iy nazar muhandislik loyihalash

jamoatning ba'zi bir texnik ob'yektlarga bo'lgan ehtiyoqi borligi tufayli boshlanadi. Texnik ob'yektlar qurulish ob'yektlari, ishlab chiqarish mahsulotlari yoki jatayonlar 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 amalgam oshiradigan tizim avtomatlashtirilgan loyihalash tizimini tashkil qiladi. (ingliz tilidan olinganda CAD System — Computer Aided Design System).

Murakkab ob'yektlarni 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 prinsipi 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 jarayinlarni qamrab olgan.

2.Loyihalashni avtomatlashtirish jarayonining asosiy bosqichlari.

Loyihalash bosqichlari- vaqt o'tishi bilan rivojlanib boradigan jarayon kabi loyihalashning katta bo'lagidir. Umumiy holda bosqichlar ilmiy-tadqiqot ishlari, loyiha eskizi yoki tajriba-konstruktorli 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.

Har qanday murakkab tizim kabi ALT(avtomatlashtirilgan loyihalash tizimlari) ham tizimostidan tashkil topadi(1-rasm). Loyihalash va xizmat ko'rsatish tizimosti mavjud. Loyihalash tizimostlari bevosita loyihalash proseduralarini bajaradi. Loyihalash tizimostiga misol qilib mexanik obyektlarni geometrik uch o'chovli modellash, konstruktorlik hujjatlarini tayyorlash, sxemotexnik tahlil qilish tizim ostilari kiradi. .

Xizmat ko'rsatuvchi tizimostilar loyihalanayotgan tizim ostini ishlashini tashkillashtiradi, ularning birlashmasi ko'pincha ALTning tizim muhiti(yoki qobiq) deb ataladi. Xizmat ko'rsatuvchi tizimostlari bo'lib loyiha kattaliklarini boshqarish(PDM — Product Data Management), loyihalash jarayonini bosqarish(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 tizimostlari xizmat qiladi.



2.2-rasm ALT tizim ostlari.

3. Loyihalashning avtomatlashtirilgan tizimi. Ularning turlari va modellari.

САПР klassifikatsiyasi ilovalari, maqsadi, ko'lam, ALT yadrosi baza tizimostining xarakteri kabi bir qator belgilari orqali amalga oshiriladi. Quyida ilovalari bo'yicha ko'proq ishonchli va keng qo'llaniluvchi ALT guruhlar 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 qurulish sohalaridagi ALT.

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

- 1) Funktsional loyihalovchi ALT, boshqacha aytganda ALT-F yoki CAE (Computer Aided Engineering) tizimlari;
- 2) Umumiy mashinasozlikning konstruktorlik SAR — ALT-K, u ko'pincha CAD-tizim deb ataladi;
- 3) Umumiy mashinasozlikning texnologik САПРi — САПР, boshqacha qilib aytganda ishlab chiqarishni texnologik tayyorlashni avtomatlashtirilgan tizimlari yoki CAM (Computer Aided Manufacturing) tizimlari.

Qo'llanilish ko'lam 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 modellashtirish подсистемыsi bazasidagi ALT. Bu ALT konstruktorlash loyihalashning asosiy prosedurasi hisoblangan ilovalarga yo'naltirilgandir. Konstruktorlash jarayoni bu fazoviy forma va ob'ektlarni 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 dasturi-uslubiy komplekslardir. Bularga masalan, ishlab chiqarish jarayonini imitasion modellashtirish, oxirgi elementlar usuli bilan chidamlilikni hisoblash, avtomatlashtirilgan boshqaruv tizimini sintez va analiz 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 подсистемlarni birlashmasidan tashki topgan. Kompleks ALTga mashinasozlikdagi CAE/CAD/CAM-tizimlari yoki ALT BIS kiradi. ALT BIS MMBT va prinsipial, mantiqiy va funksional sxemalar, kristallar topologiyasi, mahsulotni yaroqliligini tekshirish testlari komponentalarini loyihalash tizimostlaridan tashkil topgan. Bunday murakkab tizimlarni boshqarish uchun ixtisoslashtirilgan tizimli muhitlar ishlatiladi.

CAE/CAD/CAM-tizimlarning vazifalari, xarakteristikasi va unga misollar

Mashinasozlikda CAD-tizim vazufasi 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.ñ

CAD-tizimlari “engil” va “ogir” tizimlarga bo'linadi. Engil tizimlar 2D grafikaga yo'naltirilgan bo'lib, arzon va hisoblash resurslariga nisbatan talab kamroq. Ogir 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 asbobni nisbiy harakat trayektoriyasi qurish, ishlov berish jarayonini tayyorlash, ishlov vaqti normalarini hisoblashdan iborat

CAE-tizim vazifalari turli-tumandir. Sababi ular tahlil proseduralarini loyihalash, modellashtirish, loyiha qarorlarini optimallashtirish bilan bog'liqdir. Mashinasozlik CAE tizimi tarkibiga quyidagi proseduralar 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 imitasion modellashtirish.

ALTni 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 asboblarning muhiti ishlatiladi.

CALS-texnologiyalari tushunchasi.

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 zahiralarni boshqarish va ta'minlashga qaratilgan masalalar tushuniladi, CALS vazifalari esa kengroq bo'lib, sanoat mahsulotlari faoliyat davrining barcha bosqichlari bilan bog'liqdir. Bularni e'tiborga olgan holda CALS ni quyidagicha shifrdan chiqarish mumkin CALS — Computer Acquisition and LifeCycle Support.

CALSni qo'llash loyihalash ishi ko'lamini qisqartirish imkonini beradi. Sababi oldindan loyihalangan jixozlar 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 sharoitlarini 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 mashtabga ega bo'lgan umumiy tizimda joy va vaqtdan qat'iy nazar kattaliklar interpretatsiyasini va ularni bir hilda tavsiflashni ta'minlashdir. Loyiha, texnologik va ekspluatasion 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 muvoffaqiyatli bo'lishi muqarrardir. Bitta konstruktorlik hujjati turli loyuhalarda ko'p marotaba ishlatilishi mumkin. Texnologik hujjat esa turli ishlab chiqarish talablariga moslashgan. Bu esa loyihalash va ishlab chiqarish umumiy davrini qisqartirish va arzonlashtirishga olib keladi. Bundan tashqari tizim ekspluatatsiyasi soddalashadi.

Nazorat savollari:

1. Loyihalashni avtomatlashtirilgan tizimi nima?
2. Loyihalashning qanday bosqichlarini bilasiz?
3. ALT turlari va modellari?
4. Ilmiy ishlarni avtomatlashtirishning qandau vositalarini bilasiz?

Ma'ruza № 4

Mavzu: Matematik modellashtirish, sonli tahlil usullarini tizimlarda amalga oshirish. Hisobli eksperiment

Reja:

- 1. MathCAD tizimiga kirish**
- 2. MathCAD tizimining menyular xususiyati**
- 3. MathCAD tizimining komponentalari**

MathCAD tizimi va uning vazifalari. MathCAD bu matematikaning turli sohalaridagi masalalarini yechishga mo'ljallangan ajoyib sistemadir. Dasturning nomlanishi ikkita so'zdan iborat bo'lib – MAThtmatika (matematika) ba CAd (avtomatik loyihalash sistemasi).

MathCAD ni o'rganish juda oson bo'lib, uni ishlatish soddadir. Ushbu dasturni boshqarish Windows muhitida oldin ishlaganlar uchun tushinarlidir. MathCADni juda ko'p sohalarida sodda hisoblashlarni hisoblashdan tortib to elektrik sxemalarni qurishgacha ishlatish mumkin.

MathCAD formula, sonlar, matnlar va grafiklar bilan ishlaydigan universal sistemadir. MathCAD tili matematika tiliga juda ham yaqin, shu sababli unda ishlash matematiklar uchun juda oson. Zamonaviy kompyuter matematikasi matematik

hisoblarni avtomatlashtirish uchun butun bir birlashtirilgan dasturiy tizimlar va paketlarni taqdim etadi. Bu tizimlar ichida MathCAD oddiy, yetarlicha qayta ishlangan va tekshirilgan matematik hisoblashlar tizimidir.

Umuman olganda MathCAD – bu kompyuter matematikasining zamonaviy sonli usullarini qo'llashning noyob majmuasi. U o'z ichiga ko'p yillar davomida matematikaning rivojlanishi natijasida yig'ilgan tajribalar, qoidalar va matematik hisoblash usullarini jamlagan.

MathCAD paketi muhandislik va matematik hisob ishlarini bajarish uchun dasturiy vosita bo'lib, uning yordamida o'zgaruvchi va o'zgarmas parametrlar algebraik va differensial tenglamalarni yechish, funksiyalarni tahlil qilish va ularning ekstremumini izlash, topilgan yechimlarni tahlil qilish uchun jadvallar va grafiklar qurish mumkin. MathCAD murakkab masalalarni yechish uchun o'z dasturlash tiliga ega.

Hozirgi vaqtda kompyuterlarda ilmiy-texnikaviy hisoblashlarni bajarishda odatdagi dasturlash tillaridan va elektron jadvalaridan emas, balki Mathematica, MatLab, Maple, Gauss, Reduse, Eureka va boshqa turdagi maxsus matematik dasturlar keng qo'llanilyapti.

Matematik paketlar, ayniqsa Mathcad – yuqorida sanab o'tilgan ro'yxat ichida eng mashhur paket bo'lib, ilmiy – texnikaviy soha mutaxassislari dasturlashning nozik elementlariga e'tibor berilmasdan (masalan: fortran, C, paskal, BASIC va boshqalar kabi) kompyuterda matematik modellashtirishni amalga oshirishga katta yordam beradi. Quyida Mathcad matematik dasturlash muhitida ishlashning yaqqol ajralib turadigan imkoniyatlarini sanab o'tmoqchimiz:

- **Mathcad** muhitida matematik ifoda, qabul qilingan ko'rinishda ifodalanadi. Masalan, daraja yuqorida, indeks pastda, integralning yuqori va quyi chegaralari esa an'anaviy joyida turadi.

- **Mathcad** muhitida “dasturlashni” tuzish va ularning bajarilish jarayoni parallel kechadi. Foydalanuvchi **Mathcad** – hujjatida yangi ifoda kiritar ekan, uning qiymatini bira to'la hisoblash va ifodani kiritishda yo'l qo'yilgan yashiringan xatoliklarni grafigini ko'rish imkoniyati ham mavjud.

- **Mathcad** paketi yetarli darajada qudratli matematik apparat bilan qurollanganki, ular orqali tashqi pratseduralarni chaqirmasdan turib paydo bo'ladigan muammolarni hal qilishimiz mumkin.

Mathcad xos bo'lgan ayrim hisoblovchi qurilmalarni sanab o'tmoqchimiz:

- Chiziqli va chiziqli bo'lmagan algebraic tenglama va sistemalarni yechish;
- Oddiy differensial tenglama va sistemalarni (Koshi masalasi va chegaraviy masala) yechish;
- Xususiy hosilali differensial tenglamalarni yechish;

□ Berilganlarni static qayta ishlov berish (interpolyatsiya, ekstrapolyatsiya, approk-simatsiya va ko'pgina boshqa amallar);

□ Vektor va matritsalar bilan ishlash (Chiziqli algebra va boshqalar);

□ Funksional bog'liqlikning maksimum va minimumini izlash.

• **Mathcad** paketi matematik va fizik-kimyoviy formulalarga, hamda o'zgarmaslarga asoslangan yordamchi qo'llanmalar bilan boyitilgan.

• **Mathcad** paketida turli sohalar bo'yicha electron darsliklar yaratish mumkin. Masalan: oddiy differensial tenglamarni yechish, statistika, termodinamika, boshqaruv nazariyasi, materiallar qarshiligi va boshqalar bunga misol bo'la oladi.

• Foydalanuvchi o'z oldiga qo'yilgan masalani yechish bilan cheklanibgina qolmay, fizikaviy masalalarni yechishda o'lchovni hisobga olish imkoniyatiga ega. Bunda foydalanuvchi birliklar sistemasini ham tanlashi mumkin.

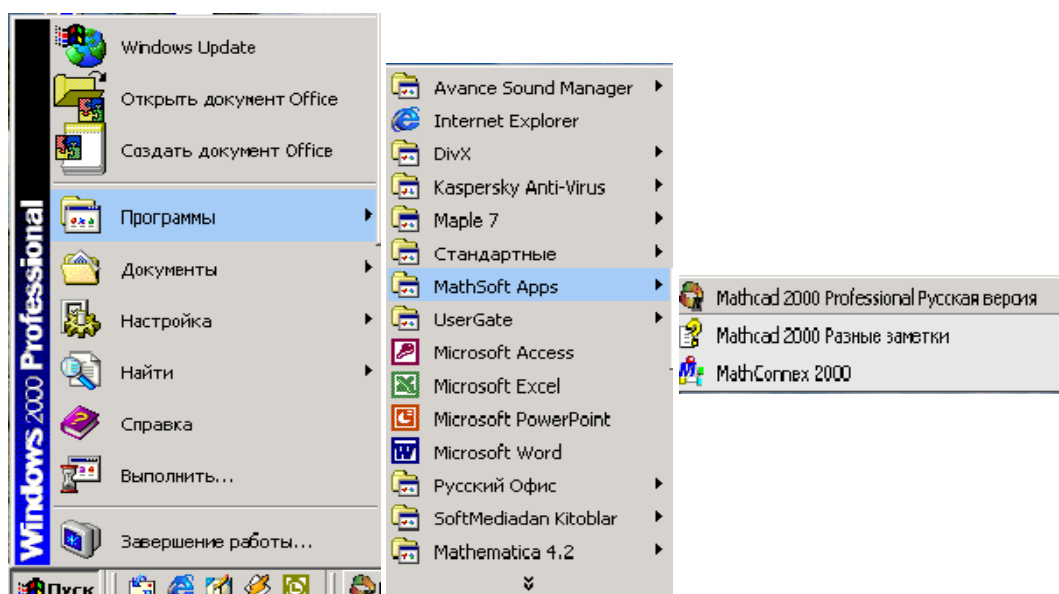
• Bundan tashqari **Mathcad** muhitida animatsiya vositasi bilan qurollangan, bunda tuzilgan modellarni nafaqat static (o'zgarmas), balki dinamik (animatsion kliplar) holda yaratish mumkin.

• **Mathcad** muhiti belgili matematika elementlari bilan boyitilgan bo'lib, bunda masalani nafaqat sonli yechish, balki analitik usulda ham yechishga imkoniyat yaratilgan.

• **Mathcad** muhitidan chiqmagan holda boshqa serverdagi hujjatlarni ishlatish va Internet tavsiya qiladigan yuqori informatsion texnologiya imkoniyatlaridan foydalanish mumkin.

Mathcad tizimida masalalarni sonli yechish bilan bir qatorda analitik usulda yechish hisobga olingan. Shuning uchun foydalanuvchilar bu dasturdan o'zlari yecha olmagan matematik masalalar uchun tayanch yechim ombori sifatida foydalanishlari mumkin. Bu tizimdan tabiiy fanlar bo'yicha electron darsliklar yaratishda asos dasturiy vosita sifatida foydalanishni tavsiya etish mumkin. Masalan differensial tenglamalarni yechish, statistika, termodinamika, boshqaruv nazariyasi kabi jarayonlarni geometric tasvirlash va animatsiyalar orqali ijro etishni yuqori darajada amalga oshirish mumkin.

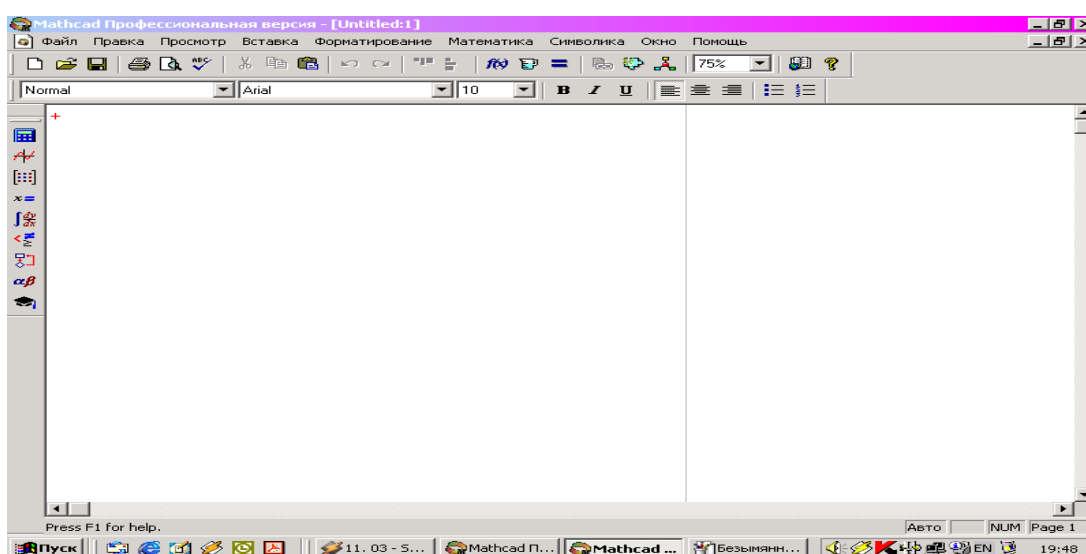
Mathcad dasturini "Пуск – Программы – MathSoft Apps - Mathcad" ketma-ketligidan foydalangan holda yuklash mumkin.



Dastur yuklanganda Mathcad muhiti zarvarag'i quyidagi ko'rinishda bo'ladi.

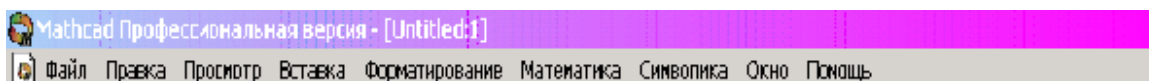


Mathcad dasturi ishga tushganda quyidagi oynani ko'ramiz.

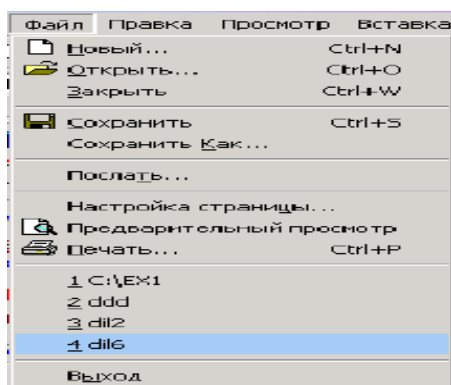


Bu oynada Mathcad programmasi interfeysining ko'rinishi bo'lib, u ishga tushishi bilan hujjat tayyorlab, uni Untitlid:1 deb nomlab foydalanuvchiga havola

etadi. Mathcad ekranining yuqori qismida "qo'shish" ko'rinishidagi kursorni ko'rasiz. Klaviaturadan kiritiladigan ma'lumotlar ushbu kursor joylashgan joydan boshlab yoziladi. Matematik misollarni yechishda, ularni kompyuter xotirasida saqlashda, grafik ko'rinishidagi tasvirlarni yaratish va qayta ishlashda Mathcad tizimining menyusi alohida o'rin tutadi. Mathcad menyu buyruqlari to'plami quyidagilardan iborat: "Файл", "Правка", "Просмотр", "Вставка", "Форматирование", "Математика", "Символика", "Окно", "Помощь".



Menyuning "Файл" bo'limida yangi hujjatlarni tayyorlash uchun yangi oyna ochish, oldingi saqlangan fayllarni yopish, tayyorlangan hujjatlarni diskka yozish, yangi oynadagi hujjatni nom berish bilan saqlash, kerakli faylni qidirib topish, matnni sahifada qanday joylashganligini oldindan ko'rish, matnni (matrisaviy, lazerli) printerlarda bir nechta nusxada, agar zaruriyat bo'lganda matnni tanlangan joyini chop etish, oxirgi to'rtta tahrir qilingan fayllar nomini ko'rish hamda Mathcad dasturdan chiqish kabi bir qator ishlarni amalga oshirish mumkin.

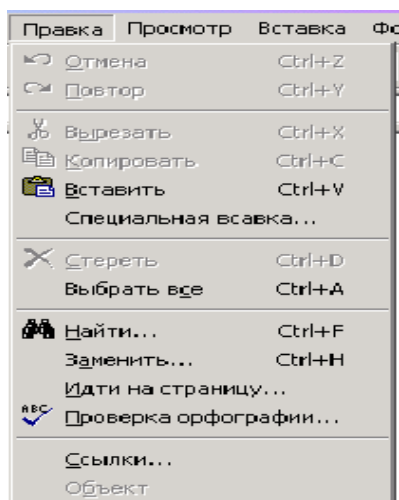


yangi fayl yaratish
matnli faylni ochish
faylni yopish
faylni xotirada saqlash
faylni nom bilan xotirada saqlas
faylni manzilga jo'natish
sahifani sozlash
sahifani ko'zdan kechirish
faylni chop qilish

oxirgi fayllar ro'yxati

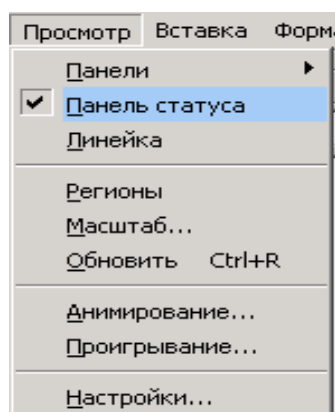
Mathcaddan chiqish

Menyuning "Правка" bo'limida hujjatni tahrir qilishga oid bir qator ishlarni amalga oshirish mumkin.



oldingi holatga qaytarish
keyingi holatga o'tish
matn qismini qirqish
matn qismidan nusxa olish
cho'ntakka olingan ma'lumotni o'rniga qo'yish
maxsus o'rniga qo'yish
o'chirish
ma'lumotni to'liq belgilash
topish
almashtirish
tanlangan varaqqa borish
orfografik xatolarni tekshirish
murojat
obe'ktlar

"Просмотр" bo'limida formulalar yozish uchun maxsus bo'limlar bilan ishlash imkoniyati bor.



panel ko'rinishini sozlash
holat satrini joriy va bekor qilish
chizg'ich yordamida sahifa o'lchashni belgilash
ob'ektlar maydonini aniqlash
masshtab tanlash
yangilash
animatsiya hosil qilish
animatsiyani ijro etish
sozlash

Menyuning "Вставка" bo'limida quyidagi ishlar amalga oshiriladi:

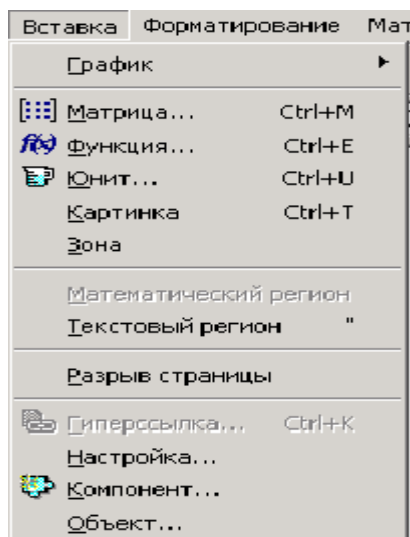
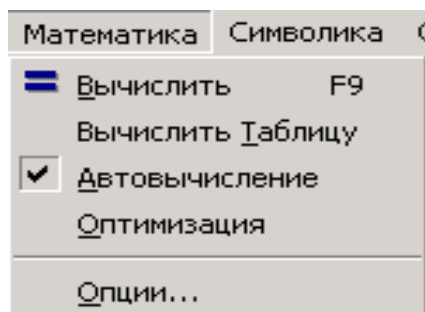


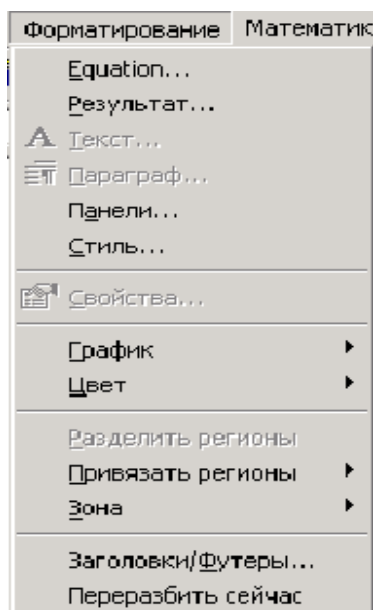
график turini tanlash
matritsa tashkil qilish
funktsiya turini tanlash
rasm tashlash
maydon hosil qilish
matematika maydoni
matn maydoni
varaqli bo'lish
gipermurojat
menu bandlari sozlash
boshqa dasturlar bilan aloqani ta'minlas
yangi ob'ekt tashkil etish

"Математика" bo'limida natijani avtomatik ravishda hisoblash va boshqa fayllarga murojaat qilish kabi ishlarni bajarish mumkin.



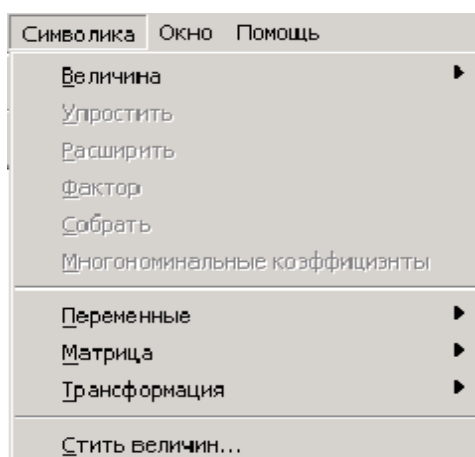
natijani hisoblash
jadval hisoblashlari
avtomatik hisoblash
optimallashtirish
boshqa fayllarga murojat qilis

"Форматирование" bo'limida sahifalar o'lchamlarini kiritish, gtafikni chizishda turini tanlash, fon tanlash va boshqa turli xil ishlarni bajarish mumkin.



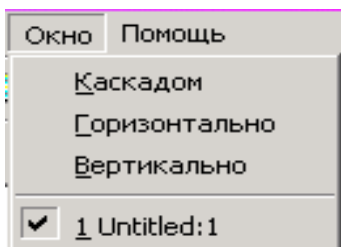
ifoda qiymatini tanlash
 natija formatini tanlash
 matn yozish
 paragraf formatini sozlash
 jadval tashkil etish
 hujjat turini tanlash
 hujjatlar haqidagi ma'lumotlarni ko'rish
 grafik turini sozlash
 rang tanlash
 qismlarga ajratish
 obe'ktlarni biri bilan bog'lash
 obe'ktlarni himoyalash, siqish, kengaytirish
 sarlavha tanlash
 qayta ajratish

"Символика" bo'limida kattalikni tanlash, yaxlitlash, yig'ish va o'zgaruvchilar ustida amallar bajarish mumkin.



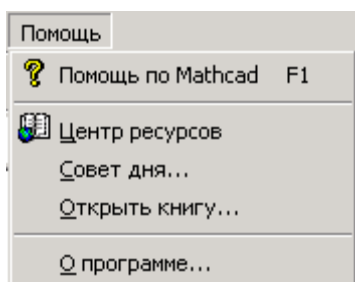
kattalik tanlash
 yaxlitlash
 kengaytirish
 фактор
 yig'ish
 o'zgaruvchilar ustida amallar bajarish
 matritsa ustida amallar bajarish
 Fur'ye va Laplas almashtirishlar bajarish
 kattalikni ajratish

"Окно" bo'limi yordamida bir nechta fayllarni ketma ket, gorizontal, vertikal joylashtirish va oynadagi mavjud fayllar ro'yxatini ko'rish mumkin.



bir nechta oynalarni ketma ket tashkil etish
 bir nechta oynalarni gorizontal joylashtirish
 bir nechta oynalarni vertikal joylashtirish
 oynadagi mavjud fayllar ro'yxati

"Помощь" bo'limi yordamida Mathcadda ishlash haqida ma'lumot olish mumkin.



Mathcad haqida ma'lumot olish
 zahiralari markazi
 kundalik maslahat
 yordam kitobini ochish
 dastur haqida ma'lumot olish

Kiritilgan hujjat ustida ishlash (uni kompyuter xotirasiga yozib qo'yish, kerakli paytda chaqirib olish, printerdan chiqarish va hokazolar) uchun Standart panelida joylashgan quyidagi tugmalar bosiladi:



Yangi fayl hosil qilish



Oldindan xotirada saqlab qo'yilgan faylni ochis



Hosil qilingan faylni yoki kiritilgan o'zgartirishlarni xotirada saqla



Joriy faylni printerdan chiqaris



Joriy faylni varoqda qanday joylashishini chamalab ko'risl



Orfografik xatolarni tekshirish



Oxirgi buyruqni bekor qilist



Bekor qilingan buyruqlarni qayta takrorlas

Yuqoridagi aytilgan tugmalar **Mathcad** Standart panelida joylashgan.

Hujjat formatini o'zgartirish.

Mathcadda katta kichik, qalin, kursiv, tagi chizilgan va hokazo formatlarda matn yozishimiz mumkin. Shu maqsadda yozadigan matnimiz yoki belgilangan matn uchun formatni **Formatting** panelidagi



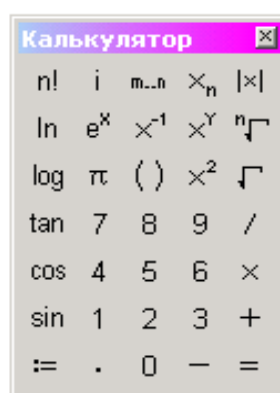
tugmalar orqali tanlashimiz kerak. Ushbu tugmalarning vazifalari quyidagi jadvalda keltirilgan.

Tugmachaning ko'rinishi	Tugmaning bajaradigan funksiyasi
	Berilgan yoki endi yoziladigan matnning shriftini tanlash. Masalan: Times New Roman, Arial, Courier va hokazo (Windowsda o'rnatilgan shriftlardan biri)
	Belgilangan yoki endi yoziladigan matn shriftining o'lchovi ko'rsatiladi. Masalan: 10, 13.5, 14, 20, 28
	Belgilangan yoki endi yoziladigan matnning qalin yoki oddiyligini belgilash
	Belgilangan yoki endi yoziladigan matnning kursiv yoki oddiyligini belgilash
	Belgilangan yoki endi yoziladigan matnning tagi chizilgan yoki oddiyligini belgilash
	Belgilangan yoki endi yoziladigan absatsni chap tomondan joylashtirish
	Belgilangan yoki endi yoziladigan absatsni o'rtadan joylashtirish
	Belgilangan yoki endi yoziladigan absatsni o'ng tomondan joylashtirish
	Har bir abzatsni marker (biror belgi) bilan belgilash Birinchi absats Ikkinchi absats va hokazo
	Har bir abzatsni nomerlash Birinchi absats Ikkinchi absats va hokazo
	Belgilangan yoki endi yoziladigan matnga maqomni tanlash. Maqom, oldindan o'rnatilgan formatlarni o'zida saqlab turadi. Maqomni o'zimiz hosil qilishimiz mumkin. Shu maqsadda kerakli formatlarni maqomda o'rnatamiz va ushbu maqomni matnga tayinlaymiz.

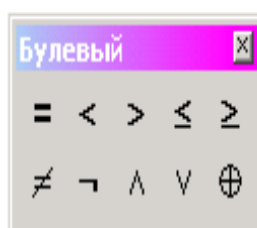


arifmetik hisob kitoblar bajariladi
ixtiyoriy turdagi grafik hosil qilinadi
vektor va matritsalar ustida amallar bajariladi
tenglamalar hisoblanadi
integral, differensial, limit, ko'paytma, yig'indilar hisoblanadi
munosabat va mantiqiy amallardan foydalanish mumkin
dasturda kerak bo'ladigan operatorlar tanlanadi
grek alifbosidan foydalanish mumkin
ifodalar analitik hisoblanadi

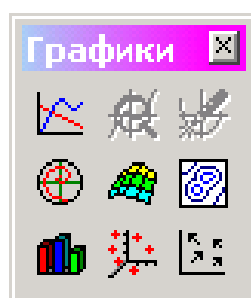
Bu paneldagi  tugmasi arifmetik hisob kitoblarni hal qilishda juda qulay vosita hisoblanadi. Bu tugma bosilganda quyidagi panel paydo bo'ladi.



 tugma yordamida munosabat va mantiqiy amallardan foydalanish mumkin. Bu tugmani bosganda quyidagi panel hosil bo'ladi.

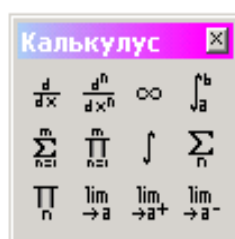


 tugma orqali ixtiyoriy turdagi grafikni hosil qilish mumkin. Bu tugmani bosganda quyidagi panel paydo bo'ladi.

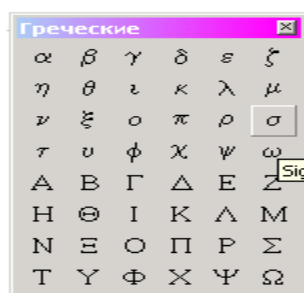




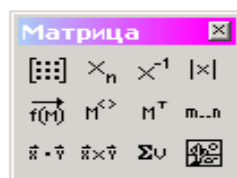
bu tugma orqali integral, differensial, limit, yig'indi va ko'paytmalarni hisoblash mumkin. Bu tugma bosilganda quyidagi panel paydo bo'ladi.



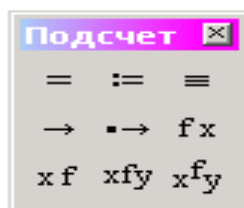
tugma orqali grek alifbosidan foydalanish mumkin. Bu tugma bosilganda quyidagi panel paydo bo'ladi.



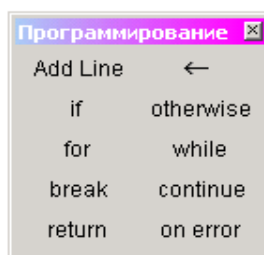
tugma orqali vektorlar va matritsalar ustida amallar bajarish mumkin. Bu tugma bosilganda quyidagi panel paydo bo'ladi.



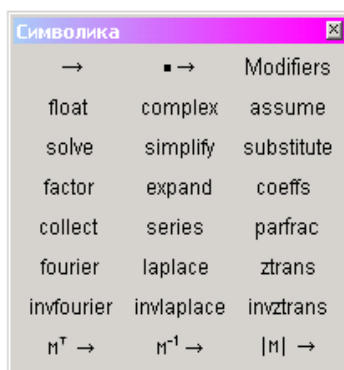
tugma orqali tenglamalarning ildizi topiladi.



tugma orqali dasturlashda kerak bo'ladigan operatorlarni tanlash mumkin.



tugma orqali ifodalarni analitik hisoblash mumkin.



Nazorat savollari:

1. MathCad dasturi haqida asosiy tushunchalar
2. MathCad dasturining interfeysi
3. MathCad dasturining miyular satri
2. MathCad dasturida formulalar bilan ishlash
3. MathCad dasturida hujjatlarni formatlash

Ma'ruza № 5

Mavzu: Grafik modellashtirish. Loyihalash jarayonida amaliy dasturlarni grafik imkoniyatlaridan foydalanish.

Reja:

1. Grafik muharir haqida ma'lumot
2. Adobe Photoshop dasturi
3. CorelDraw grafik muharriri
4. Macromedia Freehand
5. Adobe Illustrator

Axborotning asosiy qismini inson ko'rish a'zolari orqali oladi. Ko'rgazmali axborotning o'zlashtirilishi oson bo'ladi. Inson tabiatining ana shu xususiyati grafik operatsion tizimlarda ishlatiladi. Ularda axborot grafik ob'ektlar: znachokar (belgilar), oynalar va rasmlar ko'rinishida tasvirlanadi.

Operatsion tizimning barcha grafik ob'ektlari, shuningdek, boshqa barcha tasvirlar qandaydir yo'l bilan kompyuterda xosil qilinishi yoki unga kiritilishi kerak. Grafik tasvirlarni kompyuterga kiritish uchun maxsus tashqi (atrof) qurilmalari ishlatiladi. Ular bilan siz 3-bobda tanishib chiqqansiz. Eng ko'p tarqalgan qurilma — bu skanerdir. So'nggi paytda raqamli fotokameralarning ham qo'llanish ko'lemi kengayib bormoqda. Ularning oddiy fotoapparatlardan farqi shundaki, tasvir kimyoviy yo'l bilan fotoplyonkaga tushirilmaydi, balki fotokamera xotirasining mikrosxemalariga yozib qo'yiladi. U erdan axborotni kabel orqali kompyuterga uzatish mumkin. Ayrim raqamli fotoapparatlar ma'lumotlarni fayl sifatida egiluvchan diskka yozib quyish imkoniyatiga ham ega. Diskdagi axborotni esa kompyuterga o'tkazish unchalik qiyin emasligini siz yaxshi bilasiz.

Tasvirni kompyuterga videokameradan ham kiritish mumkin. Videoning ketma-ketlikdagi biror kadrni tanlashi va uni kompyuterga kiritishi tasvirni ushlab olish deyiladi.

Kompyuterga tasvirni kiritish uchun uni albatta skanerlash, rasmga olish yoki uni ushlab olish shart emas. Tasvirni kompyuterning o'zida ham xosil qilish mumkin. Buning uchun grafik muharrirlar deb ataluvchi maxsus dasturlar sinfi ishlab chiqilgan.

Axborotni grafik shaklda ishlab chiqish, taqdim etish, ularga ishlov berish, shuningdek, grafik ob'ektlar va fayllarda bo'lgan nografik ob'ektlar o'rtasida bog'lanish o'rnatishni Informatikada kompyuter grafikasi deb atash qabul qilingan. Kompyuter grafikasi uch turga bo'linadi: rastrli grafika, vektorli grafika va fraktal grafika. Ular o'rtasidagi asosiy farq nurning displey ekrandan o'tish usulidan iborat. Eslab qoluvchi elektron-nurli trubka (ENT)larga ega vektorli qurilmalarda nur berilgan traektoriya bo'ylab bir marta chopib o'tadi, uning izi esa ekranda keyingi buyruq berilgungacha saqlanib qoladi. Demak, vektorli grafikaning asosiy elementi — chiziqdor.

Vektorli grafika bilan ishlovchi dasturiy vositalar birinchi navbatda tasvirlarni yaratishga mo'ljallangan. Bunday vositalar reklama agentliklarida, dizaynerlik byurolarida va nashriyotlarda qo'llaniladi.

Rastrli qurilmalarda esa tasvir ularni tashkil etuvchi nuqtalar majmuasidan vujudga keladi. Bu nuqtalar piksellar (pixels) deb ataladi. Rastr — bu ekranning butun maydonini qoplovchi piksellar matritsasi. Demak, rastrli grafikaning asosiy elementi nuqtadan iborat.

Rastrli grafika vositalari bilan tayyorlangan tasvirlar kompyuter dasturlar yordamida kamdan - kam xoldagina yaratiladi. Ko'pincha ushbu maqsadda rassom tayyorlagan tasvirlar yoki rasmlar skanerlanadi. Rastrli tasvirlar bilan ishlashga

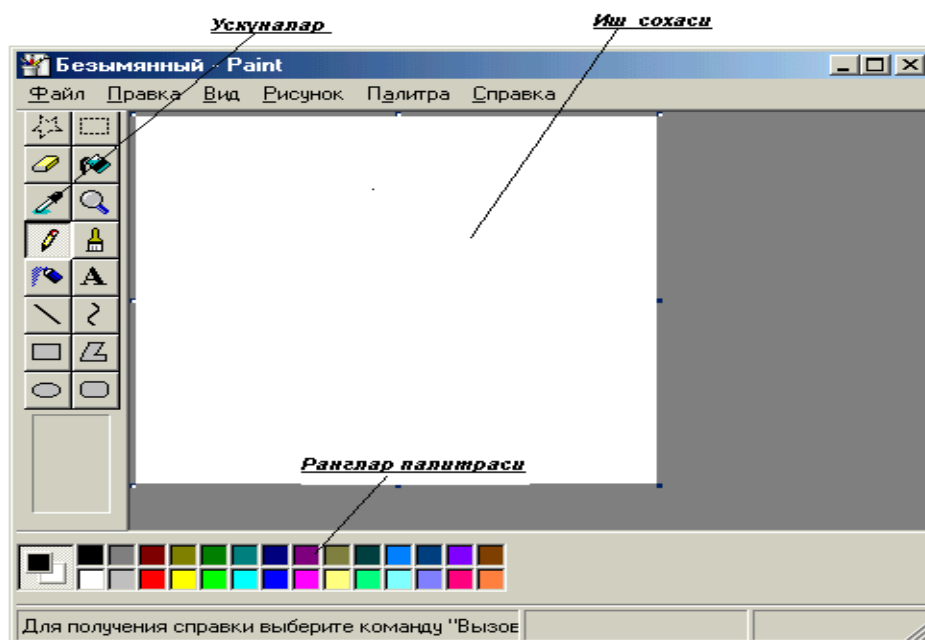
mo'ljallangan ko'pgina grafik muharrirlar asosan tasvirlarga ishlov berishga mo'ljallangan. Internet tizimida ko'proq rastrli tasvirlar qo'llanilmoqda.

Fraktal badiiy kompozitsiyani yaratish — bu tasvirni chizish yoki jixozlash emas, balki uni dasturlashdir, ya'ni bunda tasvirlar formulalar yordamida ko'riladi. Fraktal grafika odatda o'yin dasturlarida qo'llaniladi.

Paint grafik muharriri rastrli tasvirlar bilan ishlashga mo'ljallangan. U quyidagi buyruqlar ketma-ketligini bajarish bilan ishga tushiriladi:

Пуск->Программы->Стандартный-> Paint

Ayrim hollarda Paint yorlig'i WINDOWS ish stoliga ko'chirilgan bo'ladi. Bunday holda Paint yorlig'i ustida sichqoncha tugmachasini bosish orqali dasturni tezda ishga tushirish mumkin. Shundan so'ng ekranda Paint dasturining ishchi oynasi (darchasi) ochiladi. U bir necha sohalardan iborat.



Oynaning asosiy qismini ish sohasi egallaydi. Uning chap yonida uskunalar paneli joylashgan. Unda tasvir yaratishda ishlatiladigan uskunalar tugmachalari (ramziy belgilari) joylashtirilgan. Ayrim uskunalar tanlanganda panelning pastida ushbu uskunaning xossalari qo'shimcha sozlash uchun darcha paydo bo'ladi.

Ish soxasining pastida ranglar palitrasi joylashgan. U rasm chizishda ishlatiladigan ranglar to'plamini o'z ichiga olgan

Тасвирдан ихтиёрий булакни ажратиш

Учиргич

Ранг танлаш

Калам

Туркагич

Тугри чизик

Туртбурчак чизик

Эллипс



Тасвирдан туртбурчак булакни ажратиш

Сохани ранглаш

Тасвирни кенгайтириш

Мўйкалам

Ёзувларни кiritиш

Эри чизик чизик

Купбурчак

Овал

Ranglar palitrasi ranglari o'zgarib turuvchi kvadratchalardan iborat. Rangni o'zgartirish uchun sichqoncha belgisini tanlangan rang ustiga olib kelib, tugmachasini bosish kerak. Agar chap tugmacha bosilsa asosiy rang (yo'qori kvadratcha rangi), ung tugmacha bosilsa fon rangi (quyi kvadratcha rangi) o'zgaradi.



Kompyuter grafikasi uch turga bo'linadi: rastarli grafika, vektorli grafika va frontal grafika. Ular bir-biridan monitor ekranida tasvirlanishi va qog'ozda bosib chiqarilishi bilan farqlanadi.

Ma'lumki, Internetda rastarli grafika keng tarqalgan bo'lib, u bilan ishlash uchun esa ko'pincha Adobe PhotoShop dasturidan foydalaniladi.

“Adobe Photoshop 7.0” rasm tahrirlagichi

“Adobe Photoshop” Windows muhitida ishlovchi Macintosh va IBM PC kompyuterlari uchun mo'ljallangan elektron ko'rinishdagi foto tasvirlarni tahrirlovchi dasturdir. “Adobe Photoshop” dasturi Adobe System, Inc kompaniyasi tomonidan ishlab chiqarilgan bo'lib, ishlatishdagi alohida qulayliklari bilan mashhur.

“Adobe Photoshop” tasvir tahrirlagichi yordamida fotosuratlarga ko'shimchalar kiritish, fotosuratdagi dog'larni o'chirish va eski rasmlarni qayta ishlash va tiklash, rasmlarga matn kiritish, qo'shimcha maxsus samaralar bilan boyitish, bir fotosuratdagi elementlarni ikkinchi fotosuratga olib o'tish, suratdagi ranglarni o'zgartirish, almashtirish mumkin. “Adobe Photoshop” imkoniyatlari keng qamrovli bo'lib, u gazeta va jurnallarni turli-tuman rasmlar bilan boyitishda juda katta qulayliklar yaratadi.

“Adobe Photoshop” ayniqsa jurlanistlarning, rassomlarning ijodiy imkoniyatlarini to'la amalga oshirishlarida yordam beradi. Jurnalistika va bevosita

matbuot yoki nashriyot sohasiga aloqador bo'lgan shaxslarning mazkur dastur bilan ishlashni bilishi ular uchun qo'shimcha imkoniyatlarni yaratib beradi.

“Adobe Photoshop” tasvir tahrirlagichi juda murakkab dasturdir. Foydalanuvchilar uning asosiy imkoniyatlaridagina foydalanadilar xolos.

Adobe Photoshop dasturini ish jarayoniga tayyorlash

“Adobe Photoshop” dasturi quyidagi tartibda ishga tushiriladi: “Adobe Photoshop” ni yuklash va undan chiqish.

“Adobe Photoshop” dasturini ishga tushirish uchun sichqoncha Photoshop belgisida ikki marta bosiladi.

Ekrandagi mavjud “Adobe Photoshop” uchun maxsus belgida sichqonchaning chap tugmasi ikki marta bosiladi.

“Adobe Photoshop 7.0” dasturidan chiqish

“Adobe Photoshop 7.0” dasturidan chiqish uchun quyidagi usullarning biridan foydalanish mumkin:

Alt+F4 tugmachalarini bosish.

“Fayl” tavsiyanomasining выход buyrug'ini tanlash

Ekraning yuqori qismi o'ng burchagida joylashgan X belgisini bosish yoki Закрыть buyrug'ini bajarish.

Savollarga quyidagicha javob berishingiz zarur:

ДА - so'nggi kiritilgan o'zgartirishlarni saqlab va “Adobe Photoshop” dasturidan chiqish uchun.

НЕТ - kiritilgan o'zgartirishlar saqlanmagan holda “Adobe Photoshop” dasturidan chiqish uchun.

Отмена - “Adobe Photoshop” dasturidan ishlashni davom ettirish uchun.

“Adobe Photoshop” dasturi ishga tushirilgandan so'ng ekranda “Adobe Photoshop” tasvir tahrirlagichi darchasi hosil bo'ladi.

“Adobe Photoshop” darchasining yuqori qismida sarlavha satri va Windowsga xos elementlar joylashadi. Sarlavha satridan so'ng tavsiyanoma satri joylashadi. Tavsiyanomadagi kerakli buyruqlarni tanlashingiz mumkin.

Quyida asosiy tavsiyanoma buyruqlari tavsifi keltirilgan.

“Adobe Photoshop” dasturi tavsiyanomasi 9 banddan iborat. Har bir tavsiyanoma tarkibida ochiladigan tavsiyanoma bandlari mavjud. Ularni ko'rish

kursor yordamida amalga oshiriladi. Quyida eng ko'p qo'llaniladigan buyruqlarning qisqacha tavsifi keltirilib o'tiladi.

FAYL tavsiyanomasi tarkibi:

Buyruq nomi	Tasnifi
Новый Ctrl+N	Yangi fayl yaratish
Открыть Ctrl+O	Fayllarni diskdan o'qish. Bu buyruq yordamida diskda mavjud fayllar ochiladi.
Открыть Alt+Ctrl+O	как Faylni qanday ko'rinishda ochishni tanlash.
Сохранить Ctrl+S	Faylni xotiraga mavjud formatda joylashtirish.
Сохранить Shift+Ctrl +S	как Faylni xotiraga boshqa nom bilan yozish. Ushbu buyruq fayl nomi, formati va direktoriyasi kabi atributlarini o'zgartirishda foydalaniladi.
Сохранить Alt+Ctrl +S	копий Tasvir nusxasini xotiraga joylash.
Вернуть	Tasvirning dastlabki holatiga kaytish.
Поместить	Boshqa mustaqil fayl bilan birlashtirish
Импорт	Boshqa direktoriyada joylashgan faylni "Adobe Photoshop" dasturiga olib kirish.
Экспорт	Tasvirni boshqa direktoriyaga jo'natish.
Файл информация	Fayl haqidagi ma'lumotlarni kiritish
Установка страницы Shift+Ctrl +P	Tasvirni printer yordamida chop etishga tayorlash, qog'oz shaklini tanlash.
Печать Ctrl+P	Tasvirni printerga jo'natish.
Предпочтения	"Adobe Photoshop" dasturini kerakli tartibda sozlash.
Настройка света	Tasvir ranglarini sozlash.
Adobe online	Internet bilan bog'lanish.
Выход Ctrl+Q	"Adobe Photoshop" dasturidan chiqish.

ПРАВКА tavsiyanomasi tarkibi:

Buyruq nomi	Tavsifi
-------------	---------

Верн Ctrl+Z	Tasvir ustida bajarilgan oxirgi amalni bekor qilish.
Резать Ctrl+X	Tasvirning ajratilgan qismini muvaqqat xotiraga olish.
Копировать Ctrl+C	Nusxa olish.
Вставить Ctrl+V	Muvaqqat xotiradan kursor ko'rsatgan joyga qo'yish.
Вставить Shift+Ctrl+V	В Muvaqqat xotiradan belgilangan joyga qo'yish.
Очистит	Tasvirda belgilangan maydonni tozalash, o'chirish. Bunda o'chirilgan maydon fon rangiga bo'yaladi.
Залить	Tasvir yuzasini asosiy rang bilan bo'yash.
Штрих	Tasvirda belgilangan maydonni shtrixlab ko'rsatish
Трансформация Ctrl+T	Tasvir shaklini o'zgartirish.
Трансформ	Tasvir shaklini turli ko'rinishlarda o'zgartirish.
Очитска	“Istoriya” darchasida tasvir olib borilgan o'zgartirish amallarini butunlay o'chirish. Bu amal bajarilgandan so'ng o'zgartirishlarni ortga kaytarish mumkin emas.

ИЗОБРАЖЕНИЕ tavsianomasi tarkibi:

Буурууq nomi	Tavsifi
Holat	Rang modellarni o'zgartirish.
Настройка	Tasvir ranglarini sozlash.
Дубликат	Tasvirdan nusxa olish.
Наложить изображение	Tasvirni qo'shimcha ranglar bilan boyitish.
Вычисление	Tasvirdagi ranglar kanallarini o'chirish.
Размер изображения	Tasvir shaklini va o'lchamlarini o'zgartirish.
Размер холста	Tasvir ramkasi o'lchamlarini o'zgartirish.
Обрезание	Belgilangan maydondagi tasvirni kesib olish
Перевернуть Холст	Xolstni soat strelkasi bo'ylab yoki soat strelkasiga qarshi 180, 90 gradus burish.
Гистограмма	Tasvirdagi ranglar miqdori haqidagi ma'lumotlar darchasi.

СЛОЙ tavsiyanomasi tarkibidagi qo'shimcha buyruqlar:

Buyruq nomi	Tavsifi
Новый	Yangi qatlamni hosil qilish
Дублировать слой	Qatlam nusxasini hosil qilish
Удалить слой	Mavjud qatlamni muvaqqat xotiradan o'chirish
Самара	Qatlamga turli samaralarni qo'shish
Группа с предыдущим StrlQG	Qatlamlarni bir-biriga birlashtirish
Разгруппировать ShiftQCtrlQG	Qatlamlarni bir-biridan ajratish
Склеить все слои	Mavjud barcha qatlamlarni birlashtirish.

Выделить tavsiyanomasi tarkibi :

Buyruq nomi	Tavsifi
Все Ctrl+A	Tasvirni belgilash
Убрать выделение Ctrl+D	Tasvirning belgilangan qismini muvaqqat xotiradan o'chirish.
Выделить заново Shift+Ctrl+D	Qaytadan belgilash
Обратно Sift+Ctrl+I	So'nggi bajarilgan amalni qaytarish
Световой ряд	Tasvirdagi ranglar asosida belgilash maydonini aniqlash.
Модифицировать	Belgilash chizig'ini piksellarda kengaytirish.
Увеличить	Belgilash maydonini kengaytirish
Преобразовать выделение	Belgilangan maydon shaklini o'zgartirish
Сохранить выделение	Belgilangan maydon shaklini xotiraga joylashtirish.

ФИЛЬТРЫ tavsiyanomasi tarkibidagi qo'shimcha buyruqlar:

Фильтры tavsiyanomasi tarkibida keltirilgan buyruqlar tavsifi bilan o'quv qo'llanmaning filtrlar bilan ishlash bobida batafsil tanishishingiz mumkin.

Вид tavsiyanomasi tarkibi:

Buyruq nomi	Tavsifi
Новый вид	Asosiy tasvirni yangi darchada ochish.
Увеличить Ctrl++	Tasvirning ekrandagi ko'rinishini kattalashtirish
Уменьшить Ctrl+-	Tasvirning ekrandagi ko'rinishini kichraytirish
Показать весь экран	Tasvirni butun ekranga yoyish
Реальный размер	Tasvirning haqiqiy o'lchamlardagi ko'rinishi
Размер печатного оттиска	Tasvirning bosma shakldagi ko'rinishi
Выкл линейки Ctrl+R	Chizg'ichlarni o'rnatish

ОКНО tavsiyanomasi tarkibi:

Buyruq nomi	Tavsifi
Каскад	Tasvirni ekranda vertikal holatda tasvirlash
Мозаика	Tasvirni ekranda gorizontal holatda tasvirlash.
Упорядочить значки	Asboblar panelidagi buyruqlarni tartibli joylashtirish.
Заккрыть все	“Adobe PhotoShop” dasturi darchasida ochilgan barcha tasvirlarni berkitish.
Вкл панель	Asboblar panelini o'chirish yoki yoqish.
Вкл навигатор	Navigatorning ekranda paydo bo'lishini ta'minlash.
Показать информацию	Axborotlar darchasini faollashtirish.
Показать цвет	Ranglar joylashgan maxsus darchani faollashtirish
Вкл кисти	Bo'yoq chetkalari joylashgan darchani faollashtirish.
Вкл слой	Qatlamlar haqidagi ma'lumotlarni saqlovchi darchani faollashtirish.
Показать дейтсвия	Tasvirlar bilan ishlashda bajarilgan barcha amallar haqidagi ma'lumotlar darchasini faollashtirish.
Убрать строки состояния	“Adobe PhotoShop” dasturi darchasi ostidagi ma'lumotlar satrini o'chirish yoki yoqish.

ПОМОЩЬ tavsiyanomasi tarkibida “Adobe Photoshop” dasturiga taalluqli barcha ma'lumotlar joylashgan. Bu tavsiyanoma tarkibidagi buyruqlar yordamida

“Adobe Photoshop” dasturi darchasidagi asboblarni qisqacha nima vazifani bajarishi haqidagi ma’lumotlarni o’qish mumkin.

Asboblari paneli

“Adobe Photoshop” dasturi darchasida turli asboblari joylashgan. Bu panel turli tugmalardan iborat bo’lib, har bir tugmachasi “Adobe Photoshop” dasturining biror buyrug’ini anglatadi. Agar darchada asboblari paneli bo’lmasa tavsiyanoma satrining Okno punktida Vkl panel buyrug’ini tanlang.

“Adobe Photoshop” dasturida jami 46 ta asboblari mavjud bo’lib, ulardan 20 tasi bevosita dastur ishga tushirilganda darchada ko’zga tashlanib turadi. Qolganlarini qo’shimcha buyruqlarni bajarish orqali ishga tushirish mumkin. Agar asboblari panelida joylashgan tugmachaning ostki qism o’ng burchagida kichik uchburchak shakli tasvirlangan bo’lsa, bu tasvir ushbu tugmachasi tarkibida o’xshash buyruqni bajaruvchi asboblari yashiringanlidan darak beradi.

Yashiringan asbobni faollashtirish uchun kursorni maxsus belgili tugmachasi ustidan sichqonchaning chap tugmasini bosgan holda asboblari panelidan tashqariga olib chiqiladi va kerakli tugmachasi ustida kursorni qoldirib sichqonchaning chap tugmachasi qo’yib yuboriladi.

Har bir tugmachasi kursor yaqinlashtirilsa kursor belgisi ostida asbobning qanday vazifani bajarishi haqidagi axborot paydo bo’ladi.

Quyida “Adobe Photoshop” dasturida ishlash jarayonida keng qo’llaniladigan asboblarning qisqacha tavsifi keltirib o’tiladi:

	Прямоугольная область: Tasvirda to’g’ri to’rtburchak shaklidagi maydonni belgilab olish uchun qo’llaniladi. Bu asbob yordamida tasvirdagi alohida maydonni belgilab olingandan keyin tasvirga kiritilgan barcha o’zgarishlar faqatgina belgilangan maydon ichiga ta’sir etadi. Ushbu tugmachaga qo’shimcha tarzda “Shift” klavishi ishlatilsa, belgilangan maydon hududi ortadi. “Shift” tugmasi o’rnida “Alt” tugmasi qo’llanilgan taqdirda belgilangan maydon hududi qisqaradi. Ushbu amal “Lasso” va “Volshebnoy palochke” asboblari bilan ishlashda qo’llaniladi.
	Эллиптическая область: Tasvirda doira shaklidagi maydonni belgilab olish uchun qo’llaniladi. Bu asbob yordamida tasvirdagi alohida maydonni belgilab olingandan keyin tasvirga kiritilgan barcha o’zgarishlar faqatgina belgilangan maydon ichiga ta’sir etadi.
	Строка пикселей: Tasvirda gorizontali shakldagi chiziqni belgilaydi. Amalda bu asbob juda kam qo’llaniladi.
	Столбец пикселей: Tasvir yuzida vertikal chiziqni belgilaydi. Amalda bu asbob juda kam qo’llaniladi.

	Кадрирование: Ushbu asbob asosan tasvir chetlarini va keraksiz qismlarini kesib tashlash uchun qo'llaniladi. Kadrirovanie buyrug'i faollashtirilganda tasvir yuzasida to'g'ri-to'rtburchak shaklidagi ramka hosil bo'ladi. Ramkaning chetlari kichik kvadratchalardan iborat bo'lib, bu kvadratchalar yordamida ramka hajmi o'zgartiriladi. Tasvir ramka ostiga olingandan so'ng "Enter" tugmasi bosilsa, ramka tashqarisida qolgan ortiqcha bo'laklar kesib tashlanadi. Ushbu buyruqni "Esc" tugmasini bosib rad etish mumkin.
	Перемещение: Ushbu asbob tasvirdagi belgilangan maydonni yoki qatlamni siljitish va kesib olish uchun xizmat qiladi. Ba'zan Peremehenie buyrug'i bajaradigan ayni jarayonni boshqa ayrim asboblarda ham amalga oshirish mumkin. Masalan Volshebnaya palochka asbobini ishlatish chog'ida
	Лассо: tasvirdagi turli shakldagi obyektlarni belgilash uchun ishlatiladi.
	Многоугольное лассо: Asosan tasvirdagi to'g'ri chiziqlardan iborat obyektlarni belgilashda ishlatiladi. "Alt" tugmasi bilan qo'llanilganda oddiy Lasso asbobi vazifasini bajaradi.
	Магнитное лассо: Asbobi ishlatilganda "Adobe PhotoShop" dasturi tasvirdagi obyekt chegaralarini o'zi belgilaydi. Ammo bu asbob piksellardagi ranglarni o'zgarishiga bogliq tarzda chegaralarni aniqlashi bois kam qo'llaniladi.
	Волшебная палочка: Bir-biriga yaqin bo'lgan rangdagi piksellar joylashgan maydonni belgilaydi. "Shift" tugmasi bilan birgalikda qo'llansa belgilangan maydon hajmi ortadi. "Alt" bilan ishlatilganda esa, belgilangan maydon hajmi kamayadi.
	Аэрограф: Tasvirni bo'yashda ishlatiladi. Aerografni bir joyda ushlab turish siyohni tasvir bo'ylab yoyilib ketish samarasini beradi. Buyoqning tasvir bo'ylab oqishi kursorni qo'yib yubormaguncha davom etadi. Odatda bu asbob bilan yumshoq cho'tkalar ishlatiladi. Aerograf kursorni ushbu asbob ustida bosish yoki klaviaturadagi J tugmachasini bosish orqali faollashtiriladi.
	Кист: Aerograf asbobi kabi tasvirni bo'yashda ishlatiladi. Ammo Kist yordamida tasvirni sifatli bo'yash mumkin. Bu asbob Aerografga nisbatan ko'p qo'llaniladi. Kist asbobini V klavishasini bosish orqali faollashtirish mumkin. Brushes darchasi yordamida buyoq cho'tkalarining shaklini o'zgartirish mumkin (8-rasm).
	Штамп: Tasvirdagi kichik bir bo'lak nusxasini ko'chirish uchun ishlatiladi. Bu asbob tasvirdagi ayrim nuqsonlarni, dog'larni yo'qotish va eski rasmlarni tiklashda keng qo'llaniladi.
	Кист предыдущих состояний: Bu asbob tasvir haqidagi dastlabki ma'lumotlar asosida ishlaydi. Uning yordamida tasvirga kiritilgan so'nggi o'zgartirishlarni bekor qilish mumkin.
	Ластик: Tasvirni o'chirish uchun ishlatiladi. U qo'llanganda tasvirda fon qaysi rangda bo'lsa, o'sha rangdagi chiziqlar hosil bo'ladi. "Alt" tugmasini qo'llash lastochka yordamida kompyuter xotirasiga olinmagan so'nggi o'zgartirishlarni bekor qilish mumkin. Ластик asbobi "E" klavishasini bosish orqali faollashtiriladi.

	Карандаш: Turli chiziqlarni chizish uchun foydalaniladi. "Alt" klavishi bosilganda kursorning ekrandagi tasviri o'zgaradi va bevosita tasvirdan kerakli rangni tanlash mumkin. Bu amal bajarilgandan so'ng Карандаш o'sha rangda chiziq tortadi.
	Линия: To'g'ri chiziqlarni chizishda qo'llaniladi.
	Размывка: Ushbu asbob ishlatilganda, tasvirdagi yorqinlik pasayadi. "Alt" tugmasi bilan qo'llanganda yorqinlik ortadi.
	Резкость: Ushbu asbob ishlatilganda tasvirdagi yorqinlik ortadi. "Alt" klavishi bilan qo'llanganda esa tasvir xiralashadi.
	Палец: Tasvirdagi ranglarni chayqaltirib, tasvirdagi obyektlar o'rtasidagi chegaralarni bir-biriga qo'shishga xizmat qiladi.
	Осветитель: O'z nomi bilan piksellardagi ranglar yorqinlashadi. "Alt" klavishi bilan qo'llanganda esa piksellardagi ranglar xiralashadi.
	Заменитель: Tasvir ustida harakatlantirilganda piksellardagi ranglar qoramtir tus oladi.
	Губка: Tasvir ustida harakatlantirilganda, tasvirdagi ranglar miqdori pasayadi. Gubka bir joyda ko'p harakatlantirilsa tasvirning o'sha joyi kulrang tus oladi.
	Перо: Peroni tasvir ustida harakatlantirilganda, nuqtalar hosil bo'ladi. Ushbu nuqtalar yordamida chizilgan tasvirni o'zgartirish mumkin.
	Магнитное перо: Bu asbob xuddi "Magnitnoe Lasso " kabi harakatlanadi. Biror bir tasvirdagi obyekt atrofida harakatlantirilganda, "Adobe PhotoShop" dasturining o'zi obyekt chetlarini belgilab chiqadi.
	Произвольное перо: Juda qulay asbob bo'lib, xohlagan shakldagi tasvirni u yordamida ifodalash mumkin.
	Вставить точку: Bu asbob "Pero" yordamida chizilgan chiziq ustiga qo'shimcha nuqtalarni qo'shadi.
	Удалить точку: "Pero" yordamida chizilgan chiziq ustidagi ortiqcha bo'lgan nuqtalarni o'chiradi.
	Непосредственное выделение: U yoki bu "Pero" bilan chizilgan chiziqlarni tahrirlash uchun xizmat qiladi. Uning yordamida chiziqdagi nuqtalarni yakka tartibda harakatlantirish va kerakli joyga siljitish mumkin.
	Преобразовать точку: Tasvir ustida chizilgan chiziqchalarda o'rnatilgan har bir nuqta burchak yoki yoy vazifasini bajaradi. Ushbu asbob yordamida nuqtalarning vazifalarini o'zgartirish, ya'ni yoyni burchakka va burchakni yoyga almashtirish mumkin. Buning uchun kursorni nusxa ustiga olib borib sichqonchaning chap tugmasi bir marta bosiladi.

	Текст: Ushbu asbob yordamida tasvirga turli matnlarni kiritish mumkin. Tekst asbobi faollashtirilib, kursor tasvir ustida bosilsa matn kiritish uchun alohida darcha hosil bo'ladi. Bu darchada harf o'lchami, turi, rangi va boshqa o'lchamlari kiritiladi. Bu asbob yordamida kiritilgan matnni qayta tahrirlash imkoni mavjud emas.
	Текст маска: "Tekst" asbobi kabi bu asbob faollashtirilib, matn ustida bir marta bosilganda, "Tekstovo'y instrument" darchasi hosil bo'ladi. Lekin bu matn oddiy matndan tubdan farq qiladi. Harflarning cheti xuddi "Lasso" asbobida belgilash kabi ko'rinishga ega bo'ladi. Harflarni turli ranglarga bo'yash va "Перемещение" asbobi yordamida o'rnidan siljitish yoki boshqa rasmga olib o'tish mumkin.
	Вертикальный текст: Agar tasvirga pastdan yuqoriga shaklda vertikal shaklda matn kiritmoqchi bo'lsangiz ushbu asbobdan foydalanishingiz mumkin.
	Вертикальная текст маска: Huddi "Tekst maska" asbobi kabi bir xil vazifani bajaradi. Ammo bu asbob qo'llanganida harflar ustma-ust ustun kabi joylashtiriladi.
	Измеритель: Tasvirda turli o'lchovlarni bajarish uchun ishlatiladi. Bu asbob bilan bir nuqtadan ikkinchi nuqtaga kursor olib borilishi kifoya. "Adobe PhotoShor" dasturi avtomatik tarzda ikki nuqta orasidagi masofani o'lchaydi.
	Градиент: Bu asbob ishlatilganda, tasvirdagi belgilangan maydonda ranglar kombinatsiyasi hosil bo'ladi. Asosiy rangning tasvir foniga sizib o'tish samarasi hosil bo'ladi.
	Заливка: Ushbu asbobdan asosan tavsvirni yoki tasvirdagi ajratib olingan hududni bo'yashda foydalaniladi. Ranglarni qo'shimcha buyruqlarni bajarish orqali tanlanadi. Bu asbobni faollashtirish uchun K tugmasi bosiladi.
	Пипетка: Tasvirdagi asosiy yoki tasvir foni rangini o'zgartiradi, Pipetkani tasvir ustidagi biror nuqtada bosish bilan o'sha nuqtadagi, ya'ni pikseldagi rang asosiy rang sifatida tanlanadi. Agar ayni jarayonga "Alt" tugmasini qo'shilsa, tanlangan rang tasvir fonini o'zgartirishiga olib keladi.
	Выборка цветов: Ushbu asbob yordamida tasvirdagi ranglar haqidagi axborot olishga xizmat qiladi. "Info" darchasida belgi qo'yilgan nuqtada necha foiz qizil, ko'k va qora ranglar mavjudligi haqidagi axborot hosil bo'ladi.
	Рука: Tasvirning ko'zga tashlanmay turgan qismlarini ko'rsatadi. Buning uchun ushbu asbob faollashtirilib tasvir ustida sichqonchani chap tugmachasini bosgan holda kerakli tomonga harakatlantiriladi. Ayni jarayonni "Adobe PhotoShop" dasturi darchasidagi "Navigator" yordamida ham amalga oshirish mumkin.
	Масштаб: Tasvirni kattalashtirish yoki kichraytirish uchun xizmat qiladi. Agar ushbu asbob bilan birgalikda "Alt" tugmasi ishlatilsa, tasvir kichrayadi. Kursor ushbu asbob faollashtirilgandan so'ng o'z shaklini o'zgartiradi va lupa ko'rinishini oladi. Kursor tasvirning qaysi nuqtasida bosilsa "Adobe PhotoShop" dasturi avtomatik tarzda o'sha nuqtani ekranga yaqinlashtiradi. Masshtab asbobini faollashtirib, Enter klavishasi bosilsa, Опция масштабирования darchasi ochiladi. Bu darchada maxsus to'rtburchak ichiga belgi qo'yilsa tasvir o'lchamlari o'zgartirilganda tasvir darchasi ham mos tarzda o'zgaradi. Har safar tasvir o'lchamlarini kattalashtirish yoki kichraytirish uchun Masshtab asbobini faollashtirish zarur emas. Boshqa asbob

	bilan ishlash paytida Ctrl Q Пробел bosilsa ishlatilayotgan asbob vaqtinchalik Masshtab asbobi vazifasini bajaradi va tasvir kattalashadi. Ctrl Q Пробел o'rnida AltQПробел qo'llanilsa, tasvir o'lchami kichrayadi. Shuningdek, tasvir o'lchamini Strl Q Q(плюс) klavishalari yordamida kattalashtirish yoki Ctrl Q - (минус) yordamida kichraytirish mumkin. "Adobe PhotoShop" dasturi darchasi ostida Строка состояния satrida tasvir o'lchamlari haqida axborot beruvchi maxsus darcha mavjud. Bu darchada tasvir o'lchamlari sonlarda ifodalangan. Ushbu sonlarni o'zgartirish orqali tasvirning aniq o'lchamlari kiritiladi.
	Основной цвет: Ushbu asbob ustida kursor ikki marta ketma-ket bosilganda "Adobe PhotoShop" dasturining yangi darchasi (Выбор цвета) hosil bo'ladi. Bu darchada kerakli rang tanlanib, OK yoki "Enter" tugmasi bosiladi va tanlangan rangni "Karandash", "Kist", "Aerograf", "Gradient" kabi asboblari yordamida qo'llash mumkin.
	Свет фона: Ushbu tugmacha ustida kursor ikki marta ketma-ket bosilganda "Adobe PhotoShop" dasturining yangi darchasi (Color Picker) hosil bo'ladi. Bu darchada tasvir fonining rangi aniqlanadi. Tavsvir fonidagi rang "Lastochka" va "Gradient" asboblari uchun qo'llaniladi.
	Переключение цветов: Ushbu belgi ustida kursorni bosish bilan asosiy rang bilan tasvir foni ranglari o'zni almashtiriladi.
	Цвета по умолчанию: Bu belgi ustida kursorni bir marta bosish bilan asosiy rang qoraga va tasvir foni ranglari oqqa aylanadi.
	"Марширующие муравьи": Bu tugmacha yordamida "Adobe PhotoShop" dasturida tez niqoblash holati bekor qilinadi. Ekranda belgilash chegaralari chumolilar harakatini eslatuvchi punktir chiziq yordamida aks ettiriladi. Bu "Adobe PhotoShop" dasturida odatda standart holat deb ataladi.
	Быстрая маска: Ushbu tugmacha ustida kursorni bir marta bosilishi bilan "Adobe PhotoShop" dasturi tez niqoblash holatiga o'tadi va natijada tasvirdagi niqoblanmagan hudud qizil rang bilan bo'yaladi. Ushbu asbob tasvirdagi turli obyektlarni aniq niqoblashda ishlatiladi. "Kist" asbobi yordamida niqobga ishlov berish mumkin. Bunda qora rang bilan tasvir niqoblanadi, oq rang bilan tavsvirdagi niqob o'chiriladi.
	Стандартное окно: Asbob faollashtirilganda tasvir standart holatda bo'ladi.
	Полный экран с tavsiyanoma: Bu holat tasvir kompyuter ekraniga sig'magan holda ishlatiladi. Ushbu asbob faollashtirilganda ekranda tavsiyanoma satri hamda asboblari paneli qoladi.
	Полный экран: Ekranda faqat tasvir va asboblari paneli hamda tavsiyanoma satri qora fonda qoladi.

Tasvir o'lchami

"Adobe PhotoShop" dasturida tasvir o'lchamlarini ekranda tasvirning barcha qismini yoki tasvirdagi kichik detallarni ko'rish uchun xohlagancha kattalashtirish

yoki kichraytirish mumkin. Ayni jarayon foizlar hisobida orttiriladi. Masalan, 100%- bu tasvirdagi piksellar soniga ekrandagi piksellar soni tengligini anglatadi. Ya'ni 1:1. 200% ga tasvir kattalashtirilganda ekrandagi bir pikselga katta miqdordagi piksellar miqdorini to'g'ri kelayotganligidan dalolat beradi.

Tasvirning haqiqiy o'lchami

"Adobe PhotoShop" dasturida 100% li o'lcham tasvirning haqiqiy o'lchami deyiladi. 100% li o'lchamda tasvir aniq va ravshan ko'rinadi.

Tasvirning haqiqiy o'lchami quyidagi amallarni bajarish orqali o'rnatiladi:

Tavsiyanomalar satrida VID tavsiyanomasida Реальный размер buyrug'ini tanlang.

Cntrl+ Alt+ 0 (nol, "O" harfi emas) tugmalarini birgalikda bosish orqali.

Asboblarning panelidagi Masshtab knopkasi ustida kursorni ikki marta ketma-ket bosish orqali.

To'la ekranli (полноэкранный) holat

"Adobe PhotoShop" dasturi tasvirni birinchi marta ochganda uni maksimal o'lchamda ochadi. Tasvir bilan ishlash jarayonida uning o'lchamini bir necha marta kattalashtirish yoki kichraytirishga to'g'ri keladi. Ana shunday holatlarda tasvirni xohlagan paytda dastlabki Полноэкранный holat holatiga qaytarish mumkin. Buning uchun quyidagi amallarni bajarish lozim:

Tavsiyanomalar satrida VID tavsiyanomasi tarkibidagi По размерам экрана buyrug'ini tanlang

Ctrl+0 (nol, "O" harfi emas) tugmalarini birgalikda bosish orqali

Asboblarning panelidagi Ruka knopkasi ustida kursorni ikki marta ketma-ket bosish orqali.

Tasvirning bosma shakldagi o'lchami (Размер печати)

Shuningdek, "Adobe PhotoShop" dasturi tasvirning printerda chop qilingandagi ko'rinishini chop qilmasdan avval ekranda ko'rish imkonini beradi. Buning uchun Tavsiyanomalar satrida Izobrajenie tavsiyanomasi tarkibidagi Размер изображение buyrug'ini tanlang. Ammo, hammo vaqt ham tasvirning ekrandagi ko'rinishi bilan chop etilgandagi o'lchamlari aynan mos tushavermaydi. Tasvir kattaliklari 0,2% dan 16 000% miqdor o'rtasidagi sonlar bilan belgilanadi.

"Adobe PhotoShop" dasturida Navigator darchasi bilan ishlash

"Adobe PhotoShop" dasturida tasvirdagi mayda detallar bilan ishlash jarayonida tasvirni bir necha marta kattalashtirishga to'g'ri keladi. Tasvirga kiritilgan o'zgartirishlar sifatli chiqishi uchun ayni amal bajariladi. Navigator darchasi (16.6.-

rasm) asosan tasvir o'lchamlarini o'zgartirish va tasvirni boshqarish uchun xizmat qiladi. Agar Navigator darchasi "Adobe PhotoShop" dasturi ishga tushirilgan chog'da ekranda mavjud bo'lmasa, uni faollashtirish uchun tavsiyanomalar satrida Okno tavsiyanomasidagi Показать навигатор buyrug'ini tanlang.



6- rasm

"Adobe PhotoShop" dasturida Action darchasi bilan ishlash

Action darchasi "Adobe PhotoShop" dasturida ishlashni yanada tezlashtiradi va bir necha tasvir ustida amalga oshiriladigan bir xil amallarni har safar tarkorlashga zaruriyat qoldirmaydi. "Adobe PhotoShop" dasturidagi Action darchasi bilan ishlashni bilsangiz, qisqa fursat ichida ko'p miqdordagi tasvirni tahrirlashingiz mumkin. Buning uchun Action darchasida yangi Action ochiladi. Uni kerakli nom bilan nomlagandan so'ng Resord tugmachasi bosiladi. Shu daqiqadan boshlab "Adobe PhotoShop" dasturi sizni tasvir ustida bajargan barcha amallarinigizni kompyuter xotirasiga ketma-ket joylashtiradi. Tasvir ustida barcha amallar yakunlangandan so'ng Action darchasidagi Stop tugmachasi bosiladi. "Adobe PhotoShop" dasturi sizning barcha amallaringizni tartibli ravishda Action darchasida joylashtiradi. Boshqa tasvirlarga ushbu amallarni qo'llash uchun yangi tasvir ochilgandan so'ng Action darchasidagi Выполнение buyrug'ini ishga tushirish lozim. Adobe PhotoShop dasturi avtomatnk tarzda yangi ochilgan tasvirda ham siz amalga oshirgan amallarni hech bir o'zgarishlarsiz bajaradi.

Yangi tasvir, dublikat ochish va tasvirni doimiy xotiraga joylashtirish

"Adobe PhotoShop" dasturida ishlashdan avval yangi fayl tuziladi yoki kompyuter xotirasida mavjud bo'lgan tasvir ochiladi. Yangi fayl tuzish va avvaldan mavjud bo'lgan fayllarni ochishning quyidagi yo'llari mavjud:

Fayl — Новый. Tavsiyanoma satrida Fayl tavsiyanomasini ochib Новый buyrug'ini tanlang yoki Ctrl + N tugmalari kombinatsiyasidan foydalanib, yangi fayl tuzing. Yuqoridagi amal bajarilganidan so'ng "Adobe PhotoShop" dasturi yangi faylni tashkil etish uchun uning o'lchovlari haqidagi ma'lumotlar bitilgan yangi darchani hosil qiladi. Bunda yangi tuzilayotgan fayl nomi, uning o'lchamlari kiritilishi talab etiladi. Kerakli o'lchamlar kiritilganidan so'ng, OK tugmasini bosish lozim. Adobe PhotoShop dasturi oq rangdagi yangi tasvirni tuzadi. Bu tasvirga xohlagan o'zgartirish kiritish yoki boshqa tasvirlardan ayrim detallarni ko'chirib o'tib, umuman yangi tasvirni ijod qilish mumkin.

Fayl — Открыть: Tavsiyanoma satrida Fayl tavsiyanomasini ochib Открыть buyrug'ini tanlang yoki Ctrl + O tugmalari kombinatsiyasidan foydalanib kompyuter xotirasida mavjud bo'lgan faylni oching. Juda kam hollarda Fayl — Открыть как (Ctrl+ Alt+ O) buyrug'i tanlanadi.

Tasvir Formatlari

"Adobe PhotoShop" dasturi 20 dan ortiq formatdagi fayllar bilan ishlash imkoniga ega. Eng ko'p qo'llaniladigan formatlar:

BMP (Windows Bitmap — Битовая карта Windows) Windows muhitida ishlovchi kompyuterlarda ekran osti tasvirlarini qo'llovchi dastur Microsoft Paint da keng qo'llaniladi.

JPEG (Joint Phonographic Experts Group) Hozirgi kunda eng ko'p qo'llaniladigan formatlardan biri bo'lib, uning asosiy afzalliklaridan biri maxsus dastur yordamida yetarlicha siqish imkonining mavjudligidir. Ammo faylni siqib hajmini kichraytirish jarayonida tasvir sifatida o'zgarish bo'ladi. Fayl kuchli siqilganda tasvir sifati yomonlashishi mumkin. Ushbu formatdagi fayllar kompyuter xotirasida ko'p joy egallamaydi va hajm jihatidan kichikligi bois mazkur formatdagi tasvirlar bilan ishlash ancha oson.

TIFF (Tagged Image File Format) bu formatdagi fayllar ham keng qo'llaniladi. Lekin TIFF formatidagi fayllar kompyuter xotirasida ko'p joyni egallaydi. "Adobe PhotoShop" dasturida ushbu formatdagi tasvirlar bilan ishlashda dasturning ishlash tezligi sezilarli ravishda kamayishi mumkin.

GIF (Graphics Interchange Format — grafik almashish formati) Ushbu formatdagi tasvirlar 256 turdagi rang bilan tasvirlanadi. Bu formatdagi tasvirlar asosan Internet tizimida keng qo'llaniladi.

"Adobe PhotoShop" dasturida ranglar bilan ishlash

"Adobe PhotoShop" dasturi asboblari panelida ranglar bilan ishlash uchun to'rtta asbob ajratilgan.

Основной свет. Ushbu asbobda qanday rang ko'rsatilgan bo'lsa "Kovsh", "Liniya", "Karandash", "Kist", "Aerograf" va shuningdek, Alt tugmasi bilan birgalikda qo'llanganda "Pales" asboblari uchun o'sha rang asosiy hisoblanadi. Основной свет asbobidagi rang "Pipetka" yoki ushbu asbob ustida kursorni ikki marta ketma-ket bosish orqali o'zgartiriladi.

Свет фона Ko'rsatilgan rang "Lastik" asbobi bilan ishlaganda qo'llaniladi. Свет фона asbobidagi rang Osnovnoy svet asbobida rangni o'zgartirish uchun qanday amal bajarilgan bo'lsa bunda ham huddi o'sha amalga rioya qilish lozim. Yoki "Pipetka " asbobi bilan Alt tugmasi birgalikda bosish orqali almashtirish mumkin.

Переключение светов. Kursorni ushbu tugmacha ustida bir marta bosish orqali asosiy rang va fon ranggi o'rin almashadi.

Стандартный свет. Kursorni ushbu tugmacha ustida bir marta bosish orqali asosiy rang va fon rangi standart ranglar- qora va oq rangga almashtiradi.

Ranglarni tanlashda "Adobe PhotoShop" dasturida Color darchasidan ham foydalanish mumkin (7-rasm).

RGB — moduli tasvirni ekranda tahrirlash nuqtai nazaridan kelib chiqqan holda juda qulay va u 24 razryadli ranglar platasi yordamida deyarli barcha 16 million ranglarni monitorda aks ettiradi. RGV ranglar majmuasi bilan ishlangan barcha tasvirlarni xohlagan formatda diskka yozish mumkin. RGB — ranglar majmuasidagi ayrim ranglar umuman tabiatda uchramaydi.

SMYK-Tabiatda mavjud bo'lgan ranglar majmuasi. quyosh nurlari inson ko'zlari ajrata oladigan barcha ranglarni o'zida mujassamlashtirgan. Quyosh nurlari biror — bir jismga tushganda uning tasiri ostida inson ko'zlari jism shakli va rangini idrok etadi. Misol uchun binolarning o't o'chirish burchaklariga osib qo'yilgan o't o'chirgichlar to'q ko'k va zangor ranglar bilan bo'yalgan bo'lishiga qaramay bizning ko'zimizga to'q qizil rangda ko'rinadi. Ranglarni bir — biriga qo'shilishi natijasi boshqa ranglar hosil qilinadi:

S — havo rang,

M - binafsha rang ,

Y -sariq rang ,

K- qora rang ,

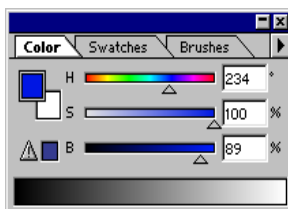
Bosma mashinalari va bosmaxonalarda tasvirlar yuqorida keltirilgan ranglarning kombinatsiyasidan foydalangan holda to'la tasvirni ifodalaydi.

RGB — ranglar majmuasida oq rang uchchala ranglarning maksimal aralashmasidan hosil qilinadi. Qora rang esa buning aksi o'laroq olinadi.

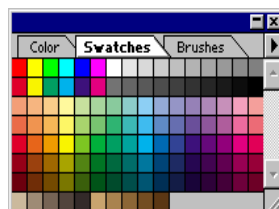
CMYK — ranglar majmuasi bilak ishlaganda qopa va oq ranglarni hosil qilish uchun buning aksini bajarish lozim. Ya'ni to'rt rangning minimal qo'shilishida oq rang hosil bo'ladi. Qora rang esa alohida kanalda mavjud.

RGB — ranglar majmui keng ko'lamdagi ranglarni taklif etadi. Lekin ularning ko'p qismi (ayniqsa yorqinlari) tasvirni chop etganda monitordagi bilan keskin fapq qiladi. Shu bois ham ko'plab mutaxassislar tasvirni chop etishdan avval uni CMYK— tizimiga o'tkazadilar. Ayrim mutaxassislar esa tasvir bilan CMYK — tizimida ishlashni maslahat beradilar. Ammo bu tasvir bilan ishlash turli qiyinchiliklarni tug'diradi. Ana shunday qiyinchiliklardan biri kompyuter juda sekin ishlaydi. Bunga asosiy sabab "Adobe PhotoShop" dasturi RGB -tizimiga sozlangan

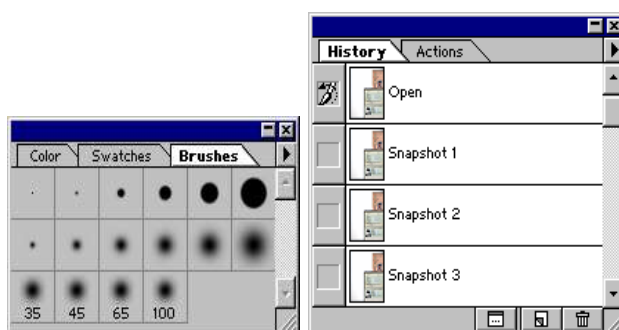
bo'lib, har bir buyruqni bajarib, uni RGB — tizimidan CMYK — tizimiga almashtirguncha kompyuter qo'shimcha vaqt talab qiladi. Bundan tashqari skaner va monitor RGB— tizimida ishlashga mo'ljallangan. Ranglar bilan ishlovchi barcha uskunalar (rangli printerdan tashqari) RGB -tizimida ishlaydi. Shuning uchun yaxshisi tasvirni chop etishdan avval CMYK — tizimiga o'tkazib olish maqsadga muvofiq. Tasvir ustida barcha



7-Rasm



8 –Pacm



9-rasm

amallarni poyoniga etqazib, tavsiyanomalar satrida ИЗОБРАЖЕНИЕ - Настройка tarkibidagi SMUK buyrug'ini tanlang.

"Adobe PhotoShop" dasturida History darchasi bilan ishlash

"Adobe PhotoShop" dasturi ishga tushirilganda ekranda History darchasi mavjud bo'lmasa Okno tavsiyanomasida Pokazat History buyrug'ini tanlang (9.-rasm). History darchasida tasvirga kiritilgan so'nggi o'zgartirishlar haqidagi ma'lumotlar joylashadi. Tasvir ustida bajarilgan har bir muhim amallar History darchasida ro'yxatga olinadi. Xohlagan paytda yoqmagan amallarni History darchasi orqali rad etish mumkin. Buning uchun kursor bajarilgan so'nggi amaldan oldingi buyruq ustida bosiladi yoki "Ctrl" + "Alt" + "7," klavishalari bosiladi. "Adobe PhotoShop" dasturi bajarilgan so'nggi 20 amalni ro'yxatga oladi. yigirma birinchi amal bajarilganda "Adobe PhotoShop" dasturi avvalgi amallarni o'chiradi.

"Adobe PhotoShop" dasturida qatlamlar bilan ishlash

"Adobe PhotoShop" dasturi tavsvirdagi biror obyekt "Прямоугольная область ", "Эллиптическая область", "Лассо", "Волшебная палочка ", "Быстрая маска" yordamida tasvirdagi detallar belgilanib ularning nusxalari olinganda "Adobe PhotoShop" dasturi yangi qatlam hosil qiladi. Bu qatlam alohida obyekt bo'lib, uni tahrirlash ham alohida tarzda bo'ladi. Bir necha tasvirlardagi ayrim detallarni

yagona tasvirga jamlanganda "Adobe PhotoShop" dasturi ko'chirib o'tilgan tasvir bo'laklarni qatlamlarga ajratadi. Ayni jarayon tasvir detallarini joylashtirish uchun juda qulay. Bir-necha qatlamlar bilan ishlaganda ularni boshqarish qiyinlashadi. "Adobe PhotoShop" dasturi bajarilganda yangi darcha hosil bo'ladi. Bu darchada tasvirdagi yangi qatlamlar haqidagi axborotlar joylashadi. Yangi qatlam hosil qilinganda "Adobe PhotoShop" dasturi uni "Слой" darchasida ro'yxatga oladi. Bu darchada qatlamlar tartibli tarzda joylashtiriladi. Shuningdek qatlamlarning o'rnini almashtirish yoki vaqtinchalik o'chirib qo'yish mumkin. Qatlamlarni o'chirish uchun avval kerakli qatlam "Слой" darchasida belgilab olinadi va tavsiyanomalar satrida Слой tavsiyanomasi tarkibidagi Удалить слой buyrug'i tanlanadi. Qatlamlarni bir-biriga birlashtirish imkoniyati ham mavjud.

"Adobe PhotoShop" dasturida matnlar bilan ishlash

"Adobe PhotoShop" dasturi tasvirlar ustiga matnlarni kiritish uchun yana bir keng imkoniyatni ochib beradi. Tasvir ustiga matn kiritish uchun asboblari panelida maxsus "Текст" asbobi mavjud. Bu asbob asosan tasvirga turli matnlarni kiritish uchun xizmat qiladi. "Текст" asbobi tarkibida "Текст-маска", "Вертикальный текст" kabi asboblari yashiringan. Bu asboblari faollashtirilib, tasvir ustida bir marta bosilishi bilan yangi "Текстовый инструмент" darchasi hosil bo'ladi. Bu asboblari faqat matnlarni tuzush uchun xizmat qiladi. Uning yordamida mavjud matnlarni tahrirlashning iloji yo'q.

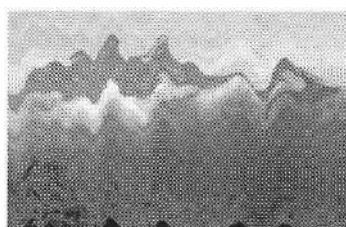
"Adobe PhotoShop" dasturida filtrlar bilan ishlash

"Adobe PhotoShop" dasturida tasvirni qo'shimcha samaralar bilan boyitish uchun filtrlardan foydalanish mumkin. Buning uchun tavsiyanomalar satrida Фильтры buyrug'i tarkibidagi filtrlardan foydalanishingiz mumkin. Quyida eng ko'p qo'llaniladigan filtrlar tavsifi misollar yordamida keltiriladi.

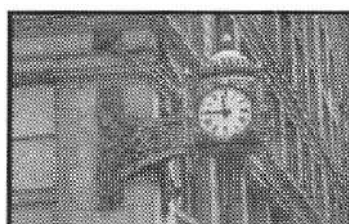
Chap ustunda filtri qo'llanilmagan asl nusxa, o'ng tomondagi ustundagi tasvirlar filtri qo'llanilgandan so'nggi varianti keltirilgan.



Asl nusxa



Filter Iskajen - Volna



CorelDraw grafik muharriri

CorelDraw vektorli grafikaning Windows amaliyot tizimida ishlaydigan tahrirlovchi dasturlardir. Uning yordamida turli grafik ko'rinishlarni loyihalash, foto, matn, tasvirlar ustida, ayniqsa badiiy ko'rinishdagi kompozitsiyalarni tahrirlash bilan bog'liq amallarni bajarish mumkin. CorelDraw muharirini ishga tushirish uchun CorelDraw ning belgisiga borib sichqonchani yoki Enter tugmasini bosish kerak

CorelDraw boshqa grafik muharrirlarga nisbatan matnlar bilan yaxshi ishlaydi, ya'ni nashriyot tizimlarida, masalan, yumoristik yoki boshqacha kitoblarni har xil turli o'lchamdagi harflar bilan yozish mumkin. Siz o'zingizning pasportdagi rasmingizni skanerdan o'tkazib va CorelDraw yordamida shu rasmingizni chiroyli portretga aylantirishingiz mumkin. O'z-o'zidan ma'lumki bunday professional murakkab grafik muharrirlarda, ya'ni CorelDraw ning boy asboblari va ranglaridan foydalanib biz o'zimiz rasm chizishimiz mumkin. CorelDraw muharirida fayllarning kengaytmasi file.cdr ko'rinishda bo'ladi. Fayllarni import va eksport qilish eng yaxshi qo'layliklaridan biri hisoblanadi. Ayniqsa, kompakt diskdan fayllarni import qilishda ko'p qo'llaniladi.

Fraktal grafika ham hisoblanuvchi grafika bo'lib, uni vektor grafikadan farqi qanday obyektlar kompyuter xotirasida saqlanmaydi. Chunki tasvirlar yoki ularni sistemalari hosil qilinadi. Shuning uchun ham xotirada bunday tenglamalargina saqlanadi.

Tenglamalarga oid parametrlar o'zgartirib tasvirlar hosil qilinadi.

Fraktal grafika matematik hisoblashlar asosida tasvirlarni avtomatik yaratish uchun qo'llaniladi. Shuning uchun ham uning asosini rassom, shakl, tasvir hosil qilishning dasturlash usuli tanlangan.

Bu grafika odatda turli jarayonlarni modellashtirish, tahlil qilish, turli qiziqtiruvchi dasturlar yaratishda ko'proq qo'llaniladi.

Macromedia Freehand 8.0 vektorli muharriri

Bu juda qulay bo'lgar vektorli muharrir, do'stona va juda tushunarlilik uchun boshlovchilarga juda omadli tanlangan bo'ladi. Bu dastur boshqarish tizimining soddaligi va harakatlanish tezligining yuqori darajasi bilan ajralib turadi. Uning yordamida shunday kompyuterlarda ishlash mumkinki boshqa vositalari ishlatilganda qiyin jarayonga aylanib ketmaydi. Boshqarish tizimining soddaligiga qaramay, Macromedia Freehand instrumental vositasi murakkab hujjatlarni ishlab chiqarish uchun yetarli. Lekin Adobe Illustrator va-Corel Draw vositalari uchun murakkabligi o'ta baland darajada bo'lsa yo'l bo'shatadi. U maxsus Quark XPress

tizimi uchun mo'ljallangan, ammo Macromedia Freehand dasturini kompyuter tizimining hamma ko'rinishlarida ishlarida ishlatish juda qulay.

Macromedia Freehand (lahja 8.0) va Corel Draw (5,0 dan 8.0 gacha bo'lgan lahjalar ishlatiladi) muharrirlarning hammasi vektorli grafikning birdan-bir obyekti ya'ni birdan-bir prinsiplarga asoslangan, o'xshash asboblarga ega, shu bilan birga vektorli tasvirlarni tuzish yo'llari bu muharrirlarda juda o'xshashdir.

Adobe Illustrator 7.0 vektorli muharriri

Bu dastur dunyo bo'yicha tan olingan vektorli grafiklar ichidagi yetakchisi hisoblanadi. Ko'pchilik professionallar ana shu dastur bilan ishlaydi. Uning asosiy qudrati shundaki, u Adobe Photoshop va Page Maker dasturlari bilan tugallangan muqaddimada ko'rsatiladi, ya'ni poligrafik bosmalarning muqovasini kompyuterda bajarish va murakkab hujjatlarni ishlab chiqarish uchun yaratilgandir. U turli obyektlarni xatosiz eksport va import qiladi. Adobe Illustrator 7.0 qo'shimcha imkoniyati shundan iboratki, bu vektorli muharrirning rus tilidagi talqini ham mavjud.

Nazorat savollari:

1. Kompyuter grafikasi deganda nimani tushunasiz?
2. Kompyuter grafikasi qanday turlarga bo'linadi?
3. Rastri grafika frontal grafikadan nimasi bilan farq qiladi?
4. Adobe Photoshop rasm taxirchisi nima uchun mo'ljallangan?
5. Adobe Photoshop dasturini ishga tayyorlash tartibi qanday?
6. Adobe Photoshopning tavsiyanomasi qanday bo'limlardan iborat?
7. Adobe Photoshopning asboblari paneli haqida tushuncha bering?
8. Tasvir o'lchami qanday o'rnatiladi?

Ma'ruza № 6

Maruza: Geoinformatsion tizimlar

Reja:

1. Geoinformatsion tizimlar va ularning vazifasi
2. Geoinformatsion texnologiyalar
3. ArcGIS bilan konturli chiziqlar yaratish
4. ArcGIS Pro-ni o'rnatish jarayoni haqida

Geoinformatsion texnologiyalarning jadal rivojlanishi va ularni qo'llashning keng doirasi, shuningdek, geoinformatika rivojlanishining nisbatan qisqa davri tufayli turli mualliflar GISga har xil ta'riflarni berishadi.

GIS - bu fazoviy muvofiqlashtirilgan ma'lumotlarni yig'ish, saqlash, qayta ishlash, kirish, namoyish qilish va tarqatishni ta'minlaydigan axborot tizimi.

Tabiat va jamiyat o'rtasidagi o'zaro ta'sirning hududiy tomoni to'g'risidagi bilimlar tizimini saqlashning avtomatik vositalari, shuningdek qidirish, kiritish, modellashtirish va hk funktsiyalaridan foydalanadigan dasturiy ta'minot yordamida amalga oshirilgan GIS ta'rifi mavjud.

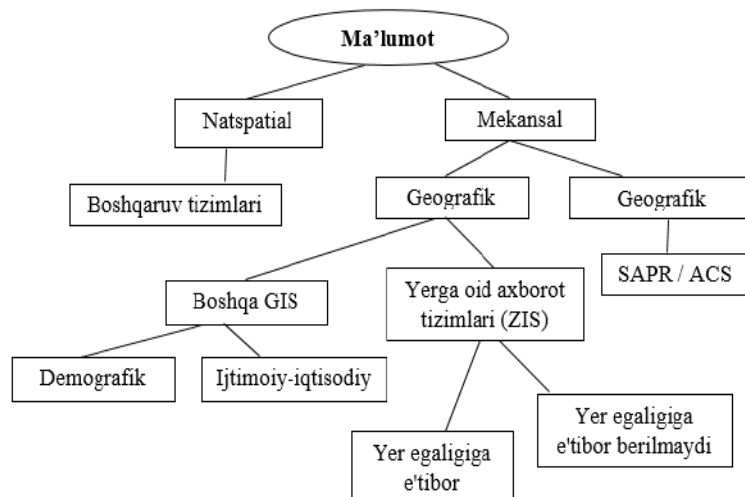
GISning eng to'liq ta'riflaridan biri klassik geoinformatika tadqiqotchilari tomonidan berilgan. Geografik axborot tizimi deganda ular fazoviy ma'lumotlarni yig'ish, qayta ishlash, namoyish qilish va tarqatish, ma'lumotlar, ma'lumotlar va bilimlarni inventarizatsiya qilish, tahlil qilish, modellashtirish, prognozlash bilan bog'liq ilmiy va amaliy muammolarni hal etishda ulardan samarali foydalanish uchun birlashtiradigan apparatli dasturlashtirilgan odam-mashinalar majmuasini anglatadi. atrof-muhitni boshqarish va jamiyatning hududiy tashkil etilishi.

Qisqartirish va imkoniyatlar nuqtai nazaridan GISning eng muvaffaqiyatli ta'rifi R.Bil va D.Frits (1991) tomonidan berilgan bo'lib, u asos bo'lib olingan: geografik axborot tizimi bu kirish, boshqarish, er yuzidagi ob'ektlar haqidagi fazoviy va atributiv (ramziy) ma'lumotlarni tahlil qilish va taqdim etish. Ushbu ta'rif nafaqat GISning tuzilishini, balki u bajaradigan funktsiyalarni ham aks ettiradi, garchi u ma'lumot yer yuzasi sohalari bilan bog'liqligini ko'rsatmasa ham. Bunday ta'rifni fazoviy ma'lumotlarni o'z ichiga olgan har qanday fazoviy axborot tizimiga berish mumkin. Shuning uchun geografik axborot tizimlarini bir qator fazoviy axborot tizimlaridan ajratib olish uchun geografik axborot tizimida (Geo land) yer maydonlari haqidagi fazoviy ma'lumotlarni o'z ichiga olganligini aniqlash maqsadga muvofiqdir. Axborot tizimlarining tasnifi va unda GIS pozitsiyasi M.DeMers tomonidan aniq ko'rsatilgan (1-rasm).

Ushbu tasnif kosmik va nospatal axborot tizimlari bilan GIS o'rtasidagi ajratishni ko'rsatadi, bu holda fazoviy axborot tizimlari toifasiga kiradi.

Shuningdek, fazoviy axborot tizimlarining ikkita umumiy klassi mavjud: geografik va geografik bo'lmagan. Ikkinchisi, ular kosmosning ba'zi qismlari bilan shug'ullansa ham, odatda er yuzasi va geodezik koordinatalari bilan bog'liq emas.

Kompyuter chizish tizimlari kabi tizimlarning sinflari geografik bo'lmagan fazoviy axborot tizimlari deb tasniflanadi.



Rasm-6.1. Axborot tizimlarining tasnifi

GISni quruqlik va boshqa axborot tizimlariga bo'lish mumkin. Ushbu ajratish asosan sun'iy bo'lsa-da, ba'zida foydalidir, chunki u GISni qo'llash sohaslarini chegaralaydi. Boshqa ma'lumotlar tizimlarining namunalari demografik bo'lib, ularning asosiy ob'ektlari bu ob'ektlar joylashgan hudud emas, balki aholi, uy-joy qurilishi, iqtisodiy faoliyatdir.

Geografik axborot tizimlarining tasnifi va tuzilishi

GISning har xil turlari bilan ular quyidagi mezonlarga muvofiq tasniflanadi:

- **muammoli** - tematik yo'nalish bo'yicha - qo'llanilish sohasiga qarab;
- **hududiy qamrov bo'yicha** - GIS ma'lumotlar bazasini tashkil etuvchi raqamli kartografik ma'lumotlarning masshtabli seriyasiga qarab;
- **geografik ma'lumotlarni tashkil qilish usuli bilan** - kartografik ma'lumotlarni kiritish, saqlash, qayta ishlash va taqdim etish formatlariga qarab.

Geoinformatsion tizimlarni maqsadga muvofiq tasniflash ularni foydalanish predmeti sohasiga va echilayotgan vazifalarning mohiyatiga muvofiq quyidagi asosiy guruhlariga bo'linishni nazarda tutadi:

- **ko'p maqsadli** - fazoviy ma'lumotlarni qayta ishlash bilan bog'liq turli sohalarda foydalanish uchun mo'ljallangan. Bunday tizimlar asosiy dasturiy ta'minot funktsiyalarining ko'p qirraliligi, shuningdek, GISni ma'lum bir foydalanish sohasidagi muammolarni hal qilishga moslashtirish uchun maxsus dasturlar uchun ichki rivojlanish muhiti mavjudligi bilan ajralib turadi;

- **ma'lumot va ma'lumotnoma** - GIS, uning asosiy maqsadi - bu atribut ma'lumotlari to'plami bilan kartografik ob'ektlarni qidirish;

- **monitoring va inventarizatsiya** - fazoviy ob'ektlarning xususiyatlari va joylashishidagi o'zgarishlarni hisobga olish, ularni tahlil qilish va tegishli hujjatlarni tayyorlash uchun mo'ljallangan;

- **tadqiqot** - turli sohalarda tadqiqotlar o'tkazish uchun mo'ljallangan. Ular yuqori darajada ixtisoslashgan ma'lumotlarni tahlil qilish funktsiyalaridan foydalanish bilan ajralib turadi;

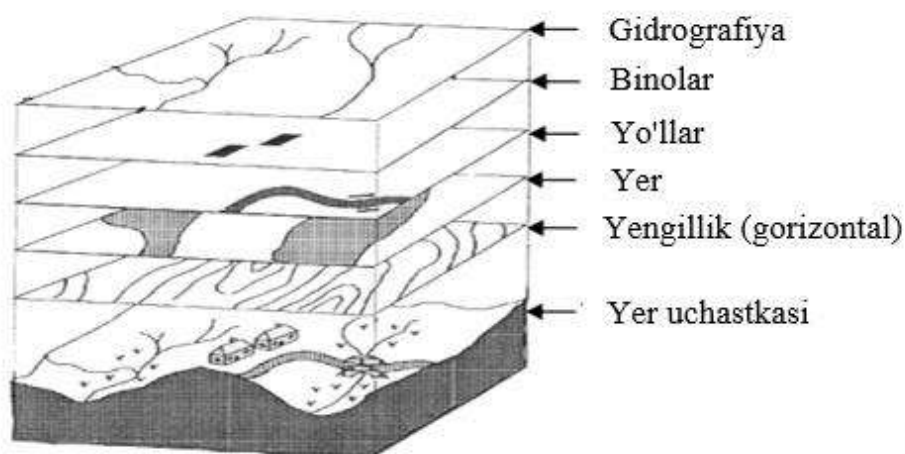
- **nashriyot** - rejalashtirish va xaritalash materiallarini loyihalash, kompilyatsiya qilish, tahrirlash, loyihalash va bosib chiqarish uchun mo'ljallangan avtomatlashtirilgan xaritalash tizimlari.

Geografik axborot tizimlarining ma'lumotlarning fazoviy modellari

Geografik axborot tizimlarida ma'lumotlar odatda ikki guruhga bo'linadi: fazoviy (geometrik) va tematik yoki atributli ma'lumotlar.

Ma'lumotlarni fazoviy tashkil etishning asosiy tamoyillari GISda fazoviy ob'ektlar vakolatxonasini belgilaydigan qatlamma-qatlam tashkil etish printsiplari (klassik) va ob'ektga yo'naltirilgan yondashuvdir. **Mekansal** ob'ekt (geografik ob'ekt, geo ob'ekti) - bu joylashgan joy, xususiyatlar va atributlar to'plami to'g'risidagi ma'lumotlarni o'z ichiga olgan mahalliy ob'ekt yoki hodisaning raqamli modeli.

Ma'lumotlarni qatlamma-qatlam tashkil etish printsiplari hududning fazoviy ob'ektlarini kartografik qatlamlarning ma'lum ro'yxatiga bo'lishidir.



Rasm-6.2. Fazoviy ma'lumotlarni qatlamlash printsiplari

Kartografik qatlam - bu ma'lum bir hududdagi bir xil turdagi bir xil turdagi ob'ektlarning to'liq to'plamidir.

Ma'lumotlarni geografik axborot tizimlariga kiritish

Har qanday geografik axborot tizimining amaliy ahamiyati va funktsionalligi, asosan, uning tarkibidagi fazoviy va atribut ma'lumotlarining dolzarbligi, to'liqligi va to'g'riligi bilan belgilanadi. Shu munosabat bilan GIS ma'lumotlarini zamonaviy holatga kiritish va saqlash jarayoni umuman geografik axborot tizimining ishlash samaradorligini ta'minlash nuqtai nazaridan juda muhimdir.

Atribut ma'lumotlarini GISga kiritish, qoida tariqasida, klaviaturani kiritish yoki tashqi ma'lumotlar bazalaridan GISda foydalanish uchun zarur bo'lgan tuzilma va formatga o'tkazishni o'z ichiga oladi.

GISga fazoviy ma'lumotlarni kiritish texnologiyalari, avvalambor, ularning manbalari va kerakli ma'lumotlar modeliga bog'liq.

Analog (qog'ozli) xaritadan GISga raster fazoviy ma'lumotlarni kiritish skanerlash yordamida amalga oshiriladi.

Barcha zamonaviy xilma-xillik bilan skanerlarning to'rt turi mavjud: qo'lda, tekis, bruska va baraban.

Qo'lda skaner - bu yorug'likka sezgir elementlarning chizig'i va bitta korpusga joylashtirilgan yorug'lik manbai. Skanerlash mexanizmini asl nusxada harakatlantirish qo'lda amalga oshiriladi. Agar skaner qilingan sahifa skanerlash mexanizmining kattaligidan oshib ketsa, jarayon bir necha bosqichda amalga oshiriladi, so'ngra hosil bo'lgan rasm chiziqlarini dasturiy ta'minot bilan moslashtirish amalga oshiriladi. Skanerlash jarayonida olingan tasvirning aniqligi pastligi sababli reja-kartografik materiallarni skanerlashda qo'l skanerlaridan foydalanilmaydi.

Baraban skanerlarida skanerlangan asl nusxasi shaffof tsilindr yuzasiga o'rnatiladi, u yuqori tezlikda aylanadi (1000 min / min atrofida). Tekshirish ob'ektivni aylanuvchi baraban bo'ylab harakatlantirish orqali amalga oshiriladi.

Barabanning nuqta maydoniga to'plangan yorug'lik oqimi halogen yorug'lik manbai yordamida shovqin ta'sirini yo'q qiladi va skanerlangan asl nusxaning butun spektrini yuqori sifatli qayta ishlaydi. Baraban skanerlari eng yuqori sifatli tasvirlarni ishlab chiqaradi.



Skaner turlari:

- a) planshet;
- b) uzoq muddatli;
- c) baraban ichra.

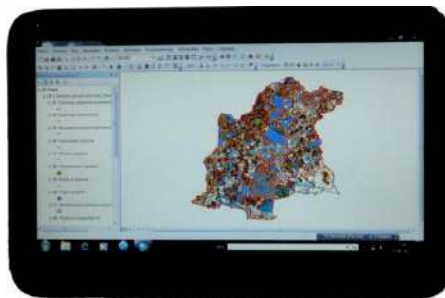
Hozirda GIS mobil dasturlarining juda katta ro'yxati mavjud. Mahalliy amaliyotda tez-tez ishlatiladiganlar qatoriga ESRI (ArcGIS Mobile, ArcPad), shuningdek, ixtisoslashgan mobil GIS Formap2.0 ishlanmalari kiradi.

ArcGIS Mobile - bu server bilan sinxronlashtirilishi mumkin bo'lgan, bir martalik, moslashtirilgan mobil dasturlarni ishlab chiqish va tarqatish uchun ArcGIS Server tarkibiga kiruvchi dasturiy ta'minotni ishlab chiqish uchun vositalar to'plami. Shunday qilib, ilova mijoz-server texnologiyasi asosida ishlaydi va ma'lumotlar bazalarining aniqligi va dolzarbligini ta'minlagan holda, serverdagi GIS ma'lumotlarini interaktiv ravishda ishlatish va yangilash imkonini beradi. Dasturiy ta'minotning asosiy funktsiyalari qatorida ma'lumotlarga fazoviy so'rovlarni

bajarish, GPS ma'lumotlaridan foydalanish, fazoviy ob'ektlarni yaratish va tahrirlash ma'lumotlari xususiyati ta'kidlanishi kerak.

ArcPad - ma'lumotlar bazasiga kirishni ta'minlash, xususiyatlarni yaratish va tahrirlash va ma'lumotlarning atributlarini yaratish, shuningdek mobil qurilmalar yordamida global joylashishni aniqlash tizimlari bilan birlashtirilishi mumkin bo'lgan xaritalash va maydon ma'lumotlarini yig'ish uchun mo'ljallangan mobil dastur.

Shuni ta'kidlash kerakki, ish stoli GIS dasturiy ta'minoti, shuningdek, aksariyat hollarda operatsion tizimga muvofiqligi sharti bilan mobil qurilmalarda o'rnatilishi mumkin.



Rasm-6.3. "Braslav ko'llari" GIS NP ning mobil versiyasi

Mobil Gap Formap2.0 ish stoli GIS dasturiy mahsulotining "O'rmon resurslari" ni ishlab chiqish natijasidir va o'rmon erlarining raqamli xaritalari va ularning atribut ma'lumotlari bilan ishlashga eng moslangan. Tizim Android platformasida ishlaydi va har qanday qurilmaga, shu jumladan aqlli telefonlarga o'rnatilishi mumkin (3-rasm).

Formap 2.0 mobil GIS-ning asosiy funktsional imkoniyatlariga quyidagilar kiradi: kartografik qatlamlarni boshqarish va xarita ob'ektlarining atribut ma'lumotlarini ko'rish, interaktiv kartografik Internet-xizmatlaridan (Google Maps, Yandex Maps, Rosreestr va boshqalar) fazoviy ma'lumotlarni yuklash, relef ob'ektlari bo'yicha GPS tadqiqotlarini o'tkazish va hk.



GIS Formap2.0 ning mobil versiyasi

Hozirgi vaqtda yopiq (hujjatsiz) fazoviy ma'lumotlarning formatlari (Garmin, Navitel) bo'lgan avtomobil va turistik mobil tizimlar ham keng qo'llanilmoqda. Ushbu tizimlar ma'lumot va ma'lumotnoma maqsadiga ega, ular foydalanuvchining raqamli xaritada joylashishini aniqlash, atribut xususiyatlarini ko'rish va ob'ektlarni topish va uchastka yo'nalishlarini topish uchun eng oddiy so'rovlarni bajarishga imkon beradi. Tizimlar ma'lumotlarni yaratishga va tahrirlashga yo'l qo'ymaydi va

ma'lumotlarning maxsus formati foydalanuvchi xaritalarini mobil qurilmaga to'g'ridan-to'g'ri yuklab olishning iloji yo'q.

Umuman olganda, ekologik turizm sohasida mobil geografik axborot tizimlari sayyohlik marshrutlarini loyihalash, relef nuqtalariga navigatsiya qilish, ekologik turizm ob'ektlarini rejalashtirish va kartografik materiallarda namoyish etish uchun global joylashishni aniqlash tizimlari texnologiyalari bilan birgalikda qo'llaniladi. Shu bilan birga, global joylashishni aniqlash tizimlaridan foydalanib, relyef nuqtalarining koordinatalari olinadi va mobil GIS dasturiy ta'minoti foydalanuvchi buyruqlariga muvofiq olingan ma'lumotlarni qayta ishlaydi.

TOPSHIRIQLAR:

1. Geoinformatsion tizim deganda nimani tushunamiz?
2. Tarixiy tadqiqotlarda GIS lardn qanday foydalaniladi?
3. Qanday GIS larni bilasiz va ular qaysi sohalarda qo'llaniladi?

Ma'ruza № 7

Mavzu: Tarmoq ma'lumot bazasi. Bulutli texnologiyalar.

Reja:

1. Internet tarixi
2. Kompyuterdan foydalanish modellari
3. Lokal kompyuter tarmog'i
4. Global kompyuter tarmog'i
5. Bulutli texnologiyalar.

Internet tarixi

Internet bu XX asrda kashf etilgan telekommunitsion va kompyuter tarmoqlar majmuidir. Uning tarixi 1960 yillaridagi Karib majorasidan sung, AQSHning ilmiy markazlaridan biri bo'lgan **RAND CORPARATION** korxonasi birinchi marta butun mamlakatni qamrab oladigan markazlashmangan kompyuter tarmog'ini yaratishni taqlif qilgandan boshlanadi. Bu loyihani amalga oshirishdan Maqsad harbiy muassasalar, ilmiy va o'quv markazlari kompyuterlarni bir tarmoqqa birlashtirib, boshqarishni markazlashtirish edi. Maqsad yadro quroli xujumiga ham, tarmoqning bir necha qismi ishdan chikkan holda ham ishlash faoliyatini saqlab qoladigan tizimni yaratish edi. Bunday tizimni tarmoqlar soni ko'p bo'lgandagina amalga oshirish mumkin edi. Shunday qilib Internetga asos solindi.

1957 yil ARPA (Advanced Research Projects Agency) tashkiloti tuzildi. 60-yillar oxirida DARPA (Defense Advanced Research Project Agency) ARPANet tajriba tarmo'ini tashkil etish shaqida qaror qabul qildi. Ilk bor tarmoq 1972 yilda namoyish etildi. U 40 ta kompyuterdan iborat blib, asosiy tuzilish printsipi tarmoqdagi barcha kompyuterlarning teng huquqli blishi edi. 1975 yil ARPANet tajriba tarmog'i maqomini sharakatdagi (amaliy) tarmoq maqomiga o'zgartirdi.

80-yillar boshida tarmoqda mashinalarning zaro ta'sir protokollari standartlashtirildi. Boshlang'ich variant TCP/IP (Transfer Control Protocol/Internet Protocol). BBN kompaniyasi bilan shartnoma tuzildi, bu esa TCP/IPni UNIX OT safiga kiritdi. 1983-yilda **ARPANET-INTERNET** deb atala boshlandi va juda kuchli, bir-biri bilan bog'langan kompyuterlar va tarmoqlar to'plamidan iborat tizimsiga aylandi.

1980-yillar INTERNETning keskin o'sish davri bo'ldi. Kompyuterlarning markazlashmagan boshqarish tarmog'i bilan bog'lanish sxemasi butun dunyoga tarqaldi va chet el tarmoqlari tashkilotchilari AQSH tarmog'iga ulanishga rozi bulishdi. INTERNET ning butun dunyoni qamrab olishi quyidagi tarmoqlarning kushilishi hisobiga bo'ldi. **NSFNET** - AQSHning ilmiy-tadqiqot institutlarini, korporatsiya va xukumat idoralarini birlashtiruvchi tarmog. (1980 yil) **EUNET** (Europe Union Network) - Yevropaning UNIX operatsion sitemasida va UUCP hamda TCP/IP da ishlaydigan mashinalari tarmog'i. Markazi Amsterdam shaxrida. (1982 yil) **EARN** (European Academic Research Network)- Yevropaning o'quv, ilmiy-tekshirish va tadqiqot muassasalarining tarmog'i (1983 yil) **JUNET** - Yaponiyaning UNIX mashinalari tarmog'i (1984 yil) **JANET** - Buyo'q Britaniyaning birlashgan akademik tarmog'i (1984 yil)

Shulardan **NSFNET**ni - **Internet Backbone** yoki "*internetning asosi*" deb atashadi. Tashkil etilgan vaqtda 1980 yilda u 56 Kbit-s tezligida axborot uzata olish qobiliyatiga ega edi. 1988 yilda esa unin tezligi 1,544 Mbit-s oshdi. 1991-yili NSFNET tarmog'i takomillashtirildi va uzatish tezligi 44.736 Mbit-s ga yetdi.

Internetga ulangan kompyuterlar soni 1987 yilda 10 000 bo'lsa, 1989 yilda 100 000 taga yetdi, 1995 yilda esa 6,5 million deb hisoblangan. Hozirgi kunda esa dunyoning 150 dan ortiq mamlakatida 100 milionlab kompyuterlar Internetga ulangan bo'lib, har oyda tarmoq abonentlar mikdori 7-10% ortib bormoqda. Internet dagi kompyuterlar aksariyati AQSH da joylashgan.

Internet ning 90-yillardagi rivojlanishiga asosiy sabab World Wide Web (Butun Dunyo Tarmog'i) ning tuzilishi bo'ldi. Uni birinchi nusxasini 1990 yilining noyabrida **CERN** (Yevropa atomni tekshirish markazi) xodimi Tim Berns-Li yaratdi, lekin 1992 yilgacha ishga tushirilmadi. 1993 yilda **NCSA** (National Center for Supercomputer Applications, Superkompyuterli Hisoblash Milliy Markazi) tomonidan **Mosaic** dastursi ishlab chikarildi va shu yil oxiriga 200 ta WWW server ishga tushirilib WWW bo'yicha axborot okimi 1% ni tashkil qilgan.

Internet va WWW bir xil emas. Internet butun dunyo kompyuterlar tarmoqlarining to'plamini belgilaydi va turli xil kompyuter xizmatlarini ko'rsatadi. Bu - **E-mail elektron pochta, Usenet telekonfirentsiyalari, FTP ma'lumot fayillarini uzatish tizimsi, Telenet uzokdan terminalga kirish tizimsi, Gopher**

tizimsi va Butun Dunyo Tarmog'i- WWW. Demak WWW Internetning faqatgina bir qismidir. Lekin u juda tez rivojlanmoqda.

WWW - Internet ning ommabop xizmat turidir. Unga ulanish uchun kompyuter bilan modem yetarlidir. Shuning uchun Butun dunyo tarmog'i butun olam axborotlar ombori-kutubxonaga aylanib qoladi va u dunyoga yoyiladi. WWW da ma'lumotlar sahifalarda joylashadi. WWW sahifalarning soni oxirgi 3 yilda yuz milliondan oshib ketdi. Bu sahifalarning egasi kim? Ular yirik korporatsiyalar yoki kichik korxonalar, universitet va maktablar, tashkilotlar, jurnal va ruznomalar yoki oddiy shaxslardir. Bu sahifalarda turli-tuman ma'lumotlar joylanadi. Hozirgi kunda WWW axborot olishning va tarkalishning eng qulay usulidir. U unga kiruvchi va ulangan barcha kompyuterlarning o'zaro ma'lumotlar almashish imkoniyatini yaratib beradi.

Kompyuterdan foydalanish modellari

Har bir shaxs uz uyida turib katta masofalarda joylashgan ma'lumotlarni olishi mumkin. Uzoq masofalarda joylashgan ma'lumotlarni olishi mumkin. Komp'yuter tarmoqlari har-xil loyihalar bo'yicha tashkil etiladi. Barcha ko'rsatkichlardan quyidagi ikkitasi asosiy hisoblanadi:

1. Axborotlarni uzatish texnologiyasi.
2. Axborotlarni uzatish miqyosi, masshtabi

Hozirgi kunda ikkita ikki xil texnologiya qo'llanilmoqda.

1. Keng ko'lamli axborot uzatish texnologiyasi bu texnologiyaga server tamonidan uzatilgan axborot tarmog'idagi barcha kompyuter tamonidan qabul kilinishi mumkin.
2. Axborotlarni ikki nuqtali uzatish texnologiyasi (point-to-point) nuqtadan nuqtagacha bu texnologiyada faqat bir juft kompyuter xizmat qiladi.

Axborotlarni uzatish miqyosi bo'yicha kompyuter tarmoqlari bir nechta turlarga bo'linadi.

Protsessorlar orasidagi masofa	Protsessorlarning joylashishi	Nomlanishi
0,1m	Ulanish chizigi	Ma'lumotlar oqimini qayta ishlaydigan mashina
10m 100m 1000m	Katta xona Bino Kompus(compus)	Maxalliy kompyuter tarmoqlari LAN(Local Area Networks)
10km	Maxalliy kompyuter tarmoqlari	MAN(Metropolitan Area Networks)

100km 1000km	Viloyat kompyuter tarmoqlari	Wan(Wide Area Networks)
10.000km	Dunyo miqyosidagi kompyuterlar	Internet (Internet Computer Networks) O'zaro bog'langan kompyuter tarmoqlari

Kompyuter bu hisoblash ishlarni va mantiqiy karorlarni insonga nisbatan million va hattoki milliard marta tezrok bajara oladigan qurilmadir. Takkoslash uchun shuni aytish lozimki zamonaviy shaxsiy kompyuterlar 1-sekundda bajaradigan ishlarni kalkulyator 10-yil mobaynida bajaradi. Kompyuter barcha amallarni kompyuter uchun tuzilgan dasturlar deb ataluvchi ko'rsatmalar asosida bajaradi. Bu ko'rsatmalarni yani dasturlar kompyuter dasturchilari bajaradi. **Microsoft** firmasi dasturiy taminotni ishlab chiqdi. Kompyuterning apparat qismi yoki manitor, printer, tizimli blok, skaner, modem, sichqoncha, audio-video kompyuterning **Hardware** hard (qattiq) ware (buyum) qismi deyiladi. Kompyuterning ishlayotgan dasturiy majmuasi esa kompterning dasturli taminoti deyiladi, va kompyuterning Software soft (yumshoq) qismi deyiladi. Internetdan foydalanish **WWW (World Wide Web-** Jahon o'rgimchak to'ri) tizimidan foyalanish uchun yangi dasturlashtirish tillari yaratildi. Bu tillardan biri **Java-** dasturlashtirish tili 1995 yildan boshlab ishlatilmoqda.

Lokal kompyuter tarmog'i

Xududiy taqsimlanishi jixatidan kompyuter tarmoqlarini uch guruhga ajratish mumkin.

1. Lokal tarmoqlar (**LAN–Local Area Network**), bir korxona, muassaning bir yoki bir nechta yaqin binolardagi ob'ektlarni boglaydi. Lokal tarmoqdagi kompyuterlar orasidagi masofa uncha katta emas, agar radiokanal aloqasidan foydalansa 20 kmni tashkil qiladi.

Lokal tarmoqda kompyuterlarni birlashtiruvchi sim (kabel) sifatida kalin koaksil, ingichka koaksil, juft-juft qilib uralgan (toking Ring «vitaya para») optik tukima (tola) simlari ishlatilishi mumkin.

Kompyuter tarmog'i maxsus operatsion tizim boshqaruvida ishlaydi. Hozir ko'p ishlatilayotgan Windows-95 operatsion tizim tarkibida lokal tarmoqda ishlash imkonini beruvchi dasturlar mavjud.

Lokal kompyuter tarmoqlari boshqa kompyuter tarmoqlari turlaridan quydagi xususiyatlari bilan farq qiladi:

1. Uzining o'lchamlari ;
2. Axborot almashish texnologiyasi bilan;
3. Topologiyasi (tuzilishi) bilan

Lokal kompyuter tarmog'ining o'lchamlari uncha katta bo'lmaganligi sababli, ular o'ziga xos dizayniga ega bo'lishi mumkin. Ko'p hollarda ular bitta kabeldan tashkil topgan axborot tashish texnologiyasiga ega bo'ladi. Lokal kompyuter tarmog'ining axborot uzatish tezligi 10 Mb/sek dan 100 Mb/sekgacha bo'ladi.

Lokal kompyuter tarmog'ining topologiyasi ko'p hollarda quydagi ikki ko'rinishga ega bo'ladi: **to'g'ri Chiziqli** topologiya va **aylana** ko'rinishli topologiya. Bunday bir topologiyada istalgan kompyuter ishlab turgan bo'lishi mumkin, hozirgi ongda ishlab turgan, ya'ni tarmoqqa axborot jo'natayotgan kompyuter **мастер** deyiladi.

Master ishini kolgan kompyuterlar kutib turadi. Shuning uchun 2 ta kompyuter bir vaqtning uzida axborot jo'natish xollarini bartaraf etish rejimi statsionar va dinamik rejim bo'lishi mumkin. Axborot jo'natish mexanizimi markazlashtirilgan va markazlashtirilmagan bo'lishi mumkin. Har qanday kompyuter tarmog'idan foydalanishning shu tarmoqqa xos koidalari, ya'ni **protokollari** ishlab chiqilgan.

2. Shahar miqyosidagi kompyuter tarmoqlari (**Metropolitan Area Networks MANs**).

Shahar miqyosidagi kompyuter tarmoqlari umuman olganda Lokal kompyuter tarmoqlariga uxshash bo'lib, ishlash texnologiyasi deyarli bir xil.

Odatda MANlarga kushni ofeslar birlashtiriladi. Yoki shahar miqyosidagi korxonalar ofislari birlashtiriladi. MANlar xususiy tarmoqlar bo'lishi mumkin. MANlarni Lokal televizion tarmoqlarga ham ulash mumkin. MANlarning alohida kategoriyalariga bo'linishi sabab shundaki, ular uchun xalqaro standartlar ishlab chiqilgan. Bu standartning nomi **DQDB** (Distributed Queue Dual Bus)-tarkatilgan navbat ikkilangan kanal, magistral. DQDB –standarti bo'yicha kompyuterlarni tarmoqqa ulash sxemasi quydagicha bo'ladi. Bu sxema 2 ta bir xil magistrantdan iborat bo'lib, bularga tarmoqqa kiradigan kompyuterlar kiradi. Har bir magistral axborotlarni jo'natish qurilmasiga ega bo'ladi. Bu qurilma **head-end** (boshi-oxiri) deyiladi.

Har bir kompyuterdan bu kompyuter ung tomonga joylashgan kompyuterlarga axborot jo'natish uchun yuqori magistaldan foydalanishga to'g'ri keladi. Chap tomondagi kompyuterlarga ma'lumot yuborish uchun esa quyi magistraldan foydalanish kerak.

MANlarni asosiy xususiyati shundan iboratki, axborotlarni keng ko'lamda tarkatish imkoniyatlari bor, bu kompyuter tarmog'i o'ziga xos dizaynerga ega bo'ladi. MANlarni topologiyasi regulyar (to'g'ri) sxemada bo'ladi.

Global kompyuter tarmoqlari (Wide Area Networks, WANs)

Texnologiyasi, miqyosi bilan kompyuter tarmoqlari bir–biri bilan farq qiladi. Odatda keng zonali kompyuter tarmoqlari bir kichikroq mamlakat, yoki kichikroq kit‘a miqyosida ishlatiladi.

WANlar foydalanuvchilar masalalarini yechishga mo‘ljallangan kompyuterlar kolleksiyasidan tashkil topadi. Kelgusida bu kompyuterlarni asosiy kompyuterlar yoki (host computers) **host** deb yuritamiz.

WAN larda host (asosiy) kompyuterlar bir-biri bilan kommunikatsiya tizim osti deb ataluvchi tizim osti bilan ulanadi. Kompyuter tarmog‘ining kommunikatsiya tizim osti **subnet** deb ataladi.

Subnetning asosiy vazivasi–tarmoqdagi asosiy kompyuterlar orasida aloqa o‘rnatishdan, ya‘ni tarmoqdagi bir kompyuterdan ikkinchi asosiy kompyuter axborot uzatishdan iboratdir. Keng zonali kompyuter tarmoqlarida–WANlarda quydagi ikkita tushunchani ajratib takidlab o‘tamiz.

1. Aloqa tarmoqlari (Subnet)
2. Tarmoqning amaliy masalalarini yechishda kullaniladigan qismi, ya‘ni host (asosiy) kompyuterlar.

Bu 2ta tushunchani bir-biridan ajratish WANlarni loyخالashni, ya‘ni ularni dizaynini soddalashtirish imkoniyatini beradi. Ko‘p hollarda komutatsiya yoki aloqa Subneti quydagi 2ta har xil kompyuterlardan iborat bo‘ladi:

1. Uzatish tarmoqlari (liniya peredachi)
2. Boshqa aloqa tarmog‘iga ulash elementlari.

Uzatish tarmoqlarining vazifasi aloqa axborotlarini uzatish, ya‘ni bitlarni uzatishdan iborat).

Axborotlarni bitta asosiy kompyuterdan boshqa asosiy kompyuterlarga uzatiladi. Bu alohida aloqa tarmog‘ining vazifasi. Bitta aloqa tarmog‘ini boshqa aloqa tarmog‘i bilan ulaydigan elementlarning vazifalari bir aloqa tarmog‘ini boshqa aloqa tarmog‘i bilan ulashdan iboratdir. Bu vazifani, ya‘ni bir tarmoqni ikkinchi tarmoq bilan ulash vazifasini maxsus moslashtirilgan kompyuterlar bajaradi. Tarmoqlarni ulash elementiga kiruvchi tarmoqdan axborot kelganda ulash elementi kelgan axborotni jo‘natish uchun chiqish tarmog‘ini tanlashi kerak. Tarmoqlarni ulaydigan kompyuterlarni **router** deb qabul kilamiz.

Router – yo‘naltiruvchi, marshrutizator degan manoni bildiradi.

Lekin asosiy kompyuterlar har doim LANlarga ulangan bo‘lishi shart emas. Agar bitta router dan jo‘natilgan axborot, ikkinchi router yetib borishi uchun orada 2-3 ta routerlardan utsa, tabiyki, jo‘natilgan axborot oralikdagi routerlarga kelib kiradi. Ma‘lumki bu oralik routerda saqlanib turishi mumkin, keyin chiqish yuli bush bo‘lganda jo‘natiladi. Axborotni bir routerdan boshqaga yetib borish modeli 2

nuqtali model deyiladi (**Point to point**). Bu modelda axborot jo'natuvchi va axborot qabul qiluvchi bor.

Katta xajimga ega bo'lmagan axborotlar va o'zaro teng bo'lgan axborotlar adabiyotda **cellelar** deyiladi. Keng zonali kompyuter tarmog'ining WANlarni topologiyasi irregulyar (noto'g'ri) sxemada bo'ladi.

Bulut texnologiyalari va ularning modellari. Bulutli dasturlar - bu turli xil Internet xizmatlarini bajarish. Bunday vazifalarga misol qilib, ma'lumotlarni saqlash, serverlar, ma'lumotlar bazalari, dasturlar va ijtimoiy tarmoqlar kiradi. Bulutli dasturlar ma'lumotlarni saqlashni kompyuter fayllariga yoki qattiq diskka emas, balki markaziy tarmoqqa saqlashga imkon beradi. Ko'p sabablarga ko'ra, Cloud Computing - bu jismoniy shaxslar va kompaniyalar uchun mashhur variant, jumladan vaqtni tejash, samaradorlikni oshirish, tezlik va himoya.



Bulut texnologiyasi bu ommabop texnologiya bo'lib, unda foydalanuvchilar Internet-platformada IT-resurslarda ma'lumot kiritish yoki olish uchun ishlatadilar. Bu texnologiya ma'lumotlar va resurslarni onlayn-serverda saqlaydigan va foydalanuvchi to'g'ridan-to'g'ri kompyuterning qattiq diskida saqlash o'rniga xavfsiz foydalanish uchun ishlatiladi. Bulutli texnologiyalarining yana bir yaxshi tarafi bu online serverlardan dunyoning xohlagan nuqtasida turib ma'lumotlarni kiritish yoki qabul qilish mumkin. Bulutli texnologiyalarning turli xil xizmatlari mavjud bo'lib, ular juda mashhur va bugungi kunda dunyoda keng qo'llanilayotgan katta hajmdagi saqlash va uni zaxiralash, dasturlarni sinash va texnik xizmat ko'rsatish, ma'lumotlarni tahlil qilish va kerakli dasturlarni yetkazib berish mumkin.

Nima uchun biz uni ishlatamiz?

Bulutli dasturlar texnologiyasi - bu Internetga asoslangan platforma, bu resurslar serverdan mijozga tarmoqlar kanallari orqali ma'lumot yetkazadi. Shu sababli, texnologiya yorug'lik tezligida yangilanayotgan hozirgi dunyoda qaysi texnologiyaga o'tish kerakligini va texnologiyadan to'g'ri foydalanishni bilish juda muhimdir. Shunday qilib, bulutli texnologiyalarning mashhurligi va Internetning

salbiy ta'siriga qaramay, u eng tezkor texnologiya bo'lib qoladi va u bugungi bozorning so'nggi tendentsiyasiga aylandi. Demak, undan foydalanishning maqsadi butun dunyo bo'ylab xaridorlarni bir-biriga bog'lash orqali resurslarni diversifikatsiya qilish va o'z vaqtida kerakli resurslardan foydalangan holda ushbu sohadagi mijozlarini saqlab qolish orqali biznes foydalanuvchilariga o'z bozorlarini kengaytirishga imkon berishdir. Mening fikrimcha, hamma bulutli dasturlarni nima uchun ishlatadi, bu bizga har qanday hajmdagi fayllarni zaxira qilishning eng zo'r tizimini taqdim etadi. Axborot texnologiyalari mutaxassislari uchun bu dunyodagi istalgan joydan resurslarga kirish va ishlash imkoniyatiga ega bo'lgan so'nggi texnologiya tendentsiyasining ulkan o'sishi bo'ldi va mijozlar bilan ishlash va ishlashni yanada moslashuvchan qildi.

Bulutli hisoblash xizmatlari uchta toifaga bo'linadi: omma buluti, bulut buluti va gibrid bulut. Xususiy bulut butunlay mamlakat ichkarisida yoki uchinchi tomon tomonidan boshqariladigan va ichki yoki tashqi tomonga joylashtiriladigan bitta korxona uchun mo'ljallangan infratuzilma. Ushbu model korxonalarga moslashuvchanlikni ta'minlaydi, boshqaruvni, nazoratni va xavfsizlikni ta'minlaydi. Avtoullov IT boshqaruvi tomonidan IT-resurslarini tezkor ravishda sotib olish va tarqatish imkonini beradigan boshqaruv interfeysi orqali boshqarishga kirishdir.

1. Bulutli dasturiy ta'minot sifatida xizmat (SaaS) - bulutli dasturiy ta'minot, bundan keyin "Xizmatlar uchun dasturiy ta'minot";
2. Cloud Platform xizmati (PaaS) - bu xizmat sifatida bulutli platforma;
3. Bulut infratuzilmasi (IaaS) - bulut infratuzilmasi sifatida xizmat.

Biz faqat ushbu xizmatning mavzusiga mos ravishda birinchi xizmat turini ochamiz. **Dasturiy ta'minot sifatida (SaaS)** Iste'molchining bulutli infratuzilmada ishlaydigan dasturlardan foydalanish imkoniyati mavjudmi? Ilovalarga kirish turli mijozlar qurilmalaridan, masalan, veb-brauzer orqali nozik dastur interfeysi orqali amalga oshiriladi. Iste'molchi dastur ishlaydigan bulut infratuzilmasini boshqarmaydi yoki boshqarmaydi, jumladan tarmoq, server, operatsion tizim, ma'lumotlar do'koni va hatto dastur sozlamalari. Mumkin bo'lgan istisno alohida dastur parametrlari.

Xuddi shunday, resurslarni boshqarish ham saqlash, ham tahlil qilish uchun avtomatlashtirilgan. Xuddi shu tarzda, xavfsizlik va menejment kompaniyaning o'ziga xos ehtiyojlariga qaratilgan. Bu tendentsiya bir necha omillarga bog'liq. Birinchidan, ayrim kompaniyalar faqat ko'proq variantlarni xohlashadi. Ko'p sotuvchini ishlatish bulutli texnologiyalar turli ilovalar va ish yuklarini qo'llab-quvvatlash uchun tashkilotlar o'z ehtiyojlariga eng mos keladigan echimdan foydalanishlari mumkin.

Yagona bulut modeliga asoslanib, kompaniya o'chirib qo'yilishi mumkin. Shuning uchun, bir nechta birliklari bo'lgan katta kompaniyalar uchun bir necha bulutlardan foydalanish muqarrar. Bundan tashqari, bitta bulut xizmatidan foydalanish ma'lumotlarning markazlari yoki tarmoqli kengligi muammolari kabi turli muammolarga kompaniyaning zaifligini oshiradi. Doimiy ravishda oflayn

rejimda ishlaydigan bulutli dastur kompaniyaning yomon qiyofasini beradi va mijozlarning yo'qolishiga olib kelishi mumkin. Agar dastur bitta bulutli provayderga bog'liq bo'lsa, kelishmovchilik yuzaga kelgan taqdirda ham muzokara qilish qiyin.

Deployment modellari:

1. Xususiy Cloud
2. Omma buluti
3. Ommaviy bulut
4. Hibrid bulut (gibrid bulut)

Biz uchinchi tarqatish modelini faqat ushbu maqola mavzusiga mos ravishda o'chamiz. **Umumiy bulut (Ommaviy Bulut)** - bu modelda bulut infratuzilmasi hamma uchun yoki keng sanoat guruhi uchun mavjud bo'lib, bulut xizmat provayderiga tegishli.

Ma'lumotlarning suverenitetiga taalluqli qarama-qarshiliklar ko'p bulutli, ayniqsa, Yevropa kompaniyalarining o'sishiga olib keladi. Mahalliy ma'lumotlarni saqlash mahalliy hokimiyatning bu muammolarini kamaytiradi. Boshqa tarafdin, ma'lumotlarni uzatish markazlariga trafikni foydalanuvchilarga iloji boricha yaqinroq qilish, kechikish yuz beradigan ilovalar uchun muhimdir.

Juda ko'p bulutlar afzalliklarga ega, lekin ular biznes uchun murakkab bo'lishi mumkin, ayniqsa, bir bulutdan ikkinchisiga o'tishda. Afsuski, barcha hisoblash muhiti har xil va bulut hech qanday istisno emas. Bulut provayderlari dasturlarni o'z platformalariga o'tkazishni soddalashtirish uchun barcha sa'y-harakatlar qilmoqdalar, biroq mijozlar o'z xizmatlarini tark etmasligini ta'minlash uchun barcha imkoniyatlarni ishga soladilar.

Yuqorida biz bulutli hisoblash ta'rifini berdik va ularning asosiy xususiyatlarini tasvirladik. Biz, shuningdek, bulutli hisoblash xizmati modellar va kengaytirish modellari klassifikatsiyasini olib, bu bulut turlari mavjud nima haqida hikoya qilinadi. Shunday qilib, kichik kompaniyalar uchun bulutli xizmat nima? tushunchasi, biz bir bulut infratuzilmasi foydalanib, ya'ni, osmondan taqdim xizmatlari haqida gapiryapsiz, degan ma'noni anglatadi so'z "bulut" va "xizmatlar", bor.

Ko'pgina kompaniyalar ikkita bulut xizmatining ma'lumotlarini petabaytlarini ko'chirish vaqti haqida juda xavotirda. Bu moslashuvchanlikni, mablag'larni tejashni va, ehtimol, yanada yaxshiroq echimlarni topadigan kompaniyalar uchun yaxshi yangilik. bir xizmat qilib, bir xizmat va dasturiy ta'minot sifatida xizmat, platforma sifatida infratuzilmasini: cloud computing marta rivojlanayotgan bo'lsa-da, u hali xizmatlar uch toifaga bo'linadi.

Foydalanuvchilarning alohida imkoniyatlari mavjud. Keyin ular boshlashlari, to'xtatishlari yoki sozlashlari mumkin virtual mashina va o'zingizni saqlashingiz mumkin. Berilgan infratuzilma turli ehtiyojlarni qondirish uchun kichik, o'rta, katta yoki juda katta bo'lishi mumkin. Bunday infratuzilma bilan korxonalar o'z uskunalariga sarmoya kiritishga hojat yo'q.

Ushbu xizmatlar "kichik kompaniyalar uchun" mo'ljallanganligi sababli:

1. Ushbu xizmatlar biznes yuritish uchun yordam berishi kerak;
2. Ushbu xizmatlar kichik kompaniyalar uchun mos bo'lishi kerak;
3. Ular ommaviy tarzda foydalanishlari kerak;
4. Foydalanuvchilarning ma'lum bilimlarga ega bo'lishini talab qilishlari kerak (masalan, axborot texnologiyalari sohasida).

Yuqoridagilardan kelib chiqadigan bo'lsak, biz quyidagi ta'riflarni beramiz. **Kichkina kompaniyalar uchun bulutli xizmatlar** - Ushbu dastur narhlarda mijozlar keng jamoatchilik bulut va mavjud orqali SaaS (xizmat qilib dasturiy ta'minot) sifatida tarqatilgan ish avtomatlashtirish, uchun.

bulut muhitlar, rivojlanish etkazib berish bulut ilovalar to'liq hayot tsikli uchun zarur hamma narsani ta'minlash - bir xizmat, deb platforma. Ular sizni o'zingiz sotib olish va asbob-uskunalar bilan ta'minlashdan ozod qilasiz, dasturiy ta'minot va hosting. Platformali provayderlar, xizmatlar sifatida, ularning infratuzilmalari uchun ishlab chiquvchi vositalarni tarqatadilar. Ular sizga yangi ilovalarni bir necha daqiqada bulutga joylashtirish imkonini beradi. Ushbu ilovalar odatda veb-xizmatlar deb ataladi.

Xulosa:

Shunday qilib, bulutli texnologiyalar har bir foydalanuvchi uchun hayotni osonlashtiradigan eng tez rivojlanayotgan va tezkor daromad keltiradigan texnologiyalarning yorqin misolidir. Internet yoki bulutli dasturlar bizni o'z tarmoqlarini yuklash tizimi orqali bog'laydi, bu bizning hayotimizni qulay qiladi va xavfsiz amalga oshadi.

Nazorat savollari

1. Kompyuter tarmog'i deb nimaga aytiladi?
2. Kompyuter tarmoqlari bir-biridan nima bilan farqlanadi?
3. Lokal kompyuter tarmog'i deganda nimani tushinasiz?
4. Shahar miqyosidagi kompyuter tarmog'i deganda nimani tushinasiz?
5. Keng zonali (global) kompyuter tarmog'i deganda nimani tushinasiz?
6. Bu tarmoqlarning topologiyalari qanday bo'ladi
7. Bulutli hisoblash texnologiyasi deganda nimani tushunasiz?

Maruza №8

Mavzu: Texnik va texnologik jarayonlarida axborot xavfsizligini ta'minlash

Reja:

1. Axborot xavfsizligiga kirish
3. Axborotlarga nisbatan xavf-xatarlar tasnifi
4. Tarmoq xavfsizligini nazorat qilish vositalari

Tayanch so'zlar: maxfiylik, konfidentsiallik, yaxlitlik, autentifikatsiya, apellyatsiya qilishlik, ishonchlilik, aniqlilik, tizimga kirishni nazorat qilish, identifikatsiyalashni nazorat qilish, qasddan buzilishlarga to'sqinlik.

1. Axborot xavfsizligiga kirish.

Mamlakatimiz milliy iqtisodining hech bir tarmog'i samarali va mo'tadi tashkil qilingan axborot infratuzilmasisiz faoliyat ko'rsatishi mumkin emas. Hozirg kunda milliy axborot resurslari har bir davlatning iqtisodiy va harbiy salohiyatin tashkil qiluvchi omillaridan biri bo'lib xizmat qilmoqda. Ushbu resursdan samaral foydalanish mamlakat xavfsizligini va demokratik axborotlashgan jamiyatn muvaffaqiyatli shakllantirishni ta'minlaydi. Bunday jamiyatda axborot almashuv tezligi yuksaladi, axborotlarni yig'ish, saqlash, qayta ishlash va ulardan foydalanish bo'yicha ilg'or axborot – kommunikatsiyalar texnologiyalarini qo'llash kengayadi. Turli xildagi axborotlar hududiy joylashishidan qat'iy nazar bizning kundalik hayotimizga Internet halqaro kompyuter tarmog'i orqali kirib keldi. Axborotlashgar jamiyat shu kompyuter tarmog'i orqali tezlik bilan shakllanib bormoqda. Axborotla dunyosiga sayohat qilishda davlat chegaralari degan tushuncha yo'qolib bormoqda. Jahon kompyuter tarmog'i davlat boshqaruvini tubdan o'zgartirmoqda, ya'ni davla axborotlarning tarqalishi mexanizmini boshqara olmay qolmoqda. Shuning uchur ham mavjud axborotlarga noqonuniy kirish, ulardan foydalanish va yo'qotish kab muammolar dolzarb bo'lib qoldi. Bularning bari shaxs, jamiyat va davlatning axboro xavfsizligi darajasining pasayishiga olib kelmoqda. Davlatning axborot xavfsizligin ta'minlash muammosi milliy xavfsizlikni ta'minlashning asosiy va ajralmas qism bo'lib, axborot himoyasi esa davlatning birlamchi masalalariga aylanmoqda.

Hozirgi kunda xavfsizlikning bir qancha yo'nalishlarini qayd etish mumkin.

Avtomatlashtiriltan axborot tizimlarida axborotlar o'zining hayotiy davriga ega bo'ladi. Bu davr uni yaratish, undan foydalanish va kerak bo'lmaganda yo'qotishdan iboratdir (2-rasm).

Axborotlar hayotiy davrining har bir bosqichida ularning himoyalanganlik darajasi turlicha baholanadi.

Maxfiy va qimmatbaho axborotlarga ruxsatsiz kirishdan himoyalash eng

muhim vazifalardan biri sanaladi. Kompyuter egalari va foydalanuvchilarning mulki huquqlarini himoyalash - bu ishlab chiqarilayotgan axborotlarni jiddiy iqtisodiy va boshqa moddiy hamda nomoddiy zararlar keltirishi mumkin bo'lgan turli kirishlar va o'g'irlashlardan himoyalashdir.

Axborot xavfsizligi deb, ma'lumotlarni yo'qotish va o'zgartirishga yo'naltirilgan tabiiy yoki sun'iy xossali tasodifiy va qasddan ta'sirlardan har qanday tashuvchilarda axborotning himoyalanganligiga aytiladi.

Ilgarigi xavf faqatgina konfidentsial (maxfiy) xabarlar va hujjatlarni o'g'irlash yoki nusxa olishdan iborat bo'lsa, hozirgi paytdagi xavf esa kompyuter ma'lumotlari to'plami, elektron ma'lumotlar, elektron massivlardan ularning egasidan ruxsat so'ramasdan foydalanishdir. Bulardan tashqari, bu harakatlardan moddiy foyda olishga intilish ham rivojlandi.

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.

Axborotning egasiga, foydalanuvchisiga va boshqa shaxsga zarar etkazmoqchi bo'lgan nohuquqiy muomaladan har qanday **hujjatlashtirilgan**, ya'ni identifikatsiya qilish imkonini beruvchi rekvizitlari qo'yilgan holda moddiy jismda qayd etilgan **axborot** himoyalaniishi kerak.

Axborot xavfsizligi nuqtai nazaridan axborotni quyidagicha turkumlash mumkin:

- **maxfiylik** — aniq bir axborotga faqat tegishli shaxslar doirasigina kirishi mumkinligi, ya'ni foydalanilishi qonuniy hujjatlarga muvofiq cheklab qo'yilib, hujjatlashtirilganligi kafolati. Bu bandning buzilishi **o'g'irlik** yoki **axborotni oshkor qilish**, deyiladi;

- **konfidentsiallik** — inshonchliligi, tarqatilishi mumkin emasligi, maxfiyligi kafolati;

- **yaxlitlik** — axborot boshlang'ich ko'rinishda ekanligi, ya'ni uni saqlash va uzatishda ruxsat etilmagan o'zgarishlar qilinmaganligi kafolati; bu bandning buzilishi **axborotni soxtalashtirish** deyiladi;

- **autentifikatsiya** — axborot zahirasi egasi deb e'lon qilingan shaxs haqiqatan ham axborotning egasi ekanligiga beriladigan kafolat; bu bandning buzilishi **xabar muallifini soxtalashtirish** deyiladi;

- **apellyatsiya qilishlik** — etarlicha murakkab kategoriya, lekin elektron biznesda keng qo'llaniladi. Kerak bo'lganda xabarning muallifi kimligini isbotlash mumkinligi kafolati.

Yuqoridagidek, axborot tizimiga nisbatan quyidagicha tasnifni keltirish mumkin:

- **ishonchlilik** — tizim me'yoriy va g'ayri tabiiy hollarda rejalashtirilganidek o'zini tutishlik kafolati;

- **aniqlilik** — hamma buyruqlarni aniq va to'liq bajarish kafolati;

- **tizimga kirishni nazorat qilish** — turli shaxs guruxlari axborot manbalariga har xil kirishga egaligi va bunday kirishga cheklashlar doim bajarilishlik kafolati;

- **nazorat qilinishi** — istalgan paytda dastur majmuasining xoxlagan kismini to'liq tekshirish mumkinligi kafolati;

- **identifikatsiyalashni nazorat qilish** — hozir tizimga ulangan mijoz aniq o'zini kim deb atagan bo'lsa, aniq o'sha ekanligining kafolati;

- **qasddan buzilishlarga to'sqinlik** — oldindan kelishilgan me'yorlar chegarasida qasddan xato kiritilgan ma'lumotlarga nisbatan tizimning oldindan kelishilgan holda o'zini tutishi.

Axborotni himoyalashning maqsadlari quyidagilardan iborat:

- axborotning kelishuvsiz chiqib ketishi, o'g'irlanishi, yuqotilishi, o'zgartirilishi, soxtalashtirilishlarning oldini olish;

- shaxs, jamiyat, davlat xavfsizligiga bo'lgan xavf – xatarning oldini olish;

- axborotni yo'q qilish, o'zgartirish, soxtalashtirish, nusxa ko'chirish, tusiqlash bo'yicha ruxsat etilmagan harakatlarning oldini olish;

- hujjatlashtirilgan axborotning miqdori sifatida huquqiy tartibini ta'minlovchi, axborot zaxirasi va axborot tizimiga har qanday noqonuniy aralashuvlarning ko'rinishlarining oldini olish;

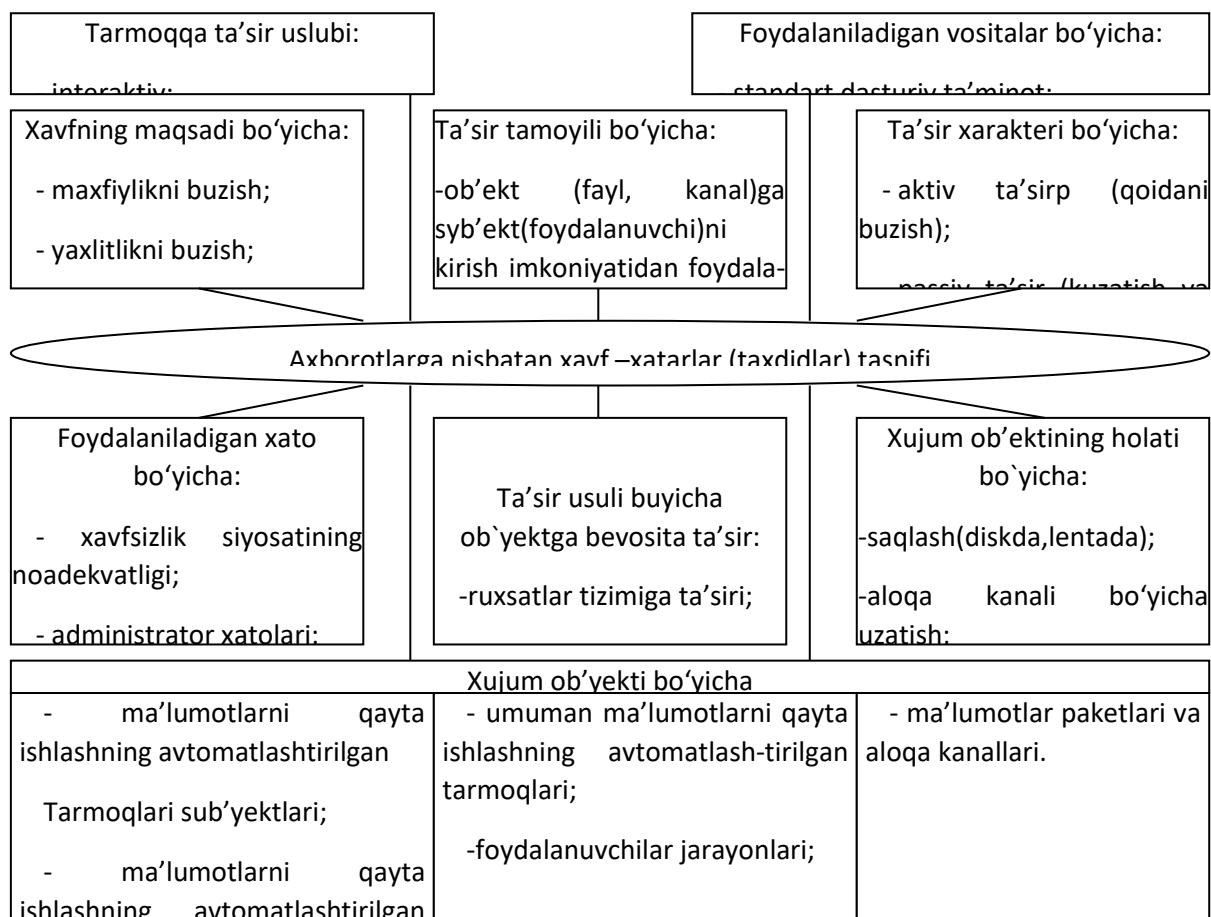
- axborot tizimida mavjud bo'lgan shaxsiy ma'lumotlarning shaxsiy maxfiyligini va konfidentsialligini saqlovchi fuqarolarning konstitutsion huquqlarini himoyalash;

- davlat sirini, qonunchilikka mos hujjatlashtirilgan axborotning konfidentsialligini saqlash;

- axborot tizimlari, texnologiyalari va ularni ta'minlovchi vositalarni yaratish, ishlab chiqish va qo'llashda sub'ektlarning huquqlarini ta'minlash.

Axborotlarga nisbatan xavf-xatarlar tasnifi

Ilmiy va Amaliy tekshirishlar natijalarini umumlashtirish natijasida axborotlarga nisbatan xavf xatarlarni quyidagicha tasniflash mumkin.



Xavfsizlik siyosatining eng asosiy vazifalaridan biri himoya tizimida potentsial xavfli joylarni qidirib topish va ularni bartaraf etish hisoblanadi.

Tekshirishlar shuni ko'rsatadiki, tarmoqdagi eng katta xavflar — bu ruxsatsiz kirishga mo'ljallangan maxsus dasturlar, kompyuter viruslari va dasturning ichiga joylashtirilgan maxsus kodlar bo'lib, ular kompyuter tarmoqlarining barcha ob'ektlari uchun katta xavf tug'diradi.

Tarmoq xavfsizligini nazorat qilish vositalari

Zamonaviy axborot - kommunikatsiyalar texnologiyalarining yutuqlari himoya uslublarining bir qator zaruriy instrumental vositalarini yaratish imkonini berdi.

Axborotlarni himoyalovchi instrumental vositalar deganda dasturlash, dasturiy - apparatli va apparatli vositalar tushuniladi. Ularning funksional to'ldirilishi xavfsizlik xizmatlari oldiga qo'yilgan axborotlarni himoyalash masalalarini echishda samaralidir. Hozirgi kunda tarmoq xavfsizligini nazorat qilish texnik vositalarining juda keng spektri ishlab chiqarilgan.

Nazorat uchun savollar:

1. Axborotlar himoyasi faninig maqsadi nimalardan iborat?
2. Fanning asosiy tushunchalari haqida ma'lumot bering.

3. Axborotlarga nisbatan xavf-xatarlar tasnifi.
4. Axborotlarning hayotiy davri deganda nimani tushunasiz?

Maruza № 9

Mavzu: Texnik tizimlardagi axborotlarni kriptografik himoyalash

Reja:

1. Nazariy qism bilan tanishish.
2. Shifrovchi jadval usuli yordamida matnlarni shifrlash.
3. Tayanch so'zli Sezar usuli.

Nazariy qism

Kriptografiya deb, mahfiy xabar mazmunini shifrlash, ya'ni ma'lumotlarni maxsus algoritm bo'yicha o'zgartirib, shifrlangan matnni yaratish yo'li bilan axborotga ruxsat etilmagan kirishga to'siq qo'yish usuliga aytiladi.

Kalit - kriptografiya o'zgartirishlar algoritmining ba'zi bir parametrlarining manfiy faolati bo'lib, barcha algoritmlardan yagona variantini tanlaydi. Kalitlarga nisbatan ishlatiladigan asosiy ko'rsatkich bo'lib kriptobardoshlilik hisoblanadi.

- Kriptografiya himoyasida shifrlarga nisbatan quyidagi talablar qo'yiladi:
- etarli darajada kriptobardoshlilik;
- shifrlash va qaytarish jarayonining oddiyligi;
- axborotlarni shifrlash oqibatida ular hajmining ortib ketmasligi;
- shifrlashdagi kichik xatolarga tasirchan bo'lmasligi.
- Ushbu talablarga quyidagi tizimlar javob beradi:
- o'rinlarini almashtirish;
- almashtirish;
- gammalashtirish;
- analitik o'zgartirish.

O'rinlarini almashtirish shifrlash usuli bo'yicha boshlang'ich matn belgilarining matnning ma'lum bir qismi doirasida maxsus qoidalar yordamida o'rinlari almashtiriladi.

Almashtirish shifrlash usuli bo'yicha boshlang'ich matn belgilari foydalanilayotgan yoki boshqa bir alifbo belgilariga almashtiriladi.

Gammalashtirish usuli bo'yicha boshlang'ich matn belgilari shifrlash gammasi belgilari, ya'ni tasodifiy belgilar ketma-ketligi bilan birlashtiriladi.

Taxliliy o'zgartirish usuli bo'yicha boshlang'ich matn belgilari analitik formulalar yordamida o'zgartiriladi, masalan, vektorni matritsaga ko'paytirish yordamida. Bu erda vektor matndagi belgilar ketma-ketligi bo'lsa, matritsa esa kalit sifatida xizmat qiladi.

O'rinlarni almashtirish usullari

Ushbu usul eng oddiy va eng qadimiy usuldir. O'rinlarni almashtirish usullariga misol sifatida quyidagilarni keltirish mumkin:

- shifrovchi jadval;
- sehrli kvadrat.
- Shifrovchi jadval usulida kalit sifatida quyidagilar qo'llaniladi:
- jadval o'lchovlari;
- so'z yoki so'zlar ketma-ketligi;
- jadval tarkibi xususiyatlari.

1-mashq. Shifrovchi jadval usuli yordamida matnni shifrlang.

quyidagi matn berilgan bo'lsin:

AXBOROT_TEXNOLOGIYALARINI_RIVOJLANTIRISH

Ushbu axborot ustun bo'yicha ketma – ket jadvalga kiritiladi:

A	O	E	O	A	–	Ж	И
X	T	X	Г	P	P	Л	P
Б	–	H	И	И	И	A	И
O	T	O	Я	H	B	H	III
P	E	Л	Л	И	O	T	*

Natijada, 5x8 o'lchovli jadval tashkil qilinadi.

Endi shifrlangan matn qatorlar bo'yicha aniqlanadi, ya'ni o'zimiz uchun 4 tadan belgilarni ajratib yozamiz.

$T_1 = \text{AOEO A_ЖИ XTXГ PPJP Б_ИИ ИИАИ ОТОЯ НВНШ РЕЛЛ ИОТ*}$

Bu erda kalit sifatida jadval o'lchovlari xizmat qiladi.

Hozirgi vaqtda kompyuter tarmoqlarida tijorat axborotlari bilan almashishda uchta asosiy algoritmlar, ya'ni DES, CLIPPER va PGP algoritmlari qo'llanilmoqda. DES va CLIPPER algoritmlari integral sxemalarda amalga oshiriladi. DES algoritmining kriptobardoshliligini quyidagi misol orqali ham baholash mumkin: 10 mln. AQSh dollari xarajat qilinganda DES shifrlash ochish uchun 21 minut, 100 mln AQSh dollari xarajat qilinganda esa 2 minut sarflanadi. CLIPPER tizimi SKIPJACK shifrlash algoritmini o'z ichiga oladi va bu algoritm DES algoritmidan 16 mln marta kuchliroqdir.

Sehrli kvadrat deb, katakchalariga 1 dan boshlab sonlar yozilgan, unda har bir ustun, satr va diagonal bo'yicha sonlar yig'indisi bitga songa teng bo'lgan kvadrat shaklidagi jadvalga aytiladi.

Sehrli kvadratga sonlar tartibi bo'yicha belgilar kiritiladi va bu belgilar satrlar bo'yicha o'qilganda matn hosil bo'ladi.

2-mashq. 4x4 o'lchovli sehrli kvadratni olamiz, bu erda sonlarning 880 ta har xil kombinatsiyasi mavjud. quyidagicha ish yuritamiz:

4	14	15	1
9	7	6	12
5	11	10	8
16	2	3	13

Boshlang'ich matn sifatida quyidagi matnni olamiz:

DASTURIY TILLAR

va jadvalga joylashtiramiz:

T	A	P	Д
–	И	P	Л
У	И	T	Й
*	A	C	Л

Shifrlangan matn jadval elementlarini satrlar bo'yicha o'qish natijasida tashkil topadi:

TARD _IRL UITY *ASL

Almashtirish usullari

Almashtirish usullari sifatida quyidagi usullarni keltirish mumkin:

- Sezar usuli;
- Affin tizimidagi Sezar usuli;
- Tayanch so'zli Sezar usuli va boshqalar.

Tsezar usulida almashtiruvchi xarflar k va siljish bilan aniqlanadi. Yuliy Sezar bevosita k q 3 bo'lganda ushbu usuldan foydalangan.

k = 3 bo'lganda va alifbodagi harflar m = 26 ta bo'lganda quyudagi jadval xosil qilinadi:

A	→	D
B	→	E
C	→	F

D	→	G
E	→	H
F	→	I
G	→	J
H	→	K
I	→	L
J	→	M
K	→	N
L	→	O
M	→	P
N	→	Q
O	→	R
P	→	S
Q	→	T
R	→	U
S	→	V
T	→	W
U	→	X
V	→	Y
W	→	Z
X	→	A
Y	→	B
Z	→	C

3-mashq. Matn sifatida KOMPYUTER so'zini oladigan bo'lsak, Sezar usuli natijasida quyidagi shifrlangan yozuv hosil bo'ladi: NRPSBXWHU.

Tsezar usulining kamchiligi bu bir xil harflarning o'z navbatida, bir xil harflarga almashishidir.

Hozirgi vaqtda kompyuter tarmoqlarida tijorat axborotlari bilan almashishda uchta asosiy algoritmlar, ya'ni DES, CLIPPER va PGP algoritmlari qo'llanilmoqda. DES va CLIPPER algoritmlari integral sxemalarda amalga oshiriladi. DES algoritmining kriptobardoshliligini quyidagi misol orqali xam baholash mumkin: 10 mln. AQSh dollari xarajat qilinganda DES shifrlash ochish uchun 21 minut, 100 mln

AQSh dollari xarajat qilinganda esa 2 minut sarflanadi. CLIPPER tizimi SKIPJACK shifrlash algoritmini o'z ichiga oladi va bu algoritm DES algoritmidan 16 mln marta kuchliroqdir.

PGP algoritmi esa 1991 yilda Filipp Simmerman (AQSh) tomonidan yozilgan va elektron pochta orqali kuzatiladigan xabarlarni shifrlash uchun ishlatiladigan PGP dasturlar paketi yordamida amalga oshiriladi. FGP dasturiy vositalari Internet tarmog'ida elektron pochta orqali axborot jo'natuvchi foydalanuvchilar tomonidan shifrlash maqsadida keng foydalanilmoqda.

PGP (Pretty Good Privacy) kriptografiya dasturining algoritmi kalitli, ochiq va yopiq bo'ladi.

Sinov savollari:

1. Kriptografiya deb nimaga aytiladi?
2. Kriptografiyada qanday ko'p qo'llaniladigan usullarni bilasiz?
3. Tsezar usuli qanday usul?
4. Ochiq va yopiq kalitlar xaqida aytib bering.

Maruza № 10

Mavzu: Zamonaviy dasturlash texnologiyalari. Ob'yektga yo'naltirilgan dasturlash tillari.

Reja:

1. Kompyuter dasturlari va dasturlash tillari.
2. Dasturlash tillarining rivojlanish bosqichlari.
3. Dasturlash tillarining umumiy universal xususiyatlari.
4. Ob'yektga mo'ljallangan dasturlash tillari (OMDT) va dasturlarni ishlab chiqish uchun vositalar.



Kompyuter dasturlari va dasturlash tillari

Agarda siz hech qachon biror bir kompyuter dasturini mustaqil tuzib ko'rmagan bo'lsangiz, sizga dasturni tuzish va uni o'zgartirish uchun kompyuter dasturchisi zarur bo'ladi. Shunday ekan, siz dasturchilarni turli dastur ishlab chiqishlari uchun zarur qanday axborotlar talab etilishini tushunishingiz kerak bo'ladi. Kompyuter dasturi kompyuterda masalalarni bajarilishi uchun yo'naltirilgan ko'rsatmalar ketma-ketligini o'z ichiga oladi. Kompyuter dasturchisi, ba'zan ishlab chiqaruchi ham deyiladi, ular kompyuter dasturlarini yaratadi va ularni o'zgartiradilar.

Dasturni yaratish uchun dasturchi ba'zan dasturlash tillari yordamida dastur buyruqlari yoki dasturiy kodlardan foydalanadilar. Dasturlash tillari o'z ichiga kompyuterga habar jo'natish uchun dasturchiga ruxsat beruvchi so'zlar to'plami va simvollarni oladi. Boshqa vaqtlarda dasturchilar dastur yaratish uchun dastur element(instrument)laridan foydalanadilar.

Dasturlash tillari dasturni qurish uchun qulay muhitni ta'minlaydi. Aniqrog'i, insonlar turli tillarda gapiradilar (ingliz tili, ispan tili, fransuz tili, o'zbek tili va hakazo), dasturchi ham Shunga o'xshash turli dasturlash tillari va instrumentlaridan dastur yaratishda foydalanadi(9.1- rasm).



10.1-rasm. Dasturchi dastur yaratishi uchun qaysi dasturlash tilidan foydalanishni aniqlab olishi zarur.

Bugungi kunda yuzga yaqin dasturlash tillari mavjud. Har bir til dastur tuzish uchun o'zining maxsus instruksiyasiga va dastur tuzish qoidalarga ega. Dasturlash tillari aniq maqsadga, ilmiy ilovalarga, biznes, veb-sahifa yaratish va albatta qishloq xo'jaligi masalalarini yechishga ham qaratiladi. Masalani yechishda yoki yechimni topishda dasturchi bittadan ortiq dasturlash tillarini birlashtirgan holda foydalanadi.

Dasturlash tillari ikki turda bo'ladi: Yuqori daraja va quyi daraja. Quyi daraja dasturlash tillari mashinaga (kompyuterga) bog'liq holda ishlaydi. Mashina yagona aniq biror bir turdagi kompyuter bo'lishi mumkin va bu kompyuterda quyi darajadagi dasturlash tillari ishlaydi. Bu dasturlar boshqa turdagi kompyuterlarga o'tkazish juda qiyin bo'ladi. Yuqori darajadagi dasturlash tillarining har bir buyrug'i bir qancha mashinalarga (kompyuterlarga) mos bo'ladi. Yuqori darajadagi dasturlash tillari mustaqil mashina asosida bo'ladi. Mustaqil-mashina turli xildagi kompyuterlarda va operatsion tizimlarda ishlashi mumkin.

Quyi darajadagi tillar

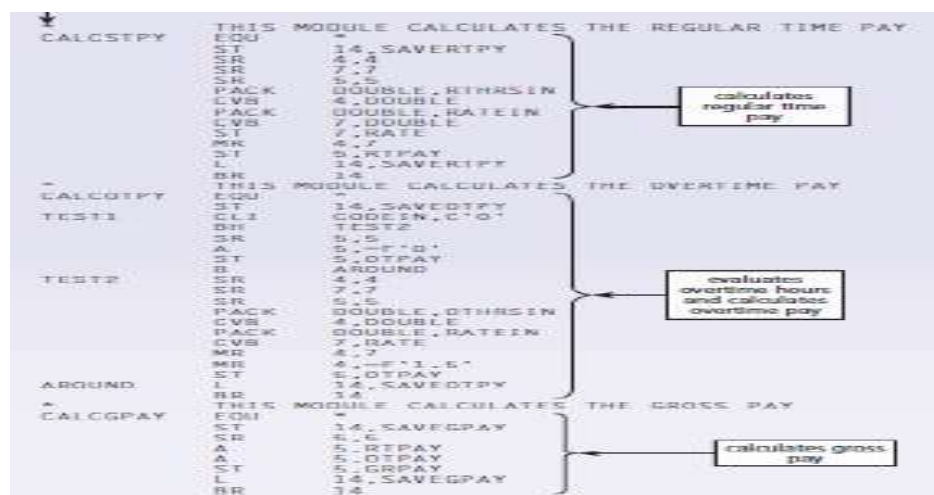
Quyi darajadagi tillar mashina tili va yig'ma tillardan iborat bo'ladi. Ma'lumki, bu mashina tili dasturlash tilining birinchi avlodi hisoblanadi, Ya'ni yagona til faqat ikkilik (0 va 1) razryadlar qatori instruksiyasidan iborat bo'lgan (9.2-rasm) yoki raqamlar yoki harflar kombinatsiyasidir.

Binar raqamlar elektr holatini o'chirish va yoqishdan iborat. O'zingiz uchun tasvirlamoqchi bo'sangiz, mashina tili bu insonga ko'p vaqt davomida charchatadigan va qiyinchilik (zo'rlik) bilan bajarish deganidir.

00000E	5A50	35AA			015AC
0000E2	47F0	2100		00102	
000102	1877				
000104	5870	304E			01050
000108	1C47				
00010A	4E50	30D6			010DB
00010E	F075	30D6	003E	010DB	0003E
000114	4F50	30D6			010DB
000118	5050	3052			01054
00011C	58E0	30B6			01088
000120	07FE				
000122	50E0	30BA			00122
000126	1855				0108C
000128	5A50	304E			
00012C	5850	3052			01050
000130	5050	305A			01054
000134	58E0	30BA			0105C
000138	07FE				0108C

10.2-rasm. Mashina tili dasturida 16 lik sanoq tizimidan foydalanib, kodiroyka qilinadi.

Assambler tili dasturlash tilining ikkinchi avlodi hisoblanadi. Dasturchi dastur instruksiyasini simvolli kodlar instruksiyasidan foydalanib tuzadi (9.3-rasm). Simvolli kodlar buyruqlari assamblerning qisqartma so'zlari asosida yozadi, masalan, S- taqqoslash uchun, L - yuklash uchun va M- ko'paytirish uchun.



10.3-rasm. Assambler dasturida dasturchi oylik ish haqi hisobini hisoblash ishlarini bajarmoqda.

Assambler tili simvolli manzillardan ham foydalanadi. Simvolli manzillar saqlash joyini aniqlaydigan muhim nomdir. Masalan, dasturchi RATE so'zidan foydalanib saqlash joyiga tayyor dasturni saqlashi mumkin.

Dasturning bu afzalligiga qaramasdan dasturni yig'ish qiyin. Undan tashqari, dasturchilar dasturni mashina tilidan assambler tiliga o'tkazishlari yoki dasturni ishga tushirishlari shart. Aniqrog'i kompyuter boshlang'ich dasturiy yig'malarini bajara olmasligi mumkin.

Dasturlar manbai bo'lib, mashina tilini o'zgartirish imkonini beruvchi til instruksiyalarini o'z ichiga oluvchi dastur hisoblanadi. Dasturchi boshlang'ich dasturni assambler mashina tilidan foydalanib o'zgartirishi mumkin. Ba'zi paytlarda tillar o'z ichiga makroslarni ham oladi. Assamblerda makros mashina tilida ko'pgina instruksiyalarni birgina instruksiya uchun yaratadi. Makroslar dasturchini dastur ishlab chiqishida uning vaqtini tejaydi.

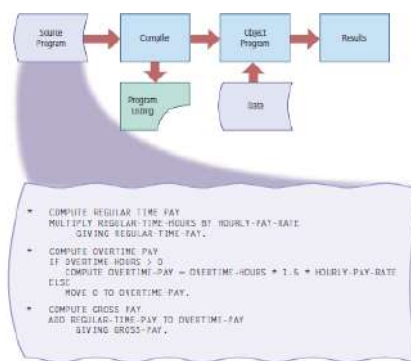
Bugungi kunda tillar asosan qurilmalarni boshqarish uchun yoki muhim vazifalarni kritik samaradorligini oshirish uchun ishlatiladi.

Protsedurali tillar. Dasturlash tili (quyi darajadagi) va mashinalarning kamchiliklari 1950-1960 yillarda protsedurali tillar ishlab chiqilganidan keyin to'g'rilandi. Protsedurali tillarda dasturchi kompyuterga habarlar qanday tarzda etib borishi va tayyorlanishini yozadi. Protsedurali tillardan dasturchi, ba'zan dasturlash tilining uchinchi avlodi (3GL)ning ingliz tilida yozilgan seriyasidan dastur instuksiyasini huddi so'zdek yozishda foydalanadi. Masalan, ADD ni kengaytirib yozsak, qo'shish yoki PRINT chop etish uchun deganidir. Ko'pgina 3GL lar arifmetik amallar, * belgisi ko'paytirish, 1 belgisi esa qo'shish amali bo'lib ham foydalaniladi. Bu inglizcha so'zlar va arifmetik belgilar dasturchiga dastur ishlab chiqish jarayonini qisqartiradi.

Assembler kabi 3GL ning kodi (instruksiyasi) dasturning boshlang'ich kodi deyiladi. Dasturchilar bu boshlang'ich dasturlarni huddi kompyuter dasturni bajarishi kabi mashina tiliga o'zgartirishlari Shart. Bu tarjima jarayoni juda qiyin kechadi. Chunki 3GL dasturning boshlang'ich kodini birgina buyrug'i mashina tilining ko'pgina buyruqlari orqali o'zgartiriladi, Ya'ni tarjima qilinadi. 3GLlar uchun dasturchilar odatda, bunday o'zgartirishlarni amalga oshirish uchun yoki kompilyatorlardan yoki interpretatorlardan foydalanadilar.

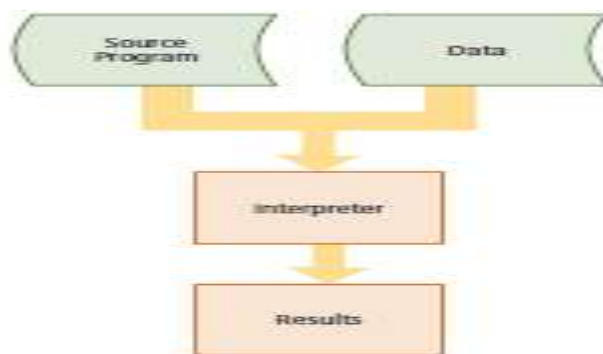
Kompilyator alohida dasturning boshlang'ich dasturini o'zgartirishni to'liqligicha mashina tilida bajarilishini o'zida ifodalaydi. Mashinaning til versiyasida 3GLning kompilyasiya natijasi ob'ekt kodi yoki dastur ob'ekti deb ataladi. Kompilyator ob'ekt kodini keyinroq tarqatish uchun saqlaydi.

10.4-rasmda boshlang'ich dasturning kompilyasiyasi ko'rsatilgan.



10.4-rasm. Kompilyator boshlang'ich dastur ob'ektini dasturning mashina tiliga o'zgartiradi.

Kompilyator barcha dasturni bajarilishidan oldin ko'chirib o'tkazadi. Interpretator esa teskarisi, bu vaqtda birgina buyruqni bajaradi va ko'chirib o'tkazadi. Interpretator kod buyruqlarini o'qiydi, uni o'zgartiradi. 179-rasmda dasturning interpretatsiya jarayoni ko'rsatilgan.



10.5-rasm. Dasturning interpretatsiya jarayoni.

Ayni paytda protsedurali tillarning yuzlab turlari mavjud. Lekin ba'zilar sanoat uchun sifat standarti bo'yicha e'tirof etiladi. Bularga C va COBOL tegishli bo'ladi.

Dasturlash tillarining rivojlanish bosqichlari.

XX asrning 50 yillar oxirida, Ya'ni 1948 yilda birinchi dasturlash tili paydo bo'lgan. Bu dasturlash tilining nomi PLANKALKUL deb nomlangan. Bu dasturlash tili dan ASSEMBLER dasturlash tili yaratilgan. 60 yillarda iqtisodiy masalalarni yechish uchun COBOL dasturlash tiliva FORTRAN dasturlash tillari yaratildi. 1964 yilga kelib boshlovchilar uchun ancha qulay bo'lgan dasturlash tili BASIC yaratildi. 1969 yilga kelib, protsedurali dasturlash tili Paskal dasturlash tili yaratildi. 1972 yilda tizimli dasturlash tili C dasturlash tili yaratildi. 1989 yilga dunyo dasturchilari tomonida ob'ektga yo'naltirilgan yondashuv natijasida C++ dasturlash tili yaratildi. Dasturlash tili oilalarga bo'linadi. Bu oilalar asosan Basic, C va Pascal oilasidir. C oilasiga kiruvchi dasturlash tillari ko'pni tashkil etadi. Quydagi jadvalda dasturlash tili oilalariga kiruvchi dasturlash tillari ko'rsatilgan.

C++ dasturlash tili C tiliga asoslangan. C esa o'z navbatida Vasic va Basic Pascal tillaridan kelib chiqqan. Basic Pascal 1967 yilda Martin Richard tomonidan tuzilgan va zamonaviy operatsion tizimlarni yozish uchun mo'ljallangan edi. Ken Tomson o'zining Vasic tilida Basic Pascalning ko'p xossalarini kiritgan va Vasicda UNIX operatsion tizimining birinchi versiyalarini yozgan. Basic Pascal ham, Vasic ham tipsiz til bo'lgan. Ya'ni o'zgaruvchilarning ma'lum bir tipi bo'lmagan – har bir o'zgaruvchi kompyuter kompyuter xotirasida faqat bir bayt joy egallagan. O'zgaruvchini qanday holda ishlatish esa, ya'ni butun sonmi, kasrli sonmi yoki harf ko'rinishdami, dasturchi vazifasi bo'lgan.

C tilini Dennis Ritchi Vasicdan keltirib chiqardi va uni 1972 yili ilk bor Bell Laboratoriyasida, DEC PDP-11 kompyuterida qo'lladi. C o'zidan oldingi Vasic va Basic Pascal tillarining juda ko'p muhim tomonlarini o'z ichiga olishi bilan bir qatorda o'zgaruvchilarni tiplashtirdi va bir qator boshqa yangiliklarni kiritdi.

Boshlanishda C asosan UNIX tizimlarida keng tarqaldi. Bu tillar tufayli so'z va gaplarning mantiqiy konsruksiyasidan foydalanib dasturlash imkoniyati yaratildi. Bu komandalarni mashina tiliga interpretatorlar va kompilyatorlar ko'chirar edi. Interpretator dasturni o'qish jarayonida uning komandalarini ketma – ket mashina tiliga o'tkazadi.

Kompilyator esa yaxlit dastur kodini biror bir oraliq forma - ob'ekt fayliga o'tkazadi. Bu bosqich **kompilyasiya** bosqichi deyiladi. Bundan so'ng kompilyator ob'ektlili faylni bajariluvchi faylga aylantiradigan dasturini chaqiradi.

Interpretatorlar bilan ishlash osonroq, chunki dastur komandalari qanday ketma – ketlikda yozilgan bo'lsa, shu tarzda bajariladi. Bu esa, dastur bajarilishini nazorat qilishni osonlashtiradi.

Kompilyator esa kompilyasiya va kompanovka kabi qo'shimcha bosqichlardan iborat bo'lganligi uchun ulardan hosil bo'ladigan bajariluvchi faylni tahlil qilish va o'zgartirish imkoniyati mavjud emas. Faqatgina kompilyasiya qilingan fayl tezroq bajariladi, chunki bundagi komandalar kompilyasiya jarayonida mashina tiliga o'tkazilgan bo'ladi.

Dasturlash tillarining umumiy universal xususiyatlari.

Dastur so'zi ham komandalarning alohida blokini (berilgan kodini) aniqlovchi so'z, ham yaxlit holdagi bajariluvchi dasturiy mahsulotni belgilovchi so'zsifatida ishlatiladi.

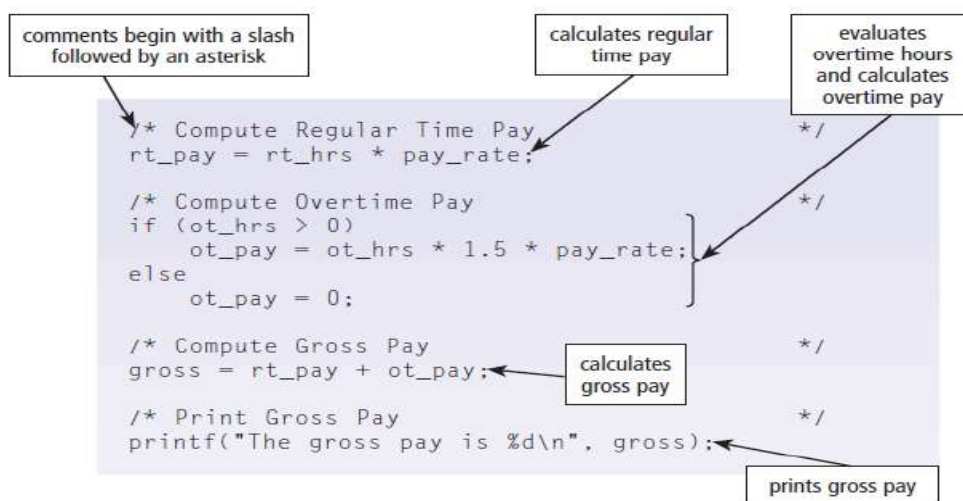
Dasturni dasturchi tomonidan yoziladigan komandalar to'plami, yoki amallar bajaradigan kompyuter mahsuloti sifatida tushinamiz.

Zamonaviy dasturlar – foydalanuvchi bilan do'stona munosabatni yuqori darajada tashkil qiladigan ko'p sondagi oynalar, menyu, muloqot oynalari va visual grafikaviy muhitlardan tarkib topgan interfeysga ega.

Dasturlashga talabni o'zgarishi nafaqat tillarning o'zgarishiga, balki uni yozish texnologiyasini ham o'zgarishiga olib keldi. Dasturlash evolyusiyasi tarixida ko'pgina bosqichlar bo'lishiga qaramay protsedurali dasturlashdan ob'ektga mo'ljallangan dasturlashga o'tildi.

C ++ dasturlash tili

C dasturlash tili 1970 yilning boshlarida Dennis Ritchi tomonidan Bell laboratoriyasida ishlab chiqildi. Bugungi kunda ko'pgina dasturlar C tilida yozilmoqda (9.6-rasm). Bu elektron jadval va matnlarga ishlov berish dasturlarini, ya'ni amaliy dasturiy vositalarni hamda operatsion tizimlarni o'zichiga oladi.

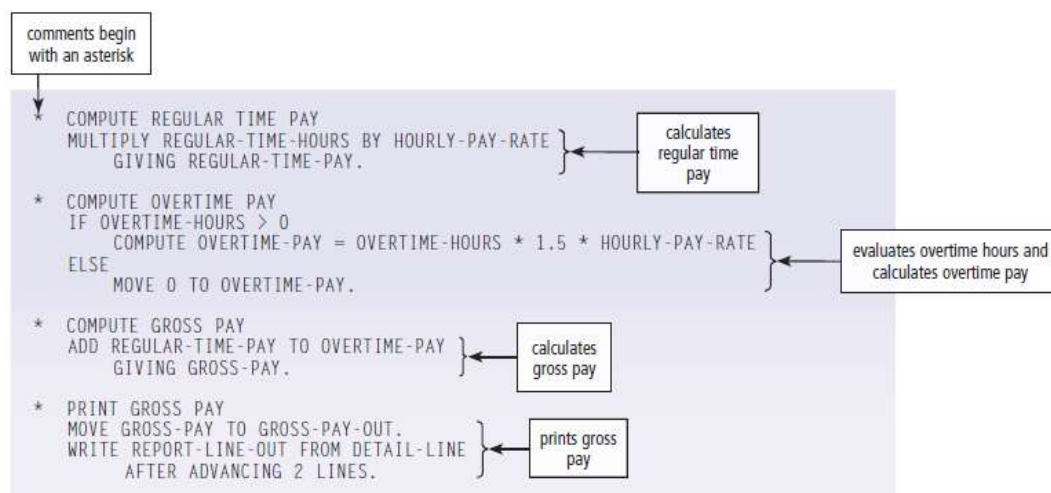


10.6-rasm. C dasturida ishlab chiqilgan dasturga misol.

C dasturi dasturlashda professional malakani talab qiladi. Ko'pchilik dasturchilar S dasturidan ilmiy muammolarni yechishda va biznesda ko'p foydalanadilar. S dasturi har qanday operatsion tizimda va har qanday kompyuterda ishlaydi. Ko'pincha C dasturlash tili UNIX i Linux operatsion tizimlarida foydalaniladi.

COBOL dasturlash tili

COBOL (Common Business-Oriented Language) dasturlash tili 1960 yil boshlarida Amerika Qo'shma Shtatlarida kuchli rivojlandi (9.7-rasm). Dengiz zobiti Greys Xopper kompyuter dasturlari muhitida COBOL tilining asoschisi hisoblanadi.



10.7-rasm. COBOL dasturida tayyorlangan dastur

COBOL dasturlash tili bu biznes-ilovalar uchun mo'ljallangan dasturdir. COBOL tili banklardagi tranzaksiya uchun hisob-kitoblarni olib borishda, ish haqlarini hisoblashda qo'llaniladi. COBOL boshqa tur kompyuterlarida ham ishlaydi.

Ob'ektga mo'ljallangan dasturlash tillari va dasturlarni ishlab chiqish uchun vositalar

Kompyuter dasturchilari ob'ektga mo'ljallangan dasturlash tillari (OMDT)dan yoki ob'ektga mo'ljallangan loyihalash uchun dasturni ishlab chiqish vositalaridan foydalanadilar. OMDTning afzalligi Shundaki, mavjud ob'ektning o'zgarishi va qayta foydalanish imkoniyati mavjud. Masalan, qachonki dasturchi biror bir ob'ekt yaratsa, u boshqa dasturlardan foydalanish uchun ham ochiq ko'rinishda bo'ladi. Shuning uchun dasturchi ushbu mavjud ob'ektdan takroran foydalanadi.

Ob'ektlar bilan ishlashning qo'shimcha imkoniyatlari bular, OMD hodisalarini boshqarishdir. Hodisalar bu ish harakatidir, Ya'ni dastur uchun javob beradi. Hodisalariga misollar o'z ichiga klaviaturada bosilgan klavishlarni oladi.

Dastur hodisalari hodisalarini tekshiradi va javob qaytaradi. Bir nechta dasturlash tillari hodisalarini boshqaradi, lekin OMD ni to'liq egallamaydi. Ba'zi dasturlar tillari esa, masalan Java, C #, F #, C ++ va Visual Basic dasturining so'ngi versiyalari to'liq ob'ektga mo'ljallangan dasturlash hisoblanadi.

OMDTni ishlab chiqarish muhitida RAD muhiti yaxshi ishlaydi. RAD (ilovalarni tezkor ishlab chiqish) o'z ichiga dasturiy ta'minotlarni ishlab chiqish uslublarini oladi.

C++ tilining egasi yo'q, u hech kimga tegishli emas. Bu til C tilidan kelib chiqqani uchun bosh harf C, ++ esa bu tildagi qiymatni bittaga ko'tarish belgisi hisoblanadi va bu belgi C++ dasturlash tilida paydo bo'lgan.

C++ dasturlash tili kompilyasiya qilinadigan til hisoblanadi, bu degani yozilgan kod oldin mashina tiliga o'giriladi va keyin ishga tushiriladi deganidir.

C++ 1980 yilda Bell Laboratories da Bern Straustrup tomonidan ob'ektga mo'ljallangan dasturlash tillari(OMDT) kabi ishlab chiqildi (9.8-rasm).

```
// portion of a C++ program that allows users to create
// a new zip code from a string or a number and expand
// zip codes, as appropriate, to a 10-digit number

ZipC::ZipC( const unsigned long zipnum )
{
    ostringstream strInt;
    strInt << zipnum;
    code = strInt.str();
}

const string ZipC::getCode()
{
    return code;
}

void ZipC::setCode(const string newCode)
{
    code = newCode;
}

void ZipC::expand( const string suffix )
{
    if(code.length() == 5 &&      // small size?
       suffix.length() == 4)     // length ok?
    {
        code += "-";
        code.append(suffix);
    }
}
```

10.8-rasm. C++ dasturiga misol.

Dasturchilar odatda, Web resurslar va ma'lumotlar bazasi ilovalarini ishlab chiqishda C++ dan foydalanadilar. Amaliy dasturiy ta'minotlar, masalan, C++ dasturida yozilgan elektron jadval va matnlar dasturini ishlab chiqishda foydalanadi.

C++dasturlash tili funksiya va ob'ektlarning juda boy kutubxonasiga ega. Ya'ni C++ tilida dasturlashni o'rganish ikki qismga bo'linadi. Birinchisi bu C++ tilini o'zini o'rganish, ikkinchisi esa C++ tilining standart kutubxonasidagi tayyor ob'ekt(funksiya)larni qo'llashni o'rganishdir.

C++ tili va ob'ektlarga mo'ljallangan dasturlash tili.

C++ tili ob'ektga mo'ljallangan dasturlash tamoyillarini qo'llab- quvvatlaydi. Bu tamoyillar quyidagilardir:

1. Inkapsulyasiya
2. Merosxo'rlik
3. Polimorfizm

Inkapsulyasiya. Agarda muxandis ishlab chiqarish jarayonida rezistorni qo'llasa, u buni yangidan ixtiro qilmaydi, omborga (magazinga) borib mos parametrlarga muvofiq kerakli detalni tanlaydi. Bu molda muxandis joriy rezistor qanday tuzilganligiga e'tiborini qaratmaydi, rezistor faqatgina zavod

xarakteristikalariga muvofiq ishlasa etarlidir. Aynan Shu tashqi konstruksiyada qo'llaniladigan yashirinlik yoki ob'ektni yashirinligi yoki avtonomligi xossasi **inkapsulyasiya** deyiladi.

Inkapsulyasiya yordamida berilganlarni yashirish ta'minlanadi. Bu juda yaxshi xarakteristika bo'lib foydalanuvchi o'zi ishlatayotgan ob'ektning ichki ishlari haqida umuman o'ylamaydi. Haqiqatan ham, xolodilnikni ishlatishda refrijecktorni ishlash prinsipini bilish Shart emas. Yaxshi ishlab chiqilgan dastur ob'ektini qo'llashda uning ichki o'zgaruvchilarining o'zaro munosabati haqida qayg'urish zarur emas.

C++ tilida inkapsulyasiya tamoyili sinf deb ataluvchi nostandart tiplarni(foydalanuvchi tiplarini) hosil qilish orqali himoya qilinadi.

Qisqacha, yashirinlik yoki ob'ektni yashirinligi yoki avtonomligi xossasi inkapsulyasiya deyiladi.

C++ tilida inkapsulyasiya tamoyili sinf deb ataluvchi nostandart tiplarni (foydalanuvchi tiplarini) hosil qilish orqali himoya qilinadi.

Noldan boshlab dastur yaratish emas, balki mavjudiga o'zgartirish qilish orqali yaratish C++ tilida ham Shunday merosxo'rlikga amal qilinadi.

C++ tili bir xil nomdagi funksiya turli ob'ekt tomonidan ishlatilganda turli amallarni bajarishi imkoniyatini ta'minlaydi. Bu funksiya va sinfning polimorfligi deb nomlanadi. Poli – “ko'p”, morfe – “shakl” degan ma'noni anglatadi. Polimorfizm – bu shaklning ko'p xilligidir.

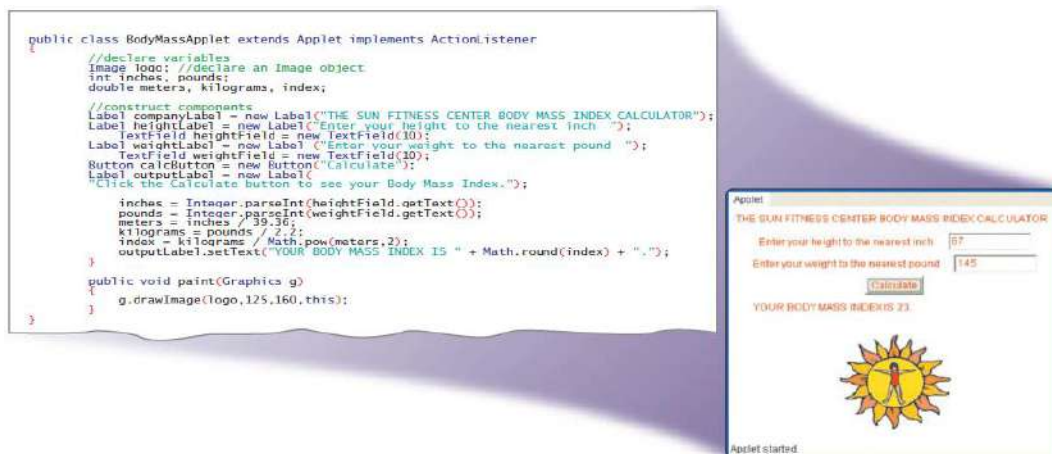
C++ dasturlash tilining buyruqlari algoritm bosqichlariga va blok-sxemalariga mos ravishda yoziladi.



10.9-rasm. C++ dasturlash tilining buyruqlari algoritmi

JAVA dasturash tili

Java ob'ektga mo'ljallangan dasturash tili hisoblanadi va bu dastur Sun Microsystems firmasida ishlab chiqariladi. 9.10-rasmda Java tilining bir qismi bo'lgan kiritish oynasi ko'rsatilgan.



10.10-rasm. Java dasturining ishchi oynasi.

Qachonki dasturchi Java dasturining kompilyasiya jarayonida ob'ekt kodi olinadi, Ya'ni ular bayt kodi deyiladi. Bu dasturlar mustaqil-mashina hisoblanadi. Dasturchilar Sun Microsystems firmasida ishlab chiqarilgan server va kompyuterlar uchun Java Platform, Standard Edition (Java SE) dasturlaridan foydalanadilar. Ulardan tashqari dasturchilar mobil qurilmalari va smartfonlar uchun maxsus dasturlar yaratishda ham ushbu dasturlardan Java Platform, Micro Edition (Java ME) foydalanadilar.

Java EE (Java Platform, Enterprise Edition) o'z ichiga dasturchi yaratayotgan Web 2.0 dan foydalanuvchi veb-ilovalarni muhiti uchun Sun Java SE firmasining texnologiyalar to'plamini oladi. Java EE ning maqsadi dastur yaratishda vaqtni qisqartirishdan iboratdir.

NET

Microsoft .NET Framework yoki .NET o'z ichiga mobil qurilmalar va avtonom ishlovchi kompyuterlar hamda ichki korporativ tarmoq yoki Internetda har qanday dasturni ishga tushiruvchi texnologiyalar to'plamini oladi. Ulardan tashqari ASP.NET o'z ichiga dinamik veb-saytlarni yaratishda veb-ilovalar strukturalarini oladi.

.NET ning afzalligi Shundaki, CLR va sinflarni o'z ichiga oladi. CLR (Common Language Runtime) turli dasturlash tillaridan foydalanish .NET dasturlarini ishlab chiqishda dasturchiga xizmat qiladi.

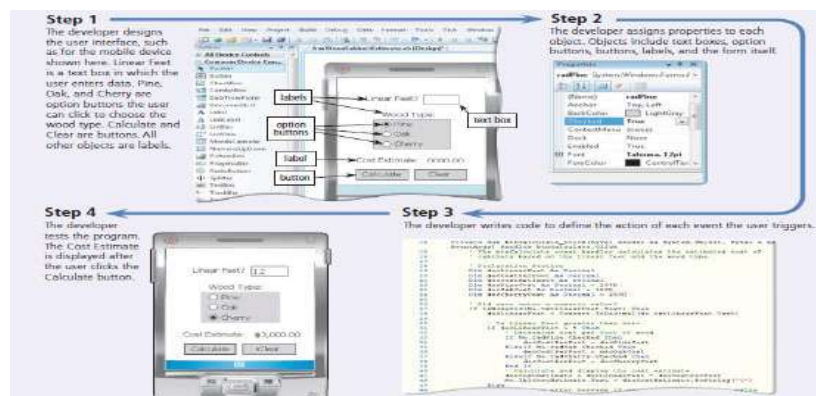
.NET va ASP.NETdan foydalanib dasturchilar Windows va veb-ilovalarni, veb-xizmatlarni engil ishlab chiqarishlari mumkin. Dasturash tillar misoliga, Ya'ni

. NETni qo'llab-quvvatlovchi C ++, C #, F #, Visual Basic, Delphi i Power Builder ni o'z ichiga oladi.

VISUAL STUDIO dasturi

Visual Studio dasturida . NETni qo'llab quvvatlovchi Windows, Windows Mobile operatsion tizimlari uchun dasturchilar maxsus dasturlar ishlab chiqadilar.

BAZALI VIZUAL DASTURLAR YARATISH



10.11-rasm. Visual Basic tilida dastur asosini yaratish.

Visual Studio dastur ishlab chiqishda vaqtni tejash uchun boshqa resurslar, maxsus til so'rovlarini LINQ (Language Integrated Query), RAD, IDE instrumentlari dasturlash tillarida xavfsizlik va ishonchlilikni yaratish ishlarini bajaradi. Masalan, Visual Studio kod fragmentlarini, shablonlarini hamda dasturlashning umumiy masalalarini o'z ichiga oladi.

Visual Studio dasturi Microsoft ofis paketlari Word, Excel, Power Point, Outlook va Project dasturlarida ishlash uchun mo'ljallangan bo'lib, o'z ichiga instrumentlar to'plamini oladi.

Visual Basic dasturlash tili dasturchiga qiyin masalalarni yechishda engilliklar yaratadi. Visual Basic BASIC dasturida asosida yaratilgan bo'lib, u 1990 yil boshlarida Microsoft Corporation da ishlab chiqilgan. Shuning uchun ham Visual Basic dasturi boshlang'ich dasturchilarga juda mos keladi.

Birinchi qadamda bazali vizual dasturni qurishda foydalanuvchiining grafik interfeysidan foydalaniladi (9.11-rasm. 1 va 2-qadamlar). Visual Basic ob'ektlari o'z ichiga turli elementlarni, tugmachalar, matn maydoni va belgilarni oladi. Keyin esa dasturchi instruksiyasini yozadi (183-rasm. 3-qadam). Va oxirida dasturchi yaratgan dasturning yakunini tekshiradi (183-rasm. 4-qadam).

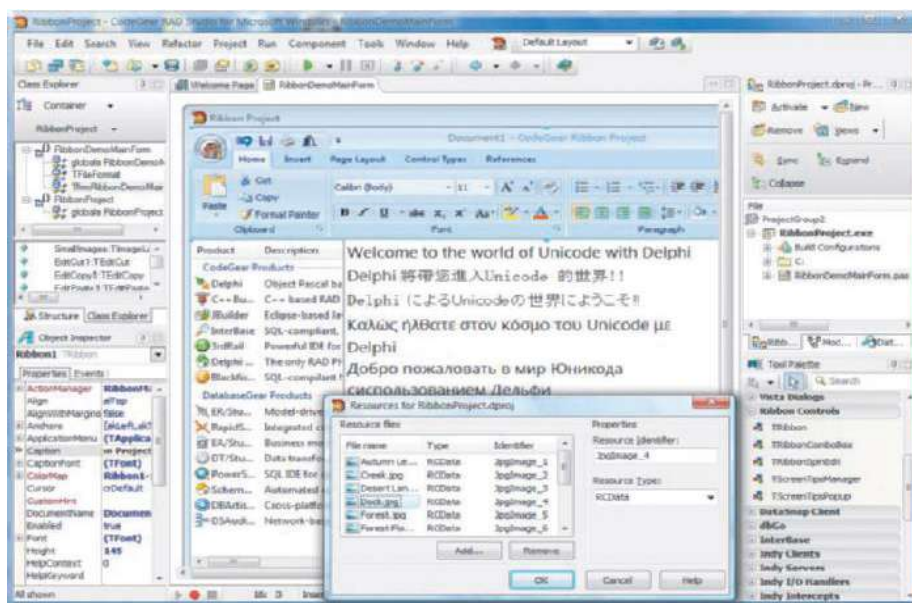
VIZUAL DASTURLASH TILLARI

Vizual dasturlash tillari boshlang'ich kodlarni yaratish uchun grafik va vizual interfeyslardan foydalanadi. Grafik interfeys dasturlashning vizual muhitida (VPE)

dasturchiga dastur ishlab chiqishida ob'ektlar shaklida xizmat qiladi. Vizual dasturlash tiliga Alisa, Mindscript va ProGraf (Alice, Mindscript, and Prograph) lar kiradi.

DELPHI dasturlash tili

Borland Delphi RAD muhitida veb-ilovalar va katta hajmli, korporativ qurilishlar uchun moskeluvchi juda kuchli dastur hisoblanadi (10.12-rasm).

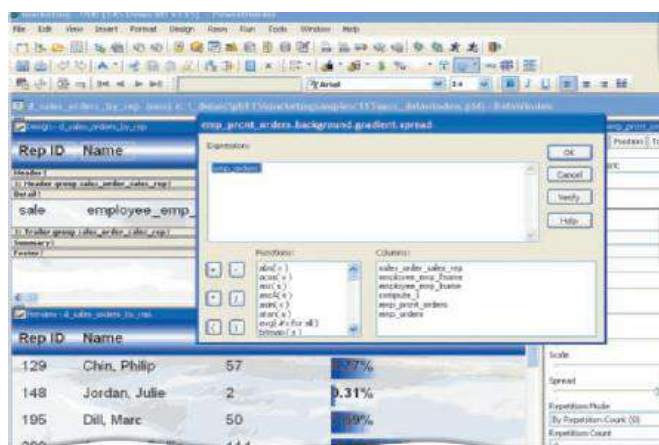


10.12-rasm. Delphining so'ngi versiyasida ishlash.

Dasturchilar Delphi dasturidan Windows, Linux va . NET platformasi uchun dasturlar yaratishda foydalanadilar.

POWER BUILDER dasturlash tili

Power Builder Sybase firmasida ishlab chiqilgan bo'lib, RAD instrumentlari rivojida juda kuchli dastur hisoblanadi. Dasturchilar Power Builderdan kichik va o'rta mijoz-server ilovalarini rivojlantirishda foydalanadilar (185-rasm). Delphi va Power Builder nuqtai nazaridan qarab, ishlash qiyinchiliklarini taqqoslash mumkin.



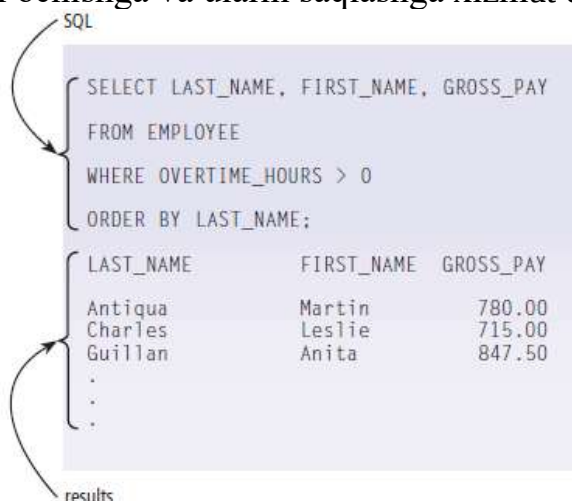
BOSHQA DASTURLASH TILLARI

1. Bu bo'limda dastur yaratishda boshqa dasturlash tillari haqida ko'rib chiqiladi.

4GLs dasturlash tili

2. 4GL (4 avlod tili) o'z ichiga ma'lumotlar bazasiga yo'l ochish bo'yicha foydalanuvchiga ruxsat berish kabi protsedurasiz tillarni oladi. Protsedurasiz tilda ma'lumotlar bazasiga grafik muhitda dasturchi ingliz tilida dasturni yozadi. Protsedurasiz tilning Asierdan foydalanish protsedurali tillarga o'xshash bo'ladi. Ko'pgina ob'ektga mo'ljallangan muhitda dasturlar tuzishda 4GLs dan foydalaniladi.

3. 4GL ning ommabop tildan biri bu SQL dir. SQL so'rovlarni boshqarish, yangilash kabi imkoniyatlarga ega dasturdir. Ya'ni ma'lumotlar bazasini boshqarish tizimi (MBBT)dir (9.14-rasm). Bu dastur ma'lumotlar bazasini boshqaruvchi kuchli strukturaviy boshqaruvga ega dastur. Bular bundan tashqari foydalanuvchiga ma'lumotlar bazasiga yo'l ochishga va ularni saqlashga xizmat qiladi.



10.14-rasm. SQL da so'rovlarni boshqarish

Dasturchilar uchun SQLni bilish nima uchun zarur?

Tuzilgan dasturlar orqali ma'lumotlar bazalari bilan bevosita muloqot o'rnatilishi kerak. Shuning uchun, dasturchilar SQL ni bilishi kerak.

Classic Programming Languages

Classic dasturlash tillari

1. ALGOL – starukturalangan protsessual dastlabki algoritmik til
2. APL – raqamlarli jadvallarni boshqarishga mo'ljallangan ilmiy dasturlash tili

3. BASIC – Djonom Kemeni va Tomasom Kursa tomonidan ishlab chiqilgan algoritmik til bo'lib, unda oddiy, interaktiv muammolar hal qilinadi

4. Forth- S tiliga o'xshash, katta bo'lmagan kompyuterlar uchun mo'ljallangan

5. FORTRAN-ilmiy ilovalar uchun ishlatiladigan oliy darajadagi ilk dasturlash tillaridan biri

6. HyperTalk-Apple tomonidan ishlab chiqilgan ob'ektga yo'naltirilgan dasturlash tili bo'lib, bu tekst, grafik va ovozlarni o'z ichiga olgan kartalarni manipulyasiya qilishga mo'ljallangan.

7. Logo – balalarning muammolarni va dasturlashtirishni o'rganish uchun ishlatiladigan ta'lim vositasi

8. Modula-2- Paskal tili tizimini rivojlantirish uchun ishlatiladigan dasturiy ta'minot

9. Pascal – dastlabki hisoblash mashinalari muhitida fransuz matematigi Blez Paskal tomonidan dasturlashtirishning strukturali konsepsiyasini talabalarga o'rgatish uchun ishlab chiqilgan tildir.

10. PL/1- FORTRAN va COBOL dasturlash tillarining ko'plab xususiyatlarini mujassamlashtirgan, biznes va ilmiy sohalarda qo'llaniladigan dasturlash tilidir.

11. Prolog- sun'iy intellekt ilovalarini ishlab chiqish uchun foydalaniladigan mantiqiy dasturlashtirish.

12. Smalltalk – ob'ektga yo'naltirilgan dasturlash tili

Dasturlash tillarining oilasi

Basic oilasi	C oilasi	Pascal oilasi
Basic	C/C++	Pascal
Visual Basic	C#	Delphi
VBScript	Java	
	JavaScript	

Dunyo reytingi bo'yicha yuqorida keltirilgan dasturlash tillaridan birinchi o'rindan uchinchi o'ringacha soilasituradi, ya'ni Java, C/C++, C# dasturlash tillari egallar ekan.

Foydalaniladigan adabiyotlar ro'yxati

1. Misty E Vermaat, Susan L Sebok, Steven M Freund. Discovering Computers (C)2016 (2016 edition). Textbook. USA, 2016

2. 2016 CppStudio - Программирование для начинающих на C++

Nazorat savollari:

1. Dasturlash tillari qanday paydo bo'lgan?
2. Dasturlash tillarining turlari?
3. Ob'ektga mo'ljallangan dasturlash deganda nimani tushinasiz?
4. C++, Java tillari haqida ma'lumot bering.

Ma'ruza № 11

Mavzu: C++ dasturlash tilining asosiy konstruksiyalari va tizimda qo'llash xususiyatlari

Reja:

1. C++ dasturlash tilining asosiy konstruksiyalari, ulardan foydalanish xususiyatlari.
2. Operatorlar, toifalar, protseduralar.
3. Dasturlar loyihasi tuzilmasi.

Tayanch iboralar: *O'zgarmaslar, o'zgaruvchilar, tayanch turlar, turlangan o'zgarmaslar, sanab o'tiluvchi turlar, turni boshqa turga keltirish.*

O'zgarmaslar.

O'zgarmas (literal) – bu fiksirlangan sonni, satrni va belgini ifodalovchi leksemadir.

O'zgarmaslar beshta guruhga bo'linadi – butun, haqiqiy (suzuvchi nuqtali), sanab o'tiluvchi, belgi (literli) va satr («string», literli satr).

Kompilyator o'zgarmasni leksema sifatida aniqlaydi, unga xotiradan joy ajratadi, ko'rinishi va qiymatiga (turiga) qarab mos guruhlarga bo'ladi.

Butun o'zgarmaslar. Butun o'zgarmaslar quyidagi formatlarda bo'ladi:

- o'nlik son;
- sakkizlik son;
- o'n oltilik son.

O'nlik o'zgarmas 0 raqamidan farqli raqamdan boshlanuvchi raqamlar ketma-ketligi va 0 hisoblanadi: 0; 123; 7987; 11.

Manfiy o'zgarmas – bu ishorasiz o'zgarmas bo'lib, unga faqat ishorani o'zgartirish amali qo'llanilgan deb hisoblanadi.

Sakkizlik o'zgarmas 0 raqamidan boshlanuvchi sakkizlik sanoq sistemasi (0,1,...,7) raqamlaridan tashkil topgan raqamlar ketma-ketligi:

023; 0777; 0.

O‘n oltilik o‘zgarmas Ox yoki OX belgilaridan boshlanadigan o‘n; oltilik sanoq sistemasi raqamlaridan iborat ketma-ketlik hisoblanadi:

Ox1A; OX9F2D; 0x23.

Harf belgilar ixtiyoriy registrlarda berilishi mumkin.

Kompilyator sonning qiymatiga qarab unga mos turni belgilaydi. Agar tilda belgilangan turlar dastur tuzuvchini qanoatlantirmasa, u oshkor ravishda turni ko‘rsatishi mumkin. Buning uchun butun o‘zgarmas raqamlari oxiriga, probelsiz l yoki L (long), u yoki U (unsigned) yoziladi. Zarur hollarda bitta o‘zgarmas uchun bu belgilarning ikkitasini ham ishlatish mumkin:

451u, 012U1, OxA2L.

Haqiqiy o‘zgarmaslar. Haqiqiy o‘zgarmaslar – suzuvchi nuqtali son bo‘lib, u ikki xil formatda berilishi mumkin:

- o‘nlik fiksirlangan nuqtali formatda. Bu ko‘rinishda son nuqta orqali ajratilgan butun va kasr qismlar ko‘rinishida bo‘ladi. Sonning butun yoki kasr qismi bo‘lmasligi mumkin, lekin nuqta albatta bo‘lishi kerak. Fiksirlangan nuqtali o‘zgarmaslarga misollar: 24.56; 13.0; 66.; .87;

- eksponensial shaklda haqiqiy o‘zgarmas 6 qismdan iborat bo‘ladi:

- 1) butun qismi (o‘nli butun son);
- 2) o‘nli kasr nuqta belgisi;
- 3) kasr qismi (o‘nlik ishorasiz o‘zgarmas);
- 4) eksponenta belgisi ‘e’ yoki ‘E’;
- 5) o‘n darajasi ko‘rsatkichi (o‘nli butun son);
- 6) qo‘shimcha belgisi (‘F’ yoki f, ‘L’ yoki ‘l’).

Eksponensial shakldagi o‘zgarmas sonlarga misollar:

1e2; 5e+3; .25e4; 31.4e-1 .

Belgi o‘zgarmaslar. Belgi o‘zgarmaslar qo‘shirnoq (‘,’-apostroflar) ichiga olingan alohida belgilardan tashkil topadi va u char kalit so‘zi bilan aniqlanadi. Belgi o‘zgarmas uchun xotirada bir bayt joy ajratiladi va unda butun son ko‘rinishidagi belgining ASCII kodi joylashadi. Quyidagilar belgi o‘zgarmaslarga misol bo‘ladi:

‘e’, ‘@’, ‘7’, ‘z’, ‘w’, ‘+’, ‘sh’, ‘*’, ‘a’, ‘s’.

1.1-jadval. C++ tilida escape –belgilar jadvali

escape belgilari	Ichki kodi (16 son)	Nomi	Belgining nomlanishi va unga mos amal

\\	0x5C	\	Teskari yon chiziqni chop etish
\'	0x27	'	Apostrofni chop etish
\"	0x22	"	Qo'shtirnoqni chop etish
\?	0x3 F	?	So'roq belgisi
\a	0x07	be1	Tovush signalini berish
\b	0x08	bs	Kursorni 1 belgi o'rniga orqaga qaytarish
\f	0x0C	ff	Sahifani o'tkazish
\n	0x0A	lf	Qatorni o'tkazish
\r	0x0D	cr	Kursorni ayni qatorning boshiga qaytarish
\t	0x09	ht	Kursorni navbatdagi tabulyatsiya joyiga o'tkazish
\v	0x0D	vt	Vertikal tabulyatsiya (pastga)
\000	000		Sakkizlik kodi
\xNN	0xNN		Belgi o'n oltilik kodi bilan berilgan

Ayrim belgi o'zgarmaslar '\ ' belgisidan boshlanadi, bu belgi birinchidan, grafik ko'rinishga ega bo'lmagan o'zgarmaslarni belgilaydi, ikkinchidan, maxsus vazifalar yuklangan belgilar – apostrof belgisi, savol belgisini (?), teskari yon chiziq belgisini (\) va ikkita ko'shtirnoq belgisini (") chop qilish uchun ishlatiladi. Undan tashqari, bu belgi orqali belgini ko'rinishini emas, balki oshkor ravishda uning ASCII kodi sakkizlik yoki o'n oltilik shaklda yozish mumkin. Bunday belgidan boshlangan belgilar escape ketma-ketliklar deyiladi (1.1-jadval).

C++ tilida qo'shimcha ravishda wide harfli o'zgarmaslar va ko'p belgili o'zgarmaslar aniqlangan.

Wide harfli o'zgarmaslar turi milliy kodlarni belgilash uchun kiritilgan bo'lib, u shar _t kalit so'zi bilan beriladi, hamda xotirada 2 bayt joy egallaydi. Bu o'zgarmas L belgisidan boshlanadi:

L'013'022' , L/cc'

Ko'p belgili o'zgarmas turi int bo'lib, u to'rtta belgidan iborat bo'lishi mumkin: 'abs', '\001\002\003\004'.

Satr o'zgarmaslar. Ikkita qo'shtirnoq ("") ichiga olingan belgilar ketma-ketligi satr o'zgarmas deyiladi:

"bu satr o'zgarmas va uning nomi string\n"

Satr ichida escape ketma-ketligi ham ishlatilishi mumkin, faqat bu ketma-ketlik apostrofsiz yoziladi.

Probel bilan ajratib yozilgan satrlar kompilyator tomonidan yagona satrga ulanadi (konkantenatsiya):

“satr –bu belgilar massivi” /* bu satr keyingi satrga qo‘shiladi */ “, uning turi char[]”;

Bu yozuvga

“satr –bu belgilar massivi, uning turi char[]”;

yozuvi bilan ekvivalent hisoblanadi.

Uzun satrni bir nechta qatorga yozish mumkin va buning uchun qator oxirida ‘\’ belgisi qo‘yiladi:

“Kompilyator har bir satr uchun kompyuter xotirasida \ satr uzunligiga teng sondagi baytlardagi alohida \ xotira ajratadi va bitta -0 qiymatli bayt qo‘shadi”;

Yuqoridagi uchta qatorda yozilgan satr keltirilgan. Teskari yon chiziq (‘\’) belgisi keyingi qatorda yozilgan belgilar ketma-ketligini yuqoridagi satrga qo‘shish kerakligini bildiradi. Agar qo‘shiladigan satr boshlanishida probellar bo‘lsa, ular ham satr tarkibiga kiradi.

Satr xotirada joylashganda uning oxiriga ‘\0’ (0 kodli belgi) qo‘shiladi va bu belgi satr tugaganligini bildiradi. Shu sababli satr uzunligi, uning «haqiqiy» qiymatidan bittaga ko‘p bo‘ladi.

Berilganlar turlari va o‘zgaruvchilar.

Dastur bajarilishi paytida qandaydir berilganlarni saqlab turish uchun o‘zgaruvchilar va o‘zgarmaslardan foydalaniladi. O‘zgaruvchi-dastur obyekti bo‘lib, xotiradagi bir nechta yacheykalarni egallaydi va berilganlarni saqlash uchun xizmat qiladi. O‘zgaruvchi nomga, o‘lchamga va boshqa atributlarga – ko‘rinish sohasi, amal qilish vaqti va boshqa xususiyatlarga ega bo‘ladi. O‘zgaruvchilarni ishlatish uchun ular albatta e‘lon qilinishi kerak. E‘lon natijasida o‘zgaruvchi uchun xotiradan qandaydir soha zahiralanadi, soha o‘lchami esa o‘zgaruvchining konkret turiga bog‘liq bo‘ladi. Shuni qayd etish zarurki, bitta turga turli apparat platformalarda turlicha joy ajratilishi mumkin.

O‘zgaruvchi e‘loni uning turini aniqlovchi kalit so‘zi bilan boshlanadi va ‘=’ belgisi orqali boshlang‘ich qiymat beriladi (shart emas). Bitta kalit so‘z bilan bir nechta o‘zgaruvchilarni e‘lon qilish mumkin. Buning uchun o‘zgaruvchilar bir-biridan ‘,’ belgisi bilan ajratiladi. E‘lonlar ‘;’ belgisi bilan tugaydi. Uzgaruvchi nomi 255 belgidan oshmasligi kerak.

Nazorat savollari.

1.O‘zgarmaslar.

2. Mantiqiy mulohazalar ustida amallar.
3. Sanab o'tiluvchi o'zgarmlar.
4. C++ tilida bir turni boshqa turga keltirish.
5. Satr o'zgarmlar.
6. C++ tilining tayanch turlari.
7. Void turi.
8. Butun o'zgarmlar.
9. Haqiqiy o'zgarmlar.

Ma'ruza №12

Mavzu: C++ dasturlash tilida oddiy dastur tuzilishi.

Reja:

1. Dasturlash muhiti bilan tanishish.
2. C++ dasturlash tilida oddiy dastur tuzilishi taxlili.
3. Dasturlashda xatoliklarni aniqlash.

Bugungi kunda kompyuter dasturlari juda ham murakkabdir va u juda ham oddiy jarayonlardan tashkil topganiga ishonish qiyindir.

Odatiy jarayonlar quyidagilar bo'lishi mumkin:

- Ekrandagi shu joyda qizil nuqta qo'yadi.
- Shu ikki raqamni qo'shadi.
- Agar qiymat salbiy bo'lsa, dastur boshqa aniq ko'rsatmalarda ishlaydi.

Kompyuter illyuzion aniq bog'lanishlarni katta tezlikda bajaruvchi ulkan miqdordagi amallardan iboratdir.

Kompyuter dasturlarini dizayn qilish va uni ijrosini ta'minlash dasturlash deyiladi. Kompyuter o'yinlarini harakat va ovoz effektlari va fantaziyali va rasmlil so'z jarayoni bilan yaratish kabi murakkab vazifa, jamoaviy yuqori malakali dasturchilarni talab qiladi. Sizning ilk dasturlashdagi muvaffaqiyatingiz maroqsiz bo'lishi mumkin. Bu modulda siz o'rganadigan tushuncha va ko'nikmalar asosiy fundament bo'ladi va sizning ilk yaratagan dasturingiz siz bilgan dasturlar bilan raqobatlasha olmasa xafsalangiz pir bo'lmaydi.

Dasturlash muhiti bilan tanishish.

Ko'p studentlar dasturchilarga kerak bo'ladigan qurilmalar ularga tanish bo'lgan dasturiy ta'minot qurilmalaridan farq qilishini bilishadi. Siz alohida vaqt

ajratib dasturiy muhit bilan tanishib chiqing. Chunki kompyuter tizimi keng farqlanadi, bu modul sizga amal qilishingiz kerak bo'lgan bosqichlar yo'riqnomasini beradi. Amaliy laboratoriya ishlarida qatnashish yoki bilimga ega do'stingiz yo'rig'ini tinglash ham samaralidir.

Bosqich 1. C++ yaratish muhitini boshlang. Kompyuter tizimlari bu borada katta farq qiladi. Ko'p kompyuterlarda yozish va dasturlarni sinash mumkin bo'lgan bir integratsiya ishlab chiqish muhiti bor. Boshqa kompyuterlarda esa C++ yo'riqnomalarini kiritish uchun avval so'z muharririni ishga tushirish lozim, keyin esa konsol oynasini ochish va bajarilishi kerak bo'lgan buyruqni kirgazishingiz kerak. Siz muhit bilan ishlashni o'rganishingiz lozim.

Bosqich 2. Oddiy dastur yozing

Yangi dasturlash tilini yaratishdagi ilk dasturda ekranda oddiy salomlashuv so'zi "Hello, World!" Keling, bu an'anaga amal qilaylik. Bu yerda C++ "HELLO , World!" dasturi:

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main()
6 {
7     cout << "HELLO , World!" << endl;
8     return 0;
9 }
```

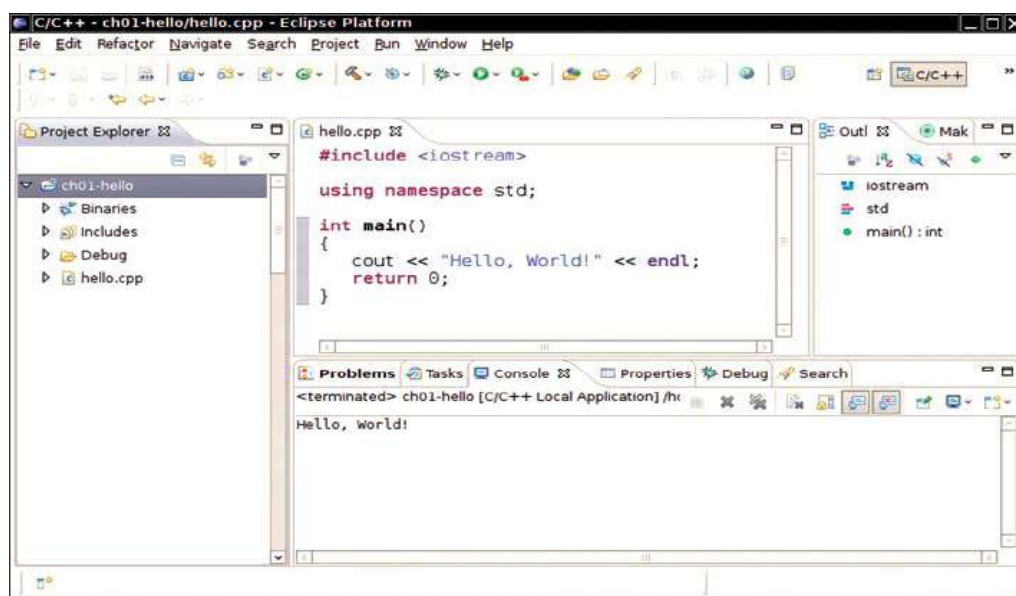
Biz keyingi bo'limda ushbu dasturni ko'rib chiqamiz. Qanday dasturiy muhitdan foydalanmang, dastur holatini muharrir oynasiga kiritishdan boshlaysiz. Yangi fayl oching va uni **Hello.** srr deb nomlang, yo'riqnomangiz uchun zarur bosqichlardan foydalaning. (Agar sizning muhitingiz loyiha nomini fayl nomiga qo'shib yozishni talab qilsa u holda siz **Hello** so'zini loyiha nomi deb olinng.) Dastur buyruqlarini tepada berilgandek aniq kirgizing. Shu bilan bir qatorda, dasturdagi manbaa fayllarini elektron nusxasini toping va muharriringizga qo'ying.

Siz shu dasturni yozar ekansiz, turli simvollarga yaxshilab e'tibor bering, C++ nozik ish sanaladi. Siz xarflarni dastur satrida ko'ringanidek bosh va kichik xarflarda kirgizishingiz shart. main yoki endl yoza olmaysiz . Agar siz e'tiborli bo'lmasangiz, 16 betdagi 1. 2 Odatiy Xatoga yo'l qo'yasiz.

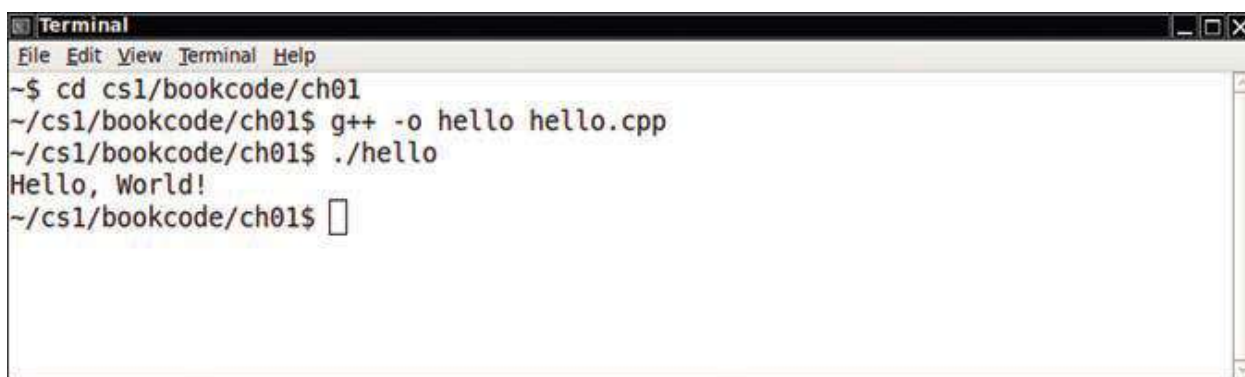
Bosqich 3. Kompilyatsiyalash va dasturni ishga tushirish.

C++ dasturini yaratish va ishga tushirish jarayoni sizning dasturlash muhitingizga bog'liq. Ayrim kompleks rivojlantirish muhitida, siz oddiygina tugmachani bosasiz. Boshqa muhitda esa, siz buyruqlar kirgizishingiz lozim. Siz dasturingizni sinovdan o'tkazishda Hello world! ekranni qaysi joyidadir paydo bo'ladi.

Eslatma1. C++ muxarririda matnni kiritish va o'zgartirish C++ da nozik jarayonlar. Siz kata va kichik xarflarni tanlab yozishda etiborli bo'lishingiz kerak. Kompilyator C++ dasturini mashina tiliga o'girib beradi.



12.1-rasm. Hello dasturini kompleks rivojlantirish muxitida ishga tushirish jarayoni



12.2-rasm. hello dasturini window oynasida kompilyatsiyalash va ishga tushirish.

Dasturingiz tuzilayotganda uning negizida nima borligi muhimdir. Birinchidan, kompilyator C++ manbaa kodini (siz kiritgan so'zlar) mashina ko'rsatmalariga o'girib beradi. Mashina kodi siz yozgan so'zni kodga o'girib berilgan shaklidir. Dasturni faqatgina ishga tushirish yetarli bo'lmaydi. Kompyuter oynasida bir tizimni ko'rinishi uchun ham, kam darajada bo'lsa ham faollik muhim. C++ rivojlantirish muhiti amaliyotchilari **cout** va uning vazifalarini o'z ichiga olgan kutubxona bilan ta'minlab beradi. **Kutubxona** bu boshqa bir inson tomonidan dasturlangan va tarjima qilingan, siz qo'llashingiz uchun tayyor dastur kodlar va jamlanmasidir. (Qo'shimcha murakkab dasturlar 1 dan ortiq mashina kodi fayli va kutubxonadan iborat). **Birlashtirgich** nomli dastur mashina kodini va C++ kutubxonasidan kerakli qismlarni birlashtirib amalga oshgan faylni yuzaga keltiradi. (10.2- rasmda bu boshqichlarni ko'rinishi tsivirlangan).

Amalga oshgan fayl kompyuter tizimiga bog'liq ravishda hello. yexe yoki hello deb nomlanadi. Siz amalga oshirgan faylni C++ rivojlantirish dasturidan chiqib ketganingizda ham ishga tushirishingiz mumkin.

Dasturchi sifatida, siz dastur tuzasiz, sinaysiz va takomillashtirasiz. Siz dasturlaringizni *Fayllarda* da saqlaysiz. Fayllarni nomlari va qonuniy nomlari qoidalar tizimi biri ikkinchisidan farq qiladi. Ayrim tizimlar fayl nomlari orasida bo'sh joy qoldirishga ruxsat beradi ayrimlari esa yo'q. Ayrimlarida katta va kichik harflar farqlanadi, ayrimlarida esa yo'q. Ko'p C++ kompilyatorlar C++ fayllari .crr, .cxx, .cc, yoki .C bilan tugashi talab qilinadi, masalan test.crr. Fayllar **papkalar** yoki **kataloglarda** saqlanadi. /Files are stored in **folders** or **directories**. / Papkalar o'z ichiga fayllarni oladi va yana ichida fayllar va papkalar bor boshqa fayllarni ham saqlay oladi. (10.3-rasm) Bu iyerarxia katta bo'lishi mumkin, va siz uning barcha bo'limlari bilan tanish bo'lmasligingiz mumkin.

Muxarrir - Manba fayli oshirilgan dastur – Kompilyator - Mashina kodi – Bog'lovchi - Amalga oshirilgan dastur - Kutubxona fayli.





12.3-rasm. Iyerarxiya fayli

Biroq siz ishingizni tashkil qilishingiz uchun papkalar yaratishingiz mumkin. Har bir dasturlash darsi uchun alohida papka ochish to'g'ri fikrdir. Papka ichida esa, xar bir topshiriq uchun alohida papka oching. Ayrim dasturlash muhiti papkani o'zingiz joylashtirmasangiz standart manzilga saqlaydi. U holda siz papka joylashgan joyini aniqlashingiz kerak bo'ladi. Siz iyerarxia papkasida fayligiz qayerga joylashganini bilishingizga ishonch hosil qiling. Bu ma'lumot papkalarni guruhlayotganingizda va zahira nusxa olayotganingizda muhimdir.

Siz ko'p vaqtingizni C++ dasturini yaratish va takomillashtirishga sarf qilasiz. Bunda ayrim fayllar kompyuter nosozligi uchun o'chib ketishi mumkin.

Qaytadan yo'qolgan fayllarni tiklamaslik uchun ishlaringizni xotira kartasiga yoki boshqa kompyuter xotirasiga zahira nusxa olib qo'ying.

Ilk dasturni taxlil qilish.

Bu bo'limda ilk C++ dasturini batafsil tahlil qilamiz. Quyida yana manbaa

kodi:

ch01/hello. cpp

Sintaksis C++ dasturi

```
#include <iostream>
using namespace std;
int main()
{
```

```
cout << "Hello, World!" << endl;  
return 0;  
}
```

Birinchi qatordagi,

#include <iostream> “kiritish/chiqarish oqimlarini” kompilyator tomonidan chaqirilishini ifodalaydi . Xozircha siz xar bir dasturda kiritish va chiqarishni amalga oshirish uchun ushbu qatorni kiritishni unutmasligingiz lozim. Keyingi qatorda, using namespace std; kompilyatorga "standart nomlanish joyi" ishlatish kerakligini aytadi.

Nomlanish joyi katta dasturlarda nomlanishdagi nizolarni bartaraf etadi. Siz nomlanish joyidan xavotir olmasangiz ham bo'ladi. Bu kitobda ishlab chiqadigan dasturingiz uchun standart nom qo'yasiz. xar bir yozadigan dasturingiz boshiga

using namespace std; ni yozing va oxirida esa #include ko'rsatmasini yozing.
Tuzilishi

```
int main()  
{  
...  
return 0;  
}
```

main deb nomlanuvchi *funksiyasi* butun sonni 0 qiymatda qaytaradi (bunda C++ int da deb nomlanuvchi bo'laklanmagan butun son) Bu qiymat dastur muvaffaqiyatli yakunlanganini ifodalaydi.

Funksiya bu belgilangan vazifani bajaruvchi dasturlash yo'riqnomasidir. Har bir C++ dasturini asosiy funksiyasi bor. Ko'p C++ dasturlaridan tashqari boshqa funksiyalarga ham ega buni keyinroq funksiyalar va qiymatlarda ko'rib chiqamiz.

Xozir esa, oddiy dastur yozish uchun kerak bo'ladigan qismlarni tahlil qilsak to'g'ri bo'ladi. Kodni amalga oshirmoqchi bo'lgan main funksiyasiga kiriting. (C++ dasturining asosiy tuzilishi Sintaksis da berilgan.) Qiymatni ekranda ko'rsatish uchun cout nomli asos va << operator(ayrim paytlarda o'rnatilgan operator)) dan foydalanishingiz mumkin. Masalan,

```
cout << 39 + 3;
```

42 soni ekranda ko'rinadi.

Eslatma 2. *Har bir C++ dasturi main funksiyaga ega. Qiymat ekranda ko'rinishi uchun cout operatoridan foydalaning.*

Sintaksiz Chiqarish operatori

```
cout << "Natija" << 6 * 7 << endl;
```

Xatolar

Dasturlash tili juda ham muhim konvensialarga asoslanadi. Siz bir inson bilan muloqot qilayotganingizda 1 yoki 2 so'zni o'tkazib yuborsangiz yoki tushirib qoldirsangiz siz bilan muloqotdagi inson nima demoqchi ekanligingizni tushuna oladi. Ammo C++ da xatoga yo'l qo'ysangiz kompilyator noto'g'ri tushunchani qabul qilmaydi. (Bu xasqiqatda yaxshi narsa, agar kompilyator noto'g'ri tushunchani qabul qilganda, u natijani ham noto'g'ri taqdim qilar edi. Bu esa falokatli oqibatlarga olib kelar edi.) Bu bo'limda siz dasturingizdagi xatolarni qanday bartaraf etishni o'rganasiz. hello. cpp dasturi bilan tajriba qilamiz. Biz quyidagi xatolarga yo'l qo'ysak nima sodir bo'lar edi.

Birinchi holatda, kompilyator orqali nimani nazarda tutayotganingiz-ni tushunmaganligidan arz qiladi. Yo'l qo'yilgan xatolikning aniq ta'rifi kompilyatorga bog'liq bo'ladi. Bu "Undefined symbol cot" kabi ko'rinishda ham bo'lishi mumkin. Bu COMPILETIME xatolik yoki sintaksis xatolik sanaladi. Imlo qoidasiga yo'l qo'yilsa ham kompilyator uni topadi. Agar kompilyator 1 yoki undan ko'p xatolikni topsa, u holda dasturni texnika tiliga o'girmaydi va natijada ishga tushiriladigan dastur ham yaratilmaydi. Siz xatolikni bartaraf etishni boshqatdan to'plashingiz kerak bo'ladi. Odatda ilk muvaffaqiyatli kompilyatsiya erishishdan oldin COMPILETIME xatolikni bartaraf etishdagi bir qancha jarayonlardan o'tiladi.

Agar kompilyator xatolikni aniqlasa u osonlikcha to'xtamaydi va rad etmaydi. U aniqlagan xatolikni hammasini ko'rsatadi, siz bunda hamma xatolikni bittada to'g'irlab olishingiz mumkin. Ayrim paytlarda bita xatoni o'zi ham dasturni ishdan chiqarishi mumkin. Bunday xatolik ikkinchi satrda ham uchrashi mumkin. Dasturchi yopuvchi qavs belgisini esdan chiqarsa kompilyator satr oxirini qidirishda davom etadi. Bunday holatlarda kompilyator qo'shni qatorlarda soxta xatolikni ko'rsatadi. Siz kerakli qatorlardagi xatolikni to'g'irlab so'ng qaytadan kompilyatsiyalashingiz kerak.

3chi satrdagi xatolik boshqacha bo'ladi. Dastur kompilyatsiyalanadi va ishga tushadi ammo chiqarilgan ma'lumot noto'g'ri bo'ladi. RUN TIME ERROR sabali dastur mantiqiy nuqson aniqlaydi va bu nuqsonlar **mantiqiy** xatolik deyiladi. Ayrim RUN TIME ERROR xatoliklari jiddiy hisoblanadiki ayrim istisnalar keltirib chiqarishga ham sababchi bo'ladi. Protsessordagi xabar xato xabar sababli dasturni tugatilishiga olib keladi. Misol uchun dasturingiz cout << 1 / 0; bo'lsa "nol bilan ajratish" istisnosi bilan tugatilinadi.

Dastur rivojlantirish mobaynida, xatolikdan qoshish qiyin. Agar dastur bir necha qatordan ko'p bo'lsa, kirgizish uchun aqlbovar qilmas diqqat e'tiborni talab qiladi. Siz vergul va qavslarni o'zingiz bilmagan tarzda tushirib qoldirasiz, lekin kompilyator uni sizga bu muammolarni qidirib topib beradi. RUN TIME ERROR muammolidir. Kompilyator ularni topa olmaydi, kompilyator gap tuzilishi xatosiz bo'lsa, istalgan dasturni tarjima qila oladi- lekin natijadagi dastur xato bo'ladi. Dastur muallifi RUN TIME eRROR topishga va dasturni tekshirishga ma'suldir. Dastur tekshiruvi ushbu kitobda muhim matnlardan biridir.

Faraz qiling siz dasturida () belgisini dan keyin tushirib qoldirdingiz, bu runtime erroyemi yoki compile time errormi?

Siz kompyuterdan foydalansangiz "crashed" (spontanlikni to'xtatish) va "hung" (kiruvchingizga samarasiz javoblar) dasturlarida tajribangiz bor. Bu xatti xarakat run time errormi yoki compile time errormi?

Nima uchun kompilyator xatosi bo'lganda dasturingizni run time error ga tekshira olmaysiz?

Amaliyot Bob so'ngida ushbu mashqlarni bajaring:

Agar siz mobodo, so'zni noto'g'ri talaffuz qilsangiz, g'alati holat yuz berishi mumkin, va nima xatolik yuz bergani xaqidagi xabardan voqif bo'lmasligingiz mumkin. Quyida oddiy talaffuz xatolari qanday muammoga olib kelishiga misol berilgan :

```
#include <iostream>

using namespace std;

int main()
{
    cout << "Hello, World!" << endl;

    return 0;
}
```

Bu kod main nomli funksiyani anglatadi. Kompilyator main funksiyasiga aynan o'xshashligini ko'rib chiqmaydi, chunki main katta xarf bilan yoziladi va C++ nozik ish. Katta xarf va kichik xarf tubdan farq qiladi va kompilyatorga main uchun main ni tanlash yaxshiroq *rain* dan ko'ra. Kompilyator main funksiasidan arz qiladi, ammo bog'lovchi amaga oshgan faylni tuzishga tayyor bo'lganda main funksiyasi yo'qligida shikoyat qiladi va bo'g'lovni rad etadi. Albatta yo'qolgan main funksiyasi xatolikni qayerdan topishni ko'rsatadi.

Agar xato xabar kompilyatorni noto'g'ri yo'ldaligini ko'rsatsa, talaffuzdagi xatolikni tekshirish va bosh har bilan yozish kerak. C++ da ko'p nomlar kichik xarf bilan yoziladi. Agar nomlanishda talaffuz xatoligiga yo'l qo'ysangiz, (misol uchun cout o'rniga out), kompilyator "noaniq belgi" ko'rsatadi. Bu xatolik talaffuzdagi xatolik sanaladi.

Nazorat savollari:

1. C++ dasturlash muhiti
2. C++ tilida oddiy dasturlarning tarkibiy qismlari
3. C++ tilida kiritish va chiqorish operatorlari
4. int main () funksiyasi
5. C++ tilida dasturlarning boshlanishi va tugashi
6. Return funksiyasi

Ma'ruza №13

Mavzu: Mantiqiy dasturlash texnologiyasi.

Reja:

1. Identifikator, o'zgaruvchilar toifalari (turlari).
2. C++ algoritmik tilida ma'lumotlarni kiritish va chiqarish, boshqarish qatori, format spetsifikatorlari va modifikatorlari, standart kutubxonasi sarlavha fayllari (.H).
3. C++ da mantiqiy dasturlash texnologiyasi.

Tayanch iboralar: *Ifodalar, operatorlar, arifmetik amallar, qiymat berish operatori, til ko'rsatmasi, inkrement, dekrement, sizeof.*

C++ tilining tayanch turlari.

C++ tilining tayanch turlari, ularning baytlardagi o'lchamlari va qiymatlarining chegaralari 1.2-jadvalda keltirilgan.

1.2-jadval. C++ tilining tayanch turlari

Tur nomi	Baytlardagi o'lchami	Qiymat chegarasi
bool	1	True yoki false
Unsigned short int	2	0..65535
Short int	2	-32768..32767
Unsigned long int	4	0..42949667295

Long int	4	- 2147483648..2147483647
int(16 razryadli)	2	-32768.. 32767
Int (32 razryadli)	4	- 2147483648..2147483647
Unsigned int (16 razryadli)	2	0..65535
Unsigned int (32 razryadli)	4	0..42949667295
Unsigned char	1	0..255
char	1	-128.. 127
float;	4	1.2E-38..3.4E38
double	8	2.2E-308..1.8E308
Long double(32 razryadli)	10	3.4e-4932..-3.4e4932
void	2 ёки 4	-

Butun son turlari. Butun son qiymatlarni qabul qiladigan o'zgaruvchilar int (butun), short: (qisqa) va long (uzun) kalit so'zlar bilan aniqlanadi. O'zgaruvchi qiymatlari ishorali bo'lishi yoki unsigned kalit so'zi bilan ishorasiz son sifatida qaralishi mumkin (1-ilovaga qarang).

Belgi turi Belgi turidagi o'zgaruvchilar char kalit so'zi bilan beriladi va ular o'zida belgining ASCII kodini saqlaydi. Belgi, turidagi qiymatlar nisbatan murakkab bo'lgan tuzilmalar – satrlar, belgilar massivlari va hakoza larni hosil qilishda ishlatiladi (2-ilovaga qarang).

Haqiqiy son turi. Haqiqiy sonlar float kalit so'zi bilan e'lon qilinadi. Bu turdagi o'zgaruvchi uchun xotirada 4 bayt joy ajratiladi va <ishora><tartib><mantissa> qolipida sonni saqlaydi. Agar kasrli son juda katta (kichik) qiymatlarni qabul qiladigan bo'lsa, u xotiradi 8 yoki 10 baytda ikkilangan aniqlik ko'rinishida saqlanadi va mos Double va long double kalit so'zlari bilan e'lon qilinadi. Oxirgi holat 32-razryadli platformalar uchun o'rinli (1-ilovaga qarang).

Mantiqiy tur. Bu turdagi o'zgaruvchi bool kalit so'zi bilan e'lon qilinadi. U turdagi o'zgaruvchi 1 bayt joy egallaydi va 0 (false, yolg'on) qiymatidan farqli qiymat (true, rost) qabul qiladi. Mantikiy tur o'zgaruvchilar qiymatlar o'rtasidagi

munosabatlarni ifodalaydigan mulohazalarni rost (true) yoki yolg'on (false) ekanligini tavsifida qo'llaniladi va ular qabul qiladigan qiymatlar matematik mantiq qonuniyatlariga asoslanadi.

Matematik mantiq - fikrlashning shakli va qonuniyatlari haqidagi fan. Uning asosini mulohazalar hisobi tashkil qiladi. Mulohaza – bu ixtiyoriy jumla bo'lib, unga nisbatan rost yoki yolg'on fikrni bildirish mumkin. Masalan « $3 > 2$ », «5 – juft son», «Moskva-Ukraina poytaxti» va hakoza. Lekin «0.000001 kichik son» jumlasini mulohaza hisoblanmaydi, chunki «kichik son» tushunchasi juda ham nisbiy, ya'ni kichik son deganda qanday sonni tushunish kerakligi aniq emas. Shuning uchun yuqoridagi jumlaning rost yoki yolg'onligi haqida bildirish qiyin.

Mulohazalarning rostligi holatlarga bog'liq ravishda o'zgarishi Masalan «bugun – chorshanba» jumlasini rost yoki yolg'onligi qaralayotgan kunga bog'liq. Xuddi shunday « $x < 0$ » jumlasini x o'zgaruvchisining ayni paytdagi qiymatiga mos ravishda rost yoki yolg'on bo'ladi.

C++ tilida mantiqiy tur nomi angliyalik matematik Jorj Bul sharafiga bool so'zi bilan ifodalangan. Mantiqiy amallar «Bul algebrasi» deyiladi.

Mantiqiy mulohazalar ustida uchta amal aniqlangan:

1) inkor – A mulohazani inkori deganda A rost bo'lganda yolg'on va yolg'on bo'lganda rost qiymat qabul qiluvchi mulohazaga aytiladi. C++ tilida inkor – '!' belgisi bilan beriladi. Masalan, A mulohaza inkori «!A» ko'rinishida yoziladi;

2) konyuksiya- ikkita A va B mulohazalar konyuksiyasi yoki mantiqiy ko'paytmasi «A && B» ko'rinishga ega. Bu mulohaza faqat A va B mulohazalar rost bo'lgandagina rost bo'ladi, aks holda yolg'on bo'ladi (odatda «&&» amali «va» deb o'qiladi). Masalan «bugun oyning 5 kuni va bugun chorshanba» mulohazasi oyning 5 kuni chorshanba bo'lgan kunlar uchun rost bo'ladi;

3) dizyunksiya – ikkita A va B mulohazalar dizyunksiyasi yoki mantiqiy yig'indisi «A || B» ko'rinishida yoziladi. Bu mulohaza rost bo'lishi uchun A yoki B mulohazalardan biri rost bo'lishi yetarli. Odatda «||» amali «yoki» deb o'qiladi.

Yuqorida keltirilgan fikrlar asosida mantiqiy amallar uchun rostlik jadvali aniqlangan (1.3-jadval).

1.3-jadval. Mantiqiy amallar uchun rostlik jadvali

Mulohazalar		Mulohazalar ustida amallar		
A	B	!A	A&&B	A B

false	false	true	false	false
false	true	true	false	true
true	false	false	false	true
true	true	false	true	true

Mantiqiy tur qiymatlari ustida mantiqiy ko‘paytirish, qo‘shish va inkor amallarini qo‘llash orqali murakkab mantiqiy ifodalarni qurish mumkin. Misol uchun, «x –musbat va u qiymati [1..3] sonlar oralig‘iga tegishli emas» mulohazasini mantiqiy ifoda ko‘rinishi quyidagicha bo‘ladi:

$(x > 0) \&\& (y < 1 \parallel y > 3)$.

Void turi. Void turidagi dastur obyekti hech qanday qiymatga ega bo‘lmaydi va bu turdan qurilmaning til sintaksisiga mos kelishini ta‘minlash uchun ishlatiladi. Masalan, C++ tili sintaksisi funksiya qiymat qaytarishini talab qiladi. Agar funksiya qiymat qaytarmaydigan bo‘lsa, u void kalit so‘zi bilan e‘lon qilinadi.

Misollar.

Int a=0 A=1; float abc=17.5;

double Ildiz;

bool ok=true;

char LETTER='z';

Void mening funksiyam ();/*funksiya qaytaradigan qiymat inobatga olinmaydi */

Turlangan o‘zgarmaslar: Turlangan o‘zgarmaslar xuddi o‘zgaruvchilardek ishlatiladi va initsializatsiya qilingandan (boshlang‘ich qiymat berilgandan) keyin ularning qiymatini o‘zgartirib bo‘lmaydi .

Turlangan o‘zgarmaslar const kalit so‘zi bilan e‘lon qilinadi, undan keyin o‘zgarmas turi va albatta initsializatsiya qismi bo‘lishi kerak.

Misol tariqasida turlangan va literal o'zgaraslardan foydalangan holda radius berilganda aylana yuzasini hisoblaydigan dastur ni keltiramiz.

```
#include <iostream.h>

int main ()
{
    const double pi=3.1415;
    const int radius=3;
    double square=0;
    square=pi*radius*radius;
    cout<<square<<"\n";
    return 0;
}
```

Dastur bosh funksiyasining boshlanishida ikkita – pi va radius o'zgaraslari e'lon qilingan. Aylana yuzasini aniqlovchi square o'zgaras deb e'lon qilinmagan, chunki u dastur bajarilishida o'zgaradi. Aylana radiusini dastur ishlashida o'zgartirish mo'ljallanmagan, shu sababli u o'zgaras sifatida e'lon qilingan.

Sanab o'tiluvchi tur: Ko'p miqdordagi, mantiqan bog'langan o'zgaraslardan foydalanganda sanab o'tiluvchi turdan foydalanilgani ma'qul. Sanab o'tiluvchi o'zgaraslar enum kalit so'zi bilan aniqlanadi. Mazmuni bo'yicha bu o'zgaraslar oddiy butun sonlardir. Sanab o'tiluvchi o'zgaraslar C++ standarti bo'yicha butun turdagi o'zgaraslar hisoblanadi. Har bir o'zgarasga (songa) mazmunli nom beriladi va bu identifikatorni dastur ning boshqa joylarida nomlash uchun ishlatilishi mumkin emas. Sanab o'tiluvchi tur qo'yidagi ko'rinishga ega:

```
enum <Sanab o'tiladigan tur nomi> { <nom1> =<qiymat1>, <nom2> qiymat2>, ... <nomn> =<qiymatn> } ;
```

bu yerda, enum – kalit so'z (inglizcha enumerate – sanamoq); <Sanab O'tiladigan tur nomi>- o'zgaraslar ro'yxatining nomi; <nom> - butun qiymatli konstantalarning nomlari; <qiymat_i> - shart bo'lmagan initsializatsiya qiymati (ifoda).

Misol uchun hafta kunlari bilan bog'liq masala yechishda hafta kunlari dush (dushanba), sesh (seshanba), shor (chorshanba), paysh (payshanba), juma (juma), shanba (shanba), yaksh (yakshanba) o'zgaraslarini ishlatish mumkin va ular sanab o'tiluvchi tur yordamida bitta satrda yoziladi:

```
enum Hafta {dush, sesh chor, paysh, juma, shanba,yaksh} ;
```

Sanab o'tiluvchi o'zgarmlar quyidagi xossaga ega: agar o'zgarmlar qiymati ko'rsatilmagan bo'lsa, u oldingi o'zgarmlar qiymatidan bittaga ortiq bo'ladi. Kelishuv bo'yicha birinchi o'zgarmlar qiymati 0 bo'ladi.

Initsializatsiya yordamida o'zgarmlar qiymatini o'zgartirish mumkin:

```
enum Hafta {dush=8 ,sesh,chor=12 ,paysh=13 , juma=16, shanba, yaksh=20} ;
```

Bu e'londa sesh qiymati 9, shanba esa 17 ga teng bo'ladi.

Sanab o'tiluvchi o'zgarmlarning nomlari har xil bo'lishi kerak, lekin ularning qiymatlari bir xil bo'lishi mumkin:

```
enum{no1=0,toza=0,bir,ikki,juft=2 ,uch} ;
```

O'zgarmlarning qiymati ifoda ko'rinishda berilishi mumkin, faqat ifodadagi nomlarning qiymatlari shu qadamdagacha aniqlangan bo'lishi kerak:

```
enum { ikki=2 ,turt=ikki*2 } ;
```

O'zgarmlarni qiymatlari manfiy son bo'lishi ham mumkin:

```
enum {minus2=-2 minus1,nol,bir} ;
```

Turni boshqa turga keltirish:

C++ tilida bir turni boshqa turga keltirishning oshkor va oshkormas yo'llari mavjud.

Umuman olganda, turni boshqa turga oshkormas keltirish ifodada har xil turdagi o'zgaruvchilar qatnashgan hollarda amal qiladi (aralash turlar arifmetikasi). Ayrim hollarda, xususan tayanch turlar bilan bog'liq turga keltirish amallarida xatoliklar yuzaga kelishi mumkin. Masalan, hisoblash natijasining xotiradan vaqtincha egallagan joyi uzunligi, uni o'zlashtiradigan o'zgaruvchi uchun ajratilgan joy uzunligidan katta bo'lsa, qiymatga ega razryadlarni yo'qotish holati yuz beradi.

Oshkor ravishda turga keltirishda, o'zgaruvchi oldiga qavs ichida boshqa tur nomi yoziladi:

```
#include <iostream.h>
```

```
int main()
```

```
{
```

```
int integer_1=54;
```

```
int integer_2;
```

```
float floating=15.854;
```

```

integer_1=(int) floating; // oshkor keltirish;
integer_2=(int) floating // oshkormas keltirish;
cout<<"yangi integer (oshkor): "<<integer_1<<"\n";
cout<<"yangi integer (oshkormas): "<<integer_2<<"\n";
return 0;
}

```

Dastur natijasi quyidagi ko'rinishida bo'ladi:

Yangi integer (oshkor):15

Yangi integer (oshkormas):15

Masala. Berilgan belgining ASCII kodi chop etilsin. Masala belgi turidagi qiymatni oshkor ravishda butun son turiga keltirib chop qilish orqali yechiladi.

Dastur matni:

```

#include <iostream.h>
int main()
{
Unsigned char A;
Cout<<"belgini kiriting:";
Cin>>A;
Cout<<" "<<A<<"-belgi ASCII kodi="(int)A<<"\n";
Return 0;
}

```

Dastur ning

Belgini kiriting:

So'roviga

A <enter>

amali bajarilsa, ekranga

'A'-belgi ASCII kodi=65

satri chop etiladi.

Arifmetik amallar. Qiymat berish operatori.

Berilganlarni qayta ishlash uchun C++ tilida amallarning juda keng majmuasi aniqlangan. Amal - bu qandaydir harakat bo'lib, u bitta (unar) yoki ikkita (binar) operandlar ustida bajariladi, hisob natijasi uning qaytaruvchi qiymati hisoblanadi.

Tayanch arifmetik amallarga qo'shish (+), ayirish (-), ko'paytirish (*), bo'lish (/) va bo'lish qoldig'ini olish (%) amallarini keltirish mumkin.

Amallar qaytaradigan qiymatlarni o'zlashtirish uchun qiymat berish amali (=) va uning turli modifikatsiyalari ishlatiladi: qo'shish, qiymat berish bilan (+=); ayirish, qiymat berish bilan (-=); ko'paytirish, qiymat berish bilan (*=); bo'lish, qiymat berish bilan (/=); bo'lish qoldig'ini olish, qiymat berish bilan (%=) va boshqalar. Bu holatlarning umumiy ko'rinishi:

<o'zgaruvchi><amal>=<ifoda>;

Quyidagi dastur matnida ayrim amallarga misollar keltirilgan.

```
#include <iostream.h>

int main()
{
    int a=0 , b=4, c=90;char z='\t';

    a=b; cout<<a<<z;           //a=4
    a=b+c+c+b; cout<<a<<z;      //a=4+90+90+4=188
    a=b-2; cout<<a<<z;          //a=2
    a=b*3 cout<<a<<z;           //a=4*3=12
    a=c/(b+6); cout<<a<<z;      //a=90/(4+6)=9
    cout<<a%2<<z;              //9%2=1
    a+=b; cout<<a<<z;           //a=a+b=9+4=13
    a*=c-50; cout<<a<<z;        //a=a*(c-50)=13*(90-50)=520
    a-=38; cout<<a<<z;          //a=a-38=520-38=482
    a%=8; cout<<a<<z;           //a=a%8=482%8=2

    return 0;
}
```

Dastur bajarilishi natijasida ekranga quyidagi sonlar satri paydo bo'ladi:

4 188 2 12 9 1 482 2

Ifoda tushunchasi:

C++ tilida ifoda - amallar, operandlar va punktatsiya belgilarining ketma-ketligi bo'lib, kompilyator tomonidan berilganlar

ustida ma'lum bir amallarni bajarishga ko'rsatma hisoblanadi. Har qanday ';' belgi bilan tugaydigan ifodaga til ko'rsatmasi deyiladi.

C++ tilidagi ifodaga (til ko'rsatmasiga) misol:

```
x=3*(y-2.45);
```

```
u=summa(a, 9,c) ;
```

Inkrement va dekrement amallari:

C++ tilida operand qiymatini birga oshirish va kamaytirishning samarali vositalari mavjud. Bular inkrement (++) va dekrement) unar amallardir.

Operandga nisbatan bu amallarning prefiks va postfiks ko'rinishlari bo'ladi. Prefiks ko'rinishda amal til ko'rsatmasi bo'yicha ish bajarilishidan oldin operandga qo'llaniladi. Postfiks holatda esa amal til ko'rsatmasi bo'yicha ish bajarilgandan keyin operandga qo'llaniladi.

Prefiks yoki postfiks amal tushunchasi faqat qiymat berish bilan bog'liq ifodalarda o'rinli:

```
x=y++; // postfiks
```

```
index =--i; // prefiks
```

```
count:++; // unar amal, "++count; " bilan ekvivalent
```

```
abc-- ; // unar amal, "--abc; " bilan ekvivalent
```

Bu yerda u o'zgaruvchining qiymatini x o'zgaruvchisiga o'zlashtiriladi va keyin bittaga oshiriladi, i o'zgaruvchining qiymati bittaga kamaytirib, index o'zgaruvchisiga o'zlashtiriladi.

sizeof amali.

Har xil turdagi o'zgaruvchilar kompyuter xotirasida turli sondagi baytlarni egallaydi. Bunda, hattoki bir turdagi o'zgaruvchilar ham kaysi kompyuterda yoki qaysi operatsion sistemada amal qilinishiga qarab turli o'lchamdagi xotirani band qilishi mumkin.

C++ tilida ixtiyoriy (tayanch va hosilaviy) turdagi o'zgaruvchilarning o'lchamini sizeof amali yordamida aniqlanadi. Bu amalni o'zgarmasga, turga va o'zgaruvchiga qo'llanishi mumkin.

Quyida keltirilgan dastur da kompyuterning platformasiga mos ravishda tayanch turlarining o'lchamlari chop qilinadi.

```

Int main()
{cout<<"int turining o'lchami: "<<sizeof (int)<<"\n"
{cout<<"float turining o'lchami: "<<sizeof (float)<<"\n";
{cout<<"double turining o'lchami: "<<sizeof (double)<<"\n";
{cout<<"char turining o'lchami: "<<sizeof (char)<<"\n";
return 0;
}

```

Razryadli mantiqiy amallar.

Dastur tuzish tajribasi shuni ko'rsatadiki, odatda qo'yilgan masalani yechishda biror holat ro'y bergan yoki yo'qligini ifodalash uchun 0 va 1 qiymat qabul qiluvchi bayroqlardan foydalaniladi. Bu maqsadda bir yoki undan ortiq baytli o'zgaruvchilardan foydalanish mumkin. Masalan, bool turidagi o'zgaruvchini shu maqsadda ishlatib bo'ladi. Boshqa tomondan, bayroq sifatida baytning razryadlaridan foydalanish ham mumkin. Chunki razryadlar faqat ikkita qiymatni – 0 va 1 sonlarini qabul qiladi. Bir baytda 8 razryad bo'lgani uchun unda 8 ta bayroqni kodlash imkoniyati mavjud.

Faraz qilaylik, qo'riqlash tizimiga 5 ta xona ulangan va tizim taxtasida 5 ta chiroqcha (indikator) xonalar holatini bildiradi: xona qo'riqlash tizimi nazoratida ekanligini mos indikatorning yonib turishi (razryadning 1 qiymati) va xonani tizimga ulanmaganligini indikator o'chganligi (razryadning 0 qiymati) bildiradi. Tizim holatini ifodalash uchun bir bayt yetarli bo'ladi va uning kichik razryadidan boshlab beshtasini shu maqsadda ishlatish mumkin:

7 6 5 4 3 2 1 0

			ind5	ind4	ind3	ind2	ind1

Masalan, baytning quyidagi holati 1, 4 va 5 xonalar qo'riqlash tizimiga ulanganligini bildiradi:

7 6 5 4 3 2 1 0

x	x	x	1	1	0	0	1
---	---	---	---	---	---	---	---

Quyidagi jadvalda C++ tilida bayt razryadlari ustida mantiqiy amallar majmuasi keltirilgan.

3.1 –jadval. Bayt razryaddari ustida mantiqiy amallar

Amallar	Mazmuni
&	Mantiqiy VA (ko'paytirish)
	Mantiqiy yoki (qo'shish))
$\wedge$	Istisno qiluvchi YOKI
$\sim$	Mantiqiy INKOR (inversiya)

Razryadli mantiqiy amallarning bajarish natijalarini jadval ko'rinishida ko'rsatish mumkin.

3.2-jadval. Razryadli mantiqiy amallarning bajarish natijalari

A	B	$C=A \& B$	$C=A B$	$C=A \wedge B$	$C=\sim A$
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

Yuqoridagi keltirilgan misol uchun qo'riqlash tizimini ifodalovchi bir baytli char turidagi o'zgaruvchini e'lon qilish mumkin:

char q_taxtasi=0;

Bu yerda q_taxtasi o'zgaruvchisiga 0 qiymat berish orqali barcha xonalar qo'riqlash tizimiga ulanmaganligi ifodalanadi:

7 6 5 4 3 2 1 0

0	0		0	0	0	0	0	0

Agar 3-xonani tizimga ulash zarur bo'lsa

$q_taxtasi = q_taxtasi \text{ } 0x04_{16};$

amalini bajarish kerak, chunki $0x04_{16} = 00000100_2$ va mantiqiy YOKI amali natijasida $q_taxtasi$ o'zgaruvchisi bayti quyidagi ko'rinishda bo'ladi:

7 6 5 4 3 2 1 0

0	0	0	0	0	1	0	0

Xuddi shunday yo'l bilan boshqa xonalarni tizimga ulash mumkin, zarur bo'lsa birdaniga ikkitasini (zarur bo'lsa barchasini):

$q_taxtasi = q_taxtasi | 0x1F_{16};$

Mantiqiy ko'paytirish orqali xonalarni qo'riqlash tizimidan chiqarish mumkin:

$q_taxtasi = q_taxtasi \& 0xFD; // 0xFD_{16} = 11111101_2$

Xuddi shu natijani '~' amalidan foydalangan holda ham olish mumkin. Ikkinchi xona tizimga ulanganligi bildiruvchi bayt qiymati 00000010_2 , demak shu holatni inkor qilgan holda mantiqiy ko'paytirishni bajarish kerak.

$q_taxtasi = q_taxtasi \& (\sim 0x02);$

Va nihoyat, agar 3-xona indikatorini, uni qanday qiymatda bo'lishidan qat'iy nazar qarama-qarshi holatga o'tkazishni «inkor qiluvchi YOKI» amali yordamida bajarish mumkin:

$q_taxtasi = q_taxtasi \wedge 0x04; // 0x04_{16} = 00000100_2$

Razryadli mantiqiy amallarni qiymat berish operatori birgalikda bajarilishining quyidagi ko'rinishlari mavjud:

$\&=$ - razryadli VA qiymat berish bilan;

$|=$ - razryadli YOKI qiymat berish bilan;

$\wedge=$ - razryadli istisno qiluvchi YOKI qiymat berish bilan.

Chapga va o'ngga surish amallari.

Baytdagi bitlar qiymatini chapga yoki o'ngga surish uchun, mos ravishda “<” va “>” amallari qo'llaniladi. Amaldan keyingi son bitlar nechta o'rin chapga yoki o'ngga surish kerakligini bildiradi.

Masalan:

unsigned char A=12; //A=00001100₂=0x0C₁₆

A=A<<2; // A=00110000₂=0x30₁₆=48₁₀

A=A>>3; //A=00000110₂=0x06₁₆=6₁₀

Razryadlarni n ta chapga (o'ngga) surish sonni 2^n soniga ko'paytirish (bo'lish) amali bilan ekvivalent bo'lib va nisbatan tez bajariladi. Shuni e'tiborga olish kerakki, operand ishorali son bo'lsa, u holda chapga surishda eng chapdagi ishora razryadi takrorlanadi (ishora saqlanib qoladi) va manfiy sonlar ustida bu amal bajarilganda matematika nuqtai-nazardan xato natijalar yuzaga keladi:

unsigned char B=-120; // B=10001000₂=0x88₁₆

B=B<<2; // B=00100000₂=0x20₁₆=32₁₀

B=-120; // B=10001000₂=0x88₁₆

B=B>>3; // B=11110001₂=0xF1₁₆=-15₁₀

Shu sababli, bu razryadli surish amallari ishorasiz (unsigned) turdagi qiymatlar ustida bajarilgani ma'qul.

Taqqoslash amallari.

C++ tilida qiymatlarni solishtirish uchun taqqoslash amallari aniqlangan (3.3-jadval). Taqqoslash amali binary amal bo'lib, quyidagi ko'rinishga ega:

<operand1> <taqqoslash amali> <operand2>

Taqqoslash amallarining natijasi – taqqoslash o'rinli bo'lsa, true (rost), aks holda false (yolg'on) qiymat bo'ladi. Agar taqqoslashda arifmetik ifoda qatnashsa, uning qiymati 0 qiymatidan farqli holatlar uchun 1 deb hisoblanadi.

13.3-jadval. Taqqoslash amallari va ularning qo'llanishi

Amallar	Qo'llanishi	Mazmuni (o'qilishi)
<	a<b	“a kichik b”

$\leq$	$a \leq b$	“a kichik yoki teng b”
$>$	$a > b$	“a katta b”
$\geq$	$a \geq b$	“a katta yoki teng ”
$==$	$a == b$	“a teng b”
$!=$	$a != b$	“a teng emas b”

«Vergul» amali.

Til qurilmalaridagi bir nechta ifodalarni kompilyator tomonidan yaxlit bir ifoda deb qabul qilishi uchun «vergul» amali qo'llaniladi. Bu amalni qo'llash orqali dastur yozishda ma'lum bir samaradorlikka erishish mumkin. Odatda «vergul» amali if va for operatorlarida keng qo'llaniladi. Masalan, if operatori qo'yidagi ko'rinishda bo'lishi mumkin:

`if(i=Ca11func(),i<7) . . .`

Bu yerda, oldin `Ca11func()` funksiyasi chaqiriladi va uning natijasi `i` o'zgaruvchisiga o'zlashtiriladi, keyin `i` qiymati 7 bilan solishtiriladi.

Amallarning ustunliklari va bajarilish yo'nalishlari.

An'anaviy arifmetikadagidek C++tilida ham amallar ma'lum bir tartib va yo'nalishda bajariladi. Ma'lumki, matematik ifodalarda bir xil ustunlikdagi (prioritetdagi) amallar uchrasa (masalan, qo'shish va ayirish), ular chapdan o'ngga bajariladi. Bu tartib C++ tilidagi ham o'rinli, biroq ayrim hollarda amal o'ngdan chapga bajarilishi mumkin (qiymat berish amalida).

Ifodalar qiymatini hisoblashda amallar ustunligi hisobga olinadi, Birinchi navbatda eng yuqori ustunlikka ega bo'lgan amal bajariladi.

Operator	Tavsifi	Ustunligi	Yo'nalishi
::	Ko'rinish sohasiga ruxsat berish	16	=>
[]	Massiv indeksi	16	=>
()	Funksiyani chaqirish	16	=>

.	Tuzilma yoki sinf elementini tanlash	16	=>
++	Postfiks inkrement	15	<=
--	Postfiks dekrement	15	<=
++	Prefiks inkrement	14	<=
--	Prefiks dekrement	14	<=
sizeof	O'lchamni olish	14	<=
(<typ>)	Turga akslantirish	14	
~	Bitli mantiqiy INKOR	14	<=
!	Mantiqiy inkor	14	<=
-	Unar minus	14	<=
+	Unar plyus	14	<=
&	Adresni olish	14	<=
*	Vositali murojaat	14	<=
new	Dinamik obyekttni yaratish	14	<=
delete	Dinamik obyekttni yo'q qilish	14	<=
casting	Turga keltirish	14	
*	Ko'paytirish	13	=>
/	Bo'lish	13	=>
%	Bo'lish qoldig'i	13	=>
+	Qo'shish	12	=>

-	Ayirish	12	=>
»	Razryad bo'yicha o'ngga surish	11	=>
«	Razryad bo'yicha chapga surish	11	=>
<	Kichik	10	=>
<=	Kichik yoki teng	10	=>
>	Katta	10	=>
>=	Katta yoki teng	10	=>
==	Teng	9	=>
!=	Teng emas	9	=>
&	Razryadli VA	8	=>
^	Razryadli istisno qiluvchi YOKI	7	=>
	Razryadli YOKI	6	=>
&&	Mantiqiy VA	5	=>
	Mantiqiy yoki	4	=>
?:	Shart amali	3	<=
=	Qiymat berish	2	<=
*=	Ko'paytirish qiymat berish amali bilan	2	<=

/=	Bo'lish qiymat berish amali bilan	2	<=
%=	Modulli bo'lish qiymat berish amali bilan	2	<=
+=	Qo'shish qiymat berish amali bilan	2	<=
-=	Ayirish qiymat berish amali bilan	2	<=
<<=	Chapga surish qiymat berish amali bilan	2	<=
>>=	o'ngga surish qiymat berish amali bilan	2	<=
&=	Razryadli va qiymat berish bilan	2	<=
^=	Razryadli istisno qiluvchi yoki qiymat berish bilan	2	<=
=	Razryadli yoki qiymat berish bilan	2	<=
Throw	Istisno holatni yuzaga keltirish	2	<=
,	vergul	1	<=

C++ tili dastur tuzuvchisiga amallarning bajarilish tartibini o'zgartirish imkoniyatini beradi. Xuddi matematikadagidek, amallarni qavslar yordamida guruhlarga jamlash mumkin. Qavs ishlatishga cheklov yo'q.

Quyidagi dastur da qavs yordamida amallarni bajarish tartibini o'zgartirish ko'rsatilgan.

```
#include <iostream.h>

int main ()
{
    int x=0,y=0;
    int a=3, b=34, c=82;
    x=a*b+c;
    y=(a*(b+c));
    cout<<"x="<<x<<"n\''
    cout<<"y="<<y<<"n\''
```

}

Dastur da amallar ustunligiga ko'ra x qiymatini hisoblashda oldin a o'zgaruvchi b o'zgaruvchiga ko'paytiriladiva unga c o'zgaruvchi qiymatiga qo'shiladi .

Navbatdagi ko'rsatmani bajarishda esa birinchi navbatda ichki qavs ichidagi ifoda – (b+c) qiymati hisoblanadi, keyin bu qiymat a ko'paytirilib, u o'zgaruvchisiga o'zlashtiriladi.

Dastur bajarilishi natijasida ekranga

x=184

y=348

satlari chop etiladi.

Operator tushunchasi.

Dasturlash tili operatorlari yechilayotgan masala algoritmini amalga oshirish uchun ishlatiladi. Operatorlar chiziqli va boshqaruv operatorlariga bo'linadi. Aksariyat holatlarda operatorlar nuqtali vergul (;) belgisi bilan tugallanadi va u kompilyator tomonidan alohida operator deb qabul qilinadi (for operatorining qavs ichida turgan ifodalari bundan mustasno). Bunday operator ifoda operatori deyiladi. Qiymat berish amallari guruhi, xususan, qiymat berish operatorlari ifoda operatorlari hisoblanadi:

i++; --j; k+=I;

Dastur tuzish amaliyotida bo'sh operator – ';' ishlatiladi. Garchi bu operator hech nima bajarmasa ham, hisoblash ifodalarini til qurilmalariga mos kelishini ta'minlaydi. Ayrim hollarda yuzaga kelgan «boshi berk» holatlardan chiqib ketish imkonini beradi.

O'zgaruvchilarni e'lon qilish ham operator hisoblanadi va ularga e'lon operatori deyiladi.

Nazorat savollari.

1. C++ tilida ifodalar.
2. Inkrement va dekrement amallari.
3. Tayanch arifmetik amallar.
4. Taqqoslash amallari.
5. C++ tilida tuzilgan arifmetik ifodalarga misollar keltiring.
6. Inkrement amalining qo'llanilishiga doir misol keltirig.
7. Dekrement amalining qo'llanilishiga doir misol keltirig.
8. Sizeof amalining qo'llanilishi.
9. C++ tilida bayt razryadlari ustida bajariladigan amallar.

10. Razryadli mantiqiy amallarni qiymat berish operatori bilan birgalikda bajarilishi.
11. Hisoblashda amallarning ustunligi.
12. Amallarning bajarilish yo'nalishlari.

Ma'ruza № 14

Mavzu: Dasturlashda funksiya va modullarni qo'llash.

Reja:

1. Borland C++ Builder 6 ning strukturalashgan toifalari. Muntazam toifa. Funksiya va modullar.
2. Qatorlar, literal (belgi)lar, funksiyalar, ko'rsatkichlar.
3. Parametrlarni funksiyalarga uzatish.
4. Lokal, static, dinamik o'zgaruvchilar.

Tayanch iboralar: *Break operatori, continue operatori, goto operatori, nishonlar. Funksiya parametrlari va argumentlari, kelishuv bo'yicha argumentlar, ko'rinish sohasi. lokal va global o'zgaruvchilar.*

Break operatori.

Takrorlash operatorlarining bajarilishida shunday holatlar yuzaga kelishi mumkinki, unda qaysidir qadamda, takrorlashni yakuniga yetkazmasdan takrorlashdan chiqish zarurati bo'lishi mumkin. Boshqacha aytganda takrorlashni «uzish» kerak bo'lishi mumkin. Bunda break operatoridan foydalaniladi. Break operatorini takrorlash operatori tanasining ixtiyoriy (zarur) joylariga qo'yish orqali shu joylardan takrorlashdan chiqishni amalga oshirish mumkin. E'tibor beradigan bo'lsak switch-case operatorining tub mohiyatiga ham break operatorini qo'llash orqali erishilgan.

Ichma – ich joylashgan takrorlash va switch operatorlarida break operatori faqat o'zi joylashgan blokdan chiqish imkoniyatini beradi.

Quyidagi dasturda ikkita ichma-ich joylashgan takrorlash operatoridan foydalangan holda foydalanuvchi tomonidan kiritilgan qandaydir sonni 3 va 7 sonlariga nisbatan qanday oraliqqa tushishi aniqlanadi. Tashqi takrorlashda 'son kiriting (0-to'xtash):_' so'rovi beriladi va javob javob_son o'zgaruvchisiga o'qiladi. Agar son noldan farqli bo'lsa, ichki takrorlash operatorida bu sonning qandaydir tushishi aniqlanib, shu haqida xabar beriladi va ichki operatoridan chiqiladi. Tashqi takrorlashdagi so'rovga javob tariqasida 0 kiritilsa, dastur o'z ishini tugatadi.

```

#include <iostream.h>

int main()
{
    int javob_son=0;
    do
    {
        while (javob_son)
        {
            if (javob_son<3)
            {cout<<"3 kichik !"; break;}
            if(3<=javob_son&& javob_son<=7)
            {cout<<"3 va 7 oralig'da !"; break;}
            if (javob_son>7)
            {cout<<"7 dan katta !"; break;}
        }
        cout<<"\nSon kiriting (0-to'xtash): _";
        cin>>javob_son;
    }
    while(javob_son !=0)
    return 0
}

```

Amaliyotda break operatoridan cheksiz takrorlashdan chiqishda foydalaniladi.

```

For (;;)
{
    // 1-shart
    if (...)
    {
        ...
    }
}

```

```

break ;
}
// 2- shart
if (...)
{
...
break;
}
...
}

```

Bu misolda cheksiz for takrorlashidan 1 yoki 2- shart bajarilganda chiqiladi,

Masala. Ishorasiz butun sonlar ketma-ketligi 0 qiymati bilan tugaydi. Bu yerda 0 ketma-ketlik hadi hisoblanmaydi. Ketma-ketlikni kamaymaydigan holda tartiblangan yoki yo‘qdigi aniqlansin.

```

#include <iostream.h>

int main()
{
unsigned int Ai_1=0,Ai;
cout<<" sonlar ketma-ketligini kiriting"
cout<<(0-tugash alomati):\n";
cin>>Ai; // ketma-ketlikning birinchi hadi
while(Ai)
{Ai_1>Ai;
cin>>Ai; // navbatdagi had
if (Ai_1>Ai) break;
}
if (Ai_1)
{
cout<<"ketma-ketlik kamaymaydigan holda tartiblangan";

```

```

if(!Ai)cout<<"emas!";
else cout<<"!";
}
}
else cout<<"ketma-ketlik bo'sh!";
return 0;
}

```

Dastur ishga tushganda, boshida ketma-ketlikning birinchi hadi alohida o'qib olinadi (Ai o'zgaruvchisiga). Keyin Ai qiymati nolga teng bo'lmaguncha takrorlash operatori amal qiladi. Takrorlash tanasida Ai qiymati oldingi qiymat sifatida Ai_1 o'zgaruvchisida eslab qolinadi va navbatdagi had Ai o'zgaruvchisiga o'qiladi. Agar oldingi had navbatdagi haddan katta bo'lsa, break operatori yordamida takrorlash jarayoni uziladi va boshqaruv takrorlashdan keyingi shart operatoriga o'tadi. Bu yerdagi shart operatorlari mazmuni quyidagicha agar Ai_1 noldan farqli bo'lsa, ketma-ketlikning kamida bitta hadi kiritilgan bo'ladi (ketma-ketlik mavjud) va oxirgi kiritilgan had tekshiriladi. O'z navbatida agar Ai noldan farqli bo'lsa, bu holat hadlar o'rtasida kamaymaslik sharti bajarilmaganligi sababli hadlarni

kiritish jarayoni uzilganligini bildiradi va bu haqda xabar chop etiladi. Aks holda ketma-ketlikni kamaymaydigan holda tartiblangan bo'ladi.

Continue operatori.

Continue operatori xuddi break operatoridek takrorlash operatori tanasini bajarishni to'xtatadi, lekin takrorlashdan chiqib ketmasdan keyingi qadamiga «sakrab» o'tishini tayinlaydi.

Continue operatorini qo'llanishiga misol tariqasida 2 va 50 sonlar oralig'idagi tub sonlarni topadigan dastur matnini keltiramiz.

```

#include <iostream.h>
int main()
{
    bool bulinadi=false;
    for (int i=2; i<50; i++)
    {
        for (int j=2; j<i/2; j++)

```

```

    {if (i%j) continue;
    else {bulinadi=true; break;}
    }
    // break bajarilganda boshqaruv o'tadigan joy
    if (!bulinadi ) cout <<i<<" ";
    bulinadi=false;
    }
    return 0;
    }

```

Keltirilgan dasturda qo'yilgan masala ichma-ich joylashgan ikkita takrorlash operatorlari yordamida yechilgan. Birinchi takrorlash operatori 2 dan 50 gacha sonlarni hosil qilishga xizmat qiladi. Ichki takrorlash esa har bir hosil qilinayotgan sonni 2 sonidan toki shu sonning yarmigacha bo'lgan sonlarga bo'lib, qoldig'ini tekshiradi, agar qoldiq 0 sonidan farkli bo'lsa, navbatdagi songa bo'lish davom etadi, aks holda bulinadi o'zgaruvchisiga true qiymat berib, ichki takrorlash uziladi (son o'zining yarmigacha bo'lgan qandaydir songa bo'linar ekan, demak u tub emas va keyingi sonlarga bo'lib tekshirishga hojat yo'q). Ichki takrorlashdan chiqqandan keyin bulinadi qiymati false bo'lsa (!bulinadi), son tub bo'ladi va u chop qilinadi.

Goto operatori va nishonlar.

Nishon – bu davomida ikkita nuqta (':') qo'yilgan identifikator. Nishon bilan qandaydir operator belgilanadi va keyinchalik, dasturning boshqa bir qismidan unga shartsiz o'tish amalga oshiriladi. Nishon bilan har qanday operator belgilanishi mumkin, shu jumladan e'lon operatori va bo'sh operatori ham. Nishon faqat funksiyalar ichida amal qiladi.

Nishonga shartsiz o'tish goto operatori yordamida bajariladi. Goto operatori orqali faqat uning o'zi joylashgan funksiya ichidagi operatorlarga o'tish mumkin. Goto operatorining sintaksisi quyidagicha:

```
goto <nishon>;
```

Ayrim hollarda, goto operatorining «sakrab o'tishi» hisobiga xatoliklar yuzaga kelishi mumkin. Masalan,

```

int i=0;
i++; if(i) goto m;
int j;

```

m: j+=I;

operatorlarining bajarilishi xatolikka olib keladi, chunki e'lon qilinishi qoladi.

Shartsiz o'tish operatori dasturni tuzishdagi kuchli va shu

Bilan birga xavfli vositalardan biri hisoblanadi. Kuchliligi shundaki, uning yordamida algoritmnining «boshi berk» joylaridan chiqib ketish mumkin. Ikkinchi tomondan, bloklarning ichiga o'tish, masalan takrorlash operatorlarini ichiga «sakrab» kirish kutilmagan holatlarni yuzaga keltirishi mumkin. Shu sababli, imkon qadar goto operatoridan foydalanmaslik kerak, ishlatilgan taqdirda ham qo'yidagi qoidaga amal qilish zarur – blok ichiga, if...else va switch operatorlari ichiga, hamda takrorlash operatorlari tanasiga tashqaridan kirish mumkin emas.

Garchi, nishon yordamida dasturning ixtiyoriy joyiga o'tish mumkin bo'lsa ham, boshlang'ich qiymat berish e'lonlaridan sakrab o'tish man etiladi, lekin bloklardan sakrab o'tish mumkin.

Masalan:

...

goto B; [\xato](#)

float x=0.0;

goto B; \ to'g'ri

{int n=10;x=n*x+x;}

B: cout<< "x="<<x;

...

Xususan, nishon yordamida ichki blokdan tashqi blokka va tashqi blokdan ichki blokka o'tishga C++ tili ruxsat beradi:

{...

goto ABC:

...

{int i=15;

...

ABC:

...

goto XYZ;

```

int y=10;

...

goto KLM;

...}

...

int k=0;

...

KLM:

...}

```

Lekin, yuqorida keltirilgan misoldagi barcha o'tishlar mazmunan xato hisoblanadi.

Quyidagi dasturda ikkita natural sonlar EKUBini topish masalasidagi takrorlash jarayonini nishon va goto operatori vositasida amalga oshirish ko'rsatilgan:

```

int main()
{
int a,b;
cout<<"A va B natural sonlar EKUBini topish.\n";
cout<<"A va B natural sonlarni kiriting: "
cin>>a>>b;
nishon:
if (a==b)
{
cout<<"Bu sonlar EKUBi="<<a;
return 0;
}
a>b?a-=b:b-=a;
goto nishon;
}

```

Dasturdagi nishon bilan belgilangan operatorlarda a va b sonlarni tengligi tekshiriladi. Agar ular teng bo'lsa, ixtiyoriy bittasi, masalan a soni EKUB bo'ladi va funksiyadan chiqiladi. Aks holda, bu sonlarning kattasidan kichigi ayriladi va goto orqali ularning tengligi tekshiriladi. Takrorlash jarayoni a va b sonlar o'zaro teng bo'lguncha davom etadi.

Shuni qayd etish kerakki, bu masalani takrorlash operatorlari yordamida bajarish ancha samarali hisoblanadi.

Dastur ta'minotini yaratish amalda murakkab jarayon hisoblanadi. Dastur tuzuvchi dastur kompleksini bir butunlikdagi va uning har bir bo'lagining ichki mazmunini va ularning sezilmas farqlarini hisobga olishi kerak bo'ladi.

Dasturlashga tizimli yondoshuv shundan iboratki, dastur tuzuvchi oldiga qo'yilgan masala oldindan ikkita, uchta va undan ortiq nisbatan kichik masala ostilarga bo'linadi. O'z navbatida bu masala ostilari ham yana kichik masala ostilariga bo'linishi mumkin.

Bu jarayon toki mayda masalalarni oddiy standart amallar yordamida yechish mumkin bo'lguncha davom etadi. Shu yo'l bilan masalani dekompozitsiyalash amalga oshiriladi.

Ikkinchi tomondan, dasturlashda shunday holatlar kuzatiladiki, unda dasturning turli joylarida mazmunan bir xil algoritmlarni bajarishga to'g'ri keladi. Algoritmning bu bo'laklari asosiy yechilayotgan masaladan ajratib olingan qandaydir masala ostini yechishga mo'ljallangan bo'lib, yetarlicha mustaqil qiymatga (natijaga) egadir. Misol uchun quyidagi masalani ko'raylik:

Berilgan $a_0, a_1, \dots, a_{30}, b_0, b_1, \dots, b_{30}, c_0, c_1, \dots, c_{30}$ va x, y, z haqiqiy sonlar yechish

$$\frac{(a_0 x^{30} + a_1 x^{29} + \dots + a_{30})^2 - (b_0 y^{30} + b_1 y^{29} + \dots + b_{30})}{c_0 (x+z)^{30} + c_1 (x+z)^{29} + \dots + c_{30}}$$

ifodaning qiymati hisoblansin.

Bu misolni yechishda kasrning surat va maxrajidagi ifodalar bir xil algoritm bilan hisoblanadi va dasturda har bir ifodani (masala osti) hisoblash uchun bu algoritmni 3 marta yozishga to'g'ri keladi. Masaladagi 30-darajali ko'phadni hisoblash algoritmini, masalan, Gornier algoritmini alohida, bitta nusxada yozib, unga turli parametrlar- bir safar a vektor va x qiymatini, ikkinchi safar b vektor va y qiymatini hamda c vektor va $(x+z)$ qiymatlari bilan murojaat qilish asosiy masalani yechish mumkin bo'ladi. Funksiyalar qo'llanishining yana bir sababini quyidagi masalada ko'rishimiz mumkin – berilgan chiziqli tenglamalar sistemasini Gauss,

Kramer, Zeydel usullarining birortasi bilan yechish talab qilinsin. U holda asosiy dasturni quyidagi bo‘laklarga bo‘lish maqsadga muvofiq bo‘lar edi: tenglama ko‘effitsentlarini kiritish bo‘lagi, yechish usulini tanlash bo‘lagi, Gauss, Kramer, Zeydel usullarini amalga oshirish uchun alohida bo‘laklar, natijani chop qilish bo‘lagi. Har bir bo‘lak uchun o‘z funksiyadir majmuasi yaratib, zarur bo‘lganda ularga bosh funksiya tanasidan murojaatni amalga oshirish orqali bosh masala yechish samarali hisoblanadi.

Bunday hollarda dasturni ixcham va samarali qilish uchun C++ tilida dastur bo‘lagini alohida ajratib olib, uni funksiya ko‘rinishida aniqlash imkoni mavjud.

Funksiya bu – C++ tilida masala yechishdagi kalit elementlaridan biridir.

Funksiya parametrlari va argumentlari.

Dasturda ishlatiladigan har qanday funksiya e‘lon qilinishi kerak. Odatda funksiyalar e‘loni sarlavha fayllarda e‘lon qilinadi va #include direktivasi yordamida dastur matniga qo‘shiladi. Funksiya e‘lonini funksiya prototipi tavsiflaydi (ayrim hollarda signatura deyiladi). Funksiya prototipi quyidagi ko‘rinishda bo‘ladi:

<qaytaruvchi qiymat turi> <funksiya nomi>(<parametrlar ro‘yxati >);

Bu yerda <qaytaruvchi qiymat turi> - funksiya ishlashi natijasida y tomonidan qaytaradigan qiymatning turi. Agar qaytariladigan qiymat turi ko‘rsatilmagan bo‘lsa, kelishuv bo‘yicha funksiya qaytaradigan qiymat turi int deb hisoblanadi, <parametrlar ro‘yxati>- vergul bilan ajratilgan funksiya parametrlarining turi va nomlari ro‘yxati. Parametr nomini yozmasa ham bo‘ladi. Ro‘yxat bo‘sh bo‘lishi ham mumkin. Funksiya prototiplariga misollar:

int almashsin(int,int);

double max(double x, double y);

void func();

void chop_etish(void);

Funksiya prototipi tushirib qoldirilishi mumkin, agar dastur matnida funksiya aniqlanishi uni chaqiradigan funksiyalar matnidan oldin yozilgan bo‘lsa. Lekin bu holat yaxshi uslub hisoblanmaydi, ayniqsa o‘zaro bir-biriga murojaat qiluvchi funksiyalarni e‘lon qilishda muammolar yuzaga kelishi mumkin.

Funksiya aniqlanishi – funksiya sarlavhasi va figurali qavsga (‘{’,’}’) olingan qandaydir amaliy mazmunga ega tanadan iborat bo‘ladi. Agar funksiya qaytaruvchi turi void turidan farqli bo‘lsa, uning tanasida albatta mos turdagi parametrga ega return operatori bo‘lishi shart. Funksiya tanasida bittadan ortiq return operatori bo‘lishi mumkin. Ularning ixtiyoriy birortasini bajarish orqali funksiyadan chiqib ketiladi. Agar funksiyaning qiymati dasturda ishlatilmaydigan bo‘lsa, funksiyadan

chiqish uchun parametrsiz return operatori ishlatilishi mumkin yoki umuman return ishlatilmaydi. Oxirgi holda funksiyadan chiqish – oxirgi yopiluvchi qavsga yetib kelganda ro‘y beradi.

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

Odatda dasturda funksiya ma‘lum bir ishni amalga oshirish uchun chaqiriladi. Funksiyaga murojaat qilganda, u qo‘yilgan masalani yechadi va o‘z ishini tugatishida qandaydir qiymatni natija sifatida qaytaradi

Funksiyani chaqirish uchun uning nomi va undan keyin qavs ichida argumentlar ro‘yxati beriladi:

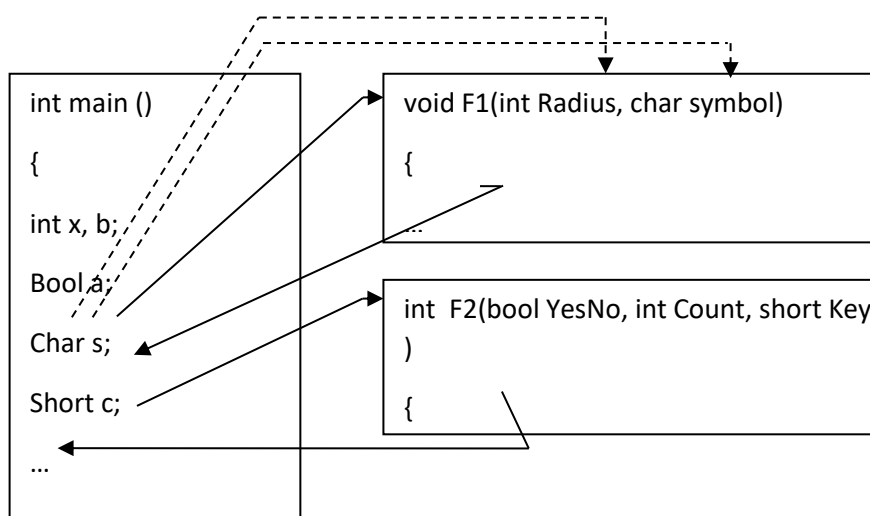
<funksiya nomi>(<argument₁>, <argument₂>, ..., <argument_n>);

Bu yerda har bir <argument> - funksiya tanasiga uzatiladigan va keyinchalik hisoblash jarayonida ishlatiladigan o‘zgaruvchi, ifoda yoki o‘zgarmasdir. Argumentlar ro‘yxati bo‘sh bo‘lishi mumkin.

Funksiyalar ham o‘z tanasida boshqa funksiyalarni, o‘zini ham chaqirishi mumkin. O‘z tanasida o‘zini chaqiradigan funksiyalarga rekursiv funksiyalar deyiladi.

Oldingi boblarda ta’kidlab o‘tilganidek, C++ tilidagi har qanqay dasturda albatta main() bosh funksiyasi bo‘lishi kerak. Ayni shu funksiyani yuklagich tomonidan chaqirilishi bilan dastur bajarilishi boshlanadi.

Quyidagi rasmda bosh funksiyadan boshqa funksiyalarni chaqirish va ulardan qaytish sxemasi ko‘rsatilgan.



14.1-rasm.

Bosh funksiyadan boshqa funksiyalarni chaqirish va qaytish

Dastur main() funksiyasini bajarishdan boshlanadi va “f1(x,y);” – funksiya chaqirishgacha davom etadi va keyinchalik boshqaruv f1(x,y) funksiya tanasidagi amallarni bajarishga o’tadi. Bunda Radius parametrining qiymati sifatida funksiya x o’zgaruvchi qiymatini, Symbol parametri sifatida y o’zgaruvchisining qiymati ishlatiladi. Funksiya tanasi return operatorigacha bajariladi. Return operatori boshqaruvni main() funksiyasi tanasidagi f1() funksiyasi chaqirilgan operatoridan keyingi operatorga o’tishni ta’minlaydi, ya’ni funksiyadan qaytish ro’y beradi. Shundan keyin main() funksiyasi operatorlari bajarilishda davom etadi va “f2(a,b,c);” – funksiya chaqirishi orqali boshqaruv f2() funksiya tanasiga o’tadi va hisoblash jarayonida mos ravishda YesNo sifatida a o’zgaruvchisining, cout sifatida o’zgaruvchisining va key sifatida c o’zgarchuvchisining qiymatlari ishlatiladi. Funksiya tanasidagi return operatori yoki oxirgi operator bajargandan keyin avtomatik ravishda bosh funksiyaga qaytish amalga oshiriladi.

Aksariyat hollarda main() funksiyasining parametrlar ro’yxati bo’sh bo’ladi. Agar dasturni ishga tushirishda, buyruq satrida ma’lum bir parametrlarni uzatish (berish) zarur bo’lsa, main() funksiyasining sintaksisi o’zgaradi:

```
int main(int argc, char* argv[]);
```

Bu yerda argc – uzatiladigan parametrlar soni, argv[]- bir-biridan punktuatsiya belgilari (va probel) bilan ajratilgan parametrlar ro’yxatini o’z ichiga olgan massivga ko’rsatkich.

Quyida funksiyalarni e’lon qilish, chaqirish va aniqlashga misollar keltirilgan:

```
// funksiyalar e’loni
```

```
int mening_funksiyam(int Number, float Point);
```

```
char Belgini_uqish();
```

```
void bitni_urnatish(short Num);
```

```
void Amal_yoq(int,char);_
```

```
// funksiyalarni chaqirish
```

```
result= mening_funksiyam (Varb1,3.14);
```

```
symb=Belgini_uqish();
```

```
bitni_urnatish(3);
```

```
Amal_yoq(2,Smb1);
```

```
// funksiyalarni aniqlash
```

```
int mening_funksiyam (int Number, float Point);
```

```

{int x;
...
return x;}

char Belgini_uqish();
{
char Symbo1;
cin>>Symbo1;
return Symbo1;
};

void bitni_urnatish(short number)
(global_bit=global_bit|number;};

void Amal_yoq(int x, char ch){};

```

Funksiyaning dasturdagi o‘rnini yanada tushunarli bo‘lishi uchun son kvadratini hisoblash masalasida funksiyadan foydalanishni ko‘raylik.

Funksiya prototipini sarlavha.h sarlavha faylida joylashtiramiz:

```
long Son_Kvadrati(int);
```

Asosiy dasturga ushbu sarlavha faylini qo‘shish orqali Son_Kvadrati() funksiya e’loni dastur matniga kiritiladi:

```

#include <iostream.h>
#include "sarlavha.h"

int main()
{
int uzgaruvchi=5;
cout<<Son_kvadrati(uzgaruvchi);
return 0;
}

long Son_kvadrati(int x)
{
return x*x;
}

```

```
}
```

Xuddi shu masalani sarlavha faylidan foydalanmagan holda, funksiya e'lonini dastur matniga yozish orqali ham hal qilish mumkin:

```
#include <iostream.h>

long Son_kvadrati(int);

int main()
{
    int uzgaruvchi=5;
    cout<<Son_kvadrati(uzgaruvchi);
    return 0;
}

long Son_kvadrati(int x)
{
    return x*x;
}
```

Dastur ishlashida o'zgarish bo'lmaydi va natija sifatida ekranga 25 sonini chop etadi.

Masala. Ikkita tub son «egizak» deyiladi, agar bir-biridan farqi 2 teng bo'lsa (masalan, 41 va 43 sonlari). Berilgan natural n uchun $[n..2n]$ kesmadagi barcha «egizak» sonlar juftliklari chop etilsin. Masalani yechish uchun berilgan k sonini tub son yoki yo'qligi aniqlovchi mantiqiy qiymat qaytaruvchi funksiya tuzish zarur bo'ladi. Funksiyada k soni $2..k/2$ gacha sonlarga bo'linadi, agar k bu sonlarning birortasiga ham bo'linmasa, u tub son hisoblanadi va funksiya true qiymatini qaytaradi. Bosh funksiya, berilgan n uchun $[n..2n]$ oraliqdagi $(n, n+2)$, $(n+1, n+3), \dots, (2n-2, 2n)$ son juftliklarini tub sonlar ekanligi tekshiriladi va shart qanoatlantirgan juftliklar chop etiladi.

Dastur matni:

```
bool TubSon (unsigned long k);

int main()
{
    unsigned long n,i;
    unsigned char egizak=0;
```

```

cout<<"n->";
cin>>n;
cout<<'['<<n<<".."<<2*n<<']';
for(i=n; i<=2*n-2; i++)
if(TubSon (i) && TubSon(i+2))
{
if(!egizak)
cout<<"oralig'idagi egizak tub sonlar☺\n";
else cout<<"; ";
egizak=1;
cout<<'{'<<i<<','<<i+2<<'}';
};
if(!egizak)
cout<<"oralig'idagi egizak tub sonlar mavjud emas.";
else cout<<'. ';
return 0;
}
bool TubSon(unsigned long k);
{
unsigned long m;
for (m=2; m<=k/2; m++)
if (k%m==0) return false;
return true;
}

```

Natural n soni uchun 100 kiritilsa, dastur quyidagi sonlar juftliklarini chop qiladi:

```

[100..200] orag'ida egizak tub sonlar:
{101,103}; {107,109}; {137,139}; {149,151}; {179,181};
{191,193}; {197,199}.

```

Kelishuv bo'yicha argumentlar.

C++ tilida funksiya chaqirilganda ayrim argumentlarni tushirib qoldirish mumkin. Bunga funksiya prototipida ushbu parametrlarni kelishuv bo'yicha qiymatini ko'rsatish orqali erishish mumkin.

Masalan, quyida prototipi keltirilgan funksiya turli chaqirishga ega bo'lishi mumkin:

```
//funksiya prototipi
void Butun_son(int I, bool bayroq=true, char belgi='n');
//funksiyani chaqirish variantlari
Butun_son (1,false,'a');
Butun_son (2,false);
Butun_son (3);
```

Birinchi chaqiruvda barcha parametrlar mos argumentlar orqali qiymatlarini qabul qiladi, ikkinchi holda I parametri 2 qiymatini, bayroq taticric false qiymatini va belgi o'zgaruvchisi kelishuv bo'yicha 'n' qabul qiladi.

Kelishuv bo'yicha qiymat berishning bitta sharti bor – parametrlar ro'yxatida kelishuv bo'yicha qiymat berilgan parametrlardan keyingi parametrlar ham kelishuv bo'yicha qiymatga ega bo'lishlari shart. Yuqoridagi misolda I parametri kelishuv bo'yicha qiymat qabul qilingan holda bayroq yoki belgi parametrlari qiymatsiz bo'lishi mumkin emas. Misol tariqasida berilgan sonni ko'rsatilgan aniqlikda chop etuvchi dasturni ko'raylik. Qo'yilgan masalani yechishda sonni darajaga oshirish funksiyasi – pow() va suzuvchi nuqtali uzun sondan

modul olish fabs1() funksiyasidan foydalaniladi. Bu funksiyalar prototipi<<math.h>> sarlavha faylida joylashgan (3-ilova qarang):

```
#include <iosream.h>
#include <math.h>
void chop_qilish (double Numb, double aniqlik=1,bool bayroq=true);
int main()
{double Mpi=-3.141592654;
 chop_qilish(Mpi,4, false);
 chop_qilish(Mpi,2);
 chop_qilish(Mpi);
 return 0;
```

```

}

void chop_qilish (double Numb, double aniqlik=1, bool bayroq=true)
{if (!bayroq)Numb=fabs1(Numb);
Numb=(int)(Numb*pow(10,aniqlik));
Numb=Numb/pow(10,aniqlik);
cout<<Numb<<'\n';
}

```

Dasturda sonni turli aniqlikda (Aniqlik parametri qiymati orqali) chop etish uchun har xil variantlarda Chop_qilish() funksiyasi chaqirilgan. Dastur ishlashi natijasida ekranda quyidagi sonlar chop etiladi:

3.1415

-3.14

-3.1

Parametrning kelishuv bo'yicha beriladigan qiymati o'zgarmas, global o'zgaruvchi yoki qandaydir funksiya qaytaradigan qiymat bo'lishi mumkin.

Lokal va global o'zgaruvchilar.

O'zgaruvchilar funksiya tanasida yoki undan tashqarida e'lon qilinishi mumkin. Funksiya ichida e'lon qilingan o'zgaruvchilarga lokal o'zgaruvchilar deyiladi. Bunday o'zgaruvchilar xotiradagi dastur stekida joylashadi va faqat o'zi e'lon qilingan funksiya tanasida amal qiladi. Boshqaruv asosiy funksiyaga qaytishi bilan lokal o'zgaruvchilar uchun ajratilgan xotira bo'shatiladi (o'chiriladi).

Har bir o'zgaruvchi o'zining amal qilish sohasi va yashash vaqti xususiyatlari bilan xarakterlanadi.

O'zgaruvchi amal qilish sohasi deganda o'zgaruvchini ishlatish mumkin bo'lgan dastur sohasi (qismi) tushuniladi. Bu tushuncha bilan o'zgaruvchining ko'rinish sohasi uzviy bog'langan. O'zgaruvchi amal qilish sohasidan chiqqanda ko'rinmay qoladi. Ikkinchi tomondan, o'zgaruvchi amal qilish sohasida bo'lishi, lekin ko'rinmasligi mumkin. Bunda ko'rinish sohasiga ruxsat berish amali «::» yordamida ko'rinmas o'zgaruvchiga murojat qilish mumkin bo'ladi.

O'zgaruvchining yashash vaqti deb, u mavjud bo'lgan dastur bo'lagining bajarilishiga ketgan vaqt intervaliga aytiladi.

Lokal o'zgaruvchilar o'zlari e'lon qilingan funksiya yoki blok chegarasida ko'rinish sohasiga ega. Blokdagi ichki bloklarda xuddi shu nomdagi o'zgaruvchi

e'lon qilingan bo'lsa, ichki bloklarda bu lokal o'zgaruvchi ham amal qilmay qoladi. Lokal o'zgaruvchi yashash vaqti – blok yoki funktsiyani bajarish vaqti bilan aniqlanadi. Bu hol shuni anglatadiki, turli funktsiyalarda bir-biriga umuman bog'liq bo'lmagan bir xil nomdagi lokal o'zgaruvchilarni ishlatish mumkin.

Quyidagi dasturda main() va sum() funktsiyalarida bir xil nomdagi o'zgaruvchilarni ishlatish ko'rsatilgan. Dasturda ikkita sonning yig'indisi hisoblanadi va chop etiladi:

```
#include <iostream.h>
```

```
// funktsiya prototipi
```

```
int sum (int a; int b);
```

```
int main()
```

```
{
```

```
// lokal o'zgaruvchi
```

```
int x=r;
```

```
cout<<sum(x,y);
```

```
return 0;
```

```
}
```

```
int sum(int a, int b)
```

```
{
```

```
// lokal o'zgaruvchi
```

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

```
return x;
```

```
}
```

Global o'zgaruvchilar dastur matnida funktsiya aniqlanishidan tashqarida e'lon qilinadi va e'lon qilingan joyidan boshlab dastur oxirigacha amal qiladi

```
#include <iostream.h>
```

```
int f1();
```

```
int f2();
```

```
int main()
```

```
{
```

```

cout<<f1()<<" "<<f2()<<endl;

return 0

}

int f1()

{

return x;// kompilyatsiya xatosi ro'y beradi

}

int x=10; //global o'zgaruvchi e'loni

int f2()

{

return x*x;

}

```

Yuqorida keltirilgan dasturda kompilyatsiya xatosi ro'y beradi, chunki f1() funksiya uchun x o'zgaruvchisi noma'lum hisoblanadi.

Dastur matnida global o'zgaruvchilarni ular e'lonidan keyin yozilgan ixtiyoriy funksiyadan ishlatish mumkin. Shu sababli, global o'zgaruvchilar dastur matnining boshida yoziladi. Funksiya ichidan global o'zgaruvchiga murojat qilish uchun funksiyada uning nomi bilan mos tushadigan lokal o'zgaruvchilar bo'lmasligi kerak. Agar global o'zgaruvchi e'lonida unga boshlang'ich qiymat berilmagan bo'lsa, ularning qiymati 0 hisoblanadi. Global o'zgaruvchining amal qilish sohasi uning ko'rinish sohasi bilan ustma-ust tushadi.

Shuni qayd etish kerakki, dastur tuzuvchilar imkon qadar global o'zgaruvchilarni ishlatmaslikka harakat qiladilar, chunki bunday o'zgaruvchilar qiymatini dasturning ixtiyoriy joyidan o'zgartirish imkoniyati mavjudligi sababli dastur ishlashida mazmunan xatolar yuzaga kelishi mumkin. Bu fikrimizni tasdiqlovchi dasturni ko'raylik.

```

#include <iostream.h>

//global o'zgaruvchi e'loni

int test=100;

void Chop_qilish(void);

int main()

{

```

```

//lokal o'zgaruvchi e'loni
int test=10;

//global o'zgaruvchi chop qilish funksiyasini chaqirish
Chop_qilish();

cout<<"Lokal o'zgaruvchi: "<<test<<"\n";

return 0;

}

void Chop_qilish(void)
{
cout<<"global o'zgaruvchi: "<<test<<"\n";

}

```

Dastur boshida test global o'zgaruvchisi 100 qiymati bilan e'lon qilinadi. Keyinchalik, main() funksiyasida test nomi bilan lokal o'zgaruvchisi 10 qiymati bilan e'lon qilinadi. Dasturda, Chop_qilish() funksiyasiga murojaat qilinganida, asosiy funksiya tanasidan vaqtincha chiqiladi va natijada main() funksiyasida e'lon qilingan barcha lokal o'zgaruvchilarga murojaat qilish mumkin bo'lmay qoladi. Shu sababli Chop_qilish() funksiyasida global test o'zgaruvchisining qiymatini chop etiladi. Asosiy dasturga qaytilgandan keyin, main() funksiyasidagi lokal test o'zgaruvchisi global test o'zgaruvchisini «berkitadi» va lokal test o'zgaruvchini qiymati chop etiladi. Dastur ishlashi natijasida ekranga quyidagi natijalar chop etiladi:

Global o'zgaruvchi: 100

Lokal o'zgaruvchi: 10

Nazorat savollari

1. Funksiya prototipi nima?
2. Funksiya parametrlari.
3. Funksiya argumentlari.
4. void turi.
5. Statik o'zgaruvchilarga izoh bering.
6. Lokal va global o'zgaruvchilar, ular o'rtasidagi farq.
7. Rekursiv funksiyalar.
8. Kelishuv bo'yicha qiymat berish.
9. O'zgaruvchining amal qilish sohasi.
10. O'zgaruvchining yashash vaqti.
11. O'zgaruvchining ko'rinish sohasi.

Mavzu: Grafika va multimedia dasturlash tizimlarida qo'llash.

Reja:

1. Ekran adapterlarining xarakteristikalar. Ularning imkoniyatlari.
2. Grafik modulining imkoniyatlari va ulardan foydalanish. Koordinatalar, darchalar, ranglar, fon ranglari va obyektlar. Nuqta, to'g'ri chiziq, ko'p burchak, yoylar, aylanalar hosil qilish. Grafiklarni hosil qilish.
3. Ekranni masshtablash. Obyektlarni bo'yash. Obyektni harakatga keltirish – animatsiya. Maxsus protseduralar.

Tayanch iboralar: C++Builder da grafika, rasm, grafika yoki asos, grafik fayllar, palitra, palitra o'zgarishiga reaksiya, bit obrazlari, o'zgarishlarga reaksiya.

C++Builder da grafikani qo'llab-quvvatlash: C++Builder dasturi Windows GDI funksiyalarini turli darajalarda inkapsulalaydi. Bu o'rinda bir usul muhim bo'lib, uning vositasida grafik komponentalar o'z tasvirlarini monitor ekranida taqdim etadi. GDI funktsiyasi to'g'ridan-to'g'ri chaqirilganda, ushbu grafik komponentalarga qurilma konteksti deskriptori (device context handle) ni uzatish kerak. Bu deskriptor siz tanlab olgan rassomchilik ashyolari - perolar, mo'yqalamlar, shriftlarni chiqarib beradi. Grafik tasvirlar bilan ishlash tugagach, siz qurilma kontekstini dastlabki holatga keltirib qo'yishga majbursiz va shundan keyingina undan ozod bo'lishingiz mumkin.

Shu darajada detallashtirilgan grafika bilan ishlashga sizni majbur qilish o'rniga, S++Builder grafik komponentalarning Canvas (Kanva - Asos) xususiyati vositasida sodda va tugal interfeysni taklif qiladi. Bu xususiyat qurilmaning to'g'ri kontekstini nomlaydi (initsiallashtiradi) hamda siz rasm chizishni to'xtatgan kerakli vaqtda uni ozod qiladi. Asos pero, mo'yqalam va shrift tavsiflari nomidan ish ko'radigan berilgan xususiyatlarga ega.

Grafik komponentalar bilan ishlashda foydalanuvchi amalga oshirishi lozim bo'lgan yagona ish - bu qo'llanayotgan rasm chizish ashyolarining tavsiflarini aniqlash. Ashyolarni yaratish, tanlash va ozod qilishda sizdan tizim zahiralarini kuzatib borish talab qilinmaydi. Asosning o'zi bu haqda qayg'uradi.

Grafika bilan ishlashda C++Builder namoyon qiladigan afzalliklardan biri - bu tizimning grafik zahiralar uchun keshlangan xotiradan foydalanish. Aytaylik, agar sizning dasturingiz biron-bir konkret turdagi peroni qaytadan yaratsa, qo'llasa va ozod etsa, siz ushbu perodan har gal foydalanganingizda bu qadamlarni takrorlashingizga to'g'ri keladi. C++Builder kesh-xotiradan grafik zahiralarini

saqlash uchun foydalanar ekan, tez-tez qo'llanadigan rasm chizish ashyosi har gal yangitdan qayta yaratilmay, balki kesh-xotiradan takroran tanlab olinishi ehtimoli oshadi. Buning natijasida sizning grafik ilovangizning takrorlanayotgan operatsiyalarining samarasi ancha ortishi aniq.

Quyida Windows uchun ilovalar darchasida ko'k kontur bilan aylantirilgan sariq ellipsni chizish masalasini hal qilayotgan kod fragmenti keltirilgan. Bu masala rasm chizish asosi vositasida yechiladi.

```
void_fastcall TFormPaint(TObject*Sender)
```

```
{
    Canvas->Pen->Color=clBlue;      // Canvas->Brush->Color=clYellow
    konturining rangini tanlash; // Canvas->Ellipse(10, 20, 50, 50) ichining rangini
    tanlash; // ellips rasmini chizish
}
```

Asosdan foydalanish: Asosning ob'ekli sinfi Windows ning grafik funksiyalarini, alohida chiziqlar, shakllar va matnni chizish uchun mo'ljallangan yuqori darajadagi funksiyalardan boshlab, inkapsulalaydi. Keyin rasm chizish uchun asos bilan ishlaydigan o'rta darajadagi usul va metodlar keladi. Nihoyat, quyi darajada Windows GDI funksiyalarining o'ziga kirish ta'minlanadi. Quyidagi jadvalda asosning asosiy metod va usullarining umumlashtirilgan tavsiflari berilgan.

Daraja	Xatti-harakatlar	Metodlar	Xususiyatlar
Yuqori	Peroning joriy pozitsiyasini belgilaydi	MoveTo	PenPos
	To'g'ri chiziqni berilgan nuqtagacha chizadi		
	Berilgan o'lchamdagi to'g'ri to'rturchak chizadi		
	Berilgan o'lchamda ellips chizadi		
	Matn satrini chizadi		
	Matn satrini chiqarish uchun ajratilgan bo'yni beradi		
	Matn satrini chiqarish uchun ajratilgan enni beradi		
	To'rtburchak ichida matn chiqarish		
	Ko'rsatilgan to'rtburchak ichiga rang va joriy mo'yqalam teksturasini quyish		
	Asos sohasiga (ixtiyoriy shakldagi) berilgan rang quyish		

O'rta	Peroning rangi, uslubi, eni va rejimini o'rnatish uchun qo'llanadi		
	Grafik shakllar va asos fonini quyishda rang va teksturani o'rnatish uchun qo'llanadi		
	Berilgan rang, o'lcham va uslubdagi shriftni o'rnatish uchun qo'llanadi		
	Berilgan asos pikselini o'qish va rangini yozish uchun qo'llanadi		
	CopyMode rejimida asosning to'rtburchak sohasidan nusxa ko'chiradi		
	Asosning to'rtburchak sohasidan rangni almashtirib nusxa ko'chiradi		
	Asosning berilgan joyida bit obrazini, piktogramma va metafilelning rasmini chizadi		
	Bit obrazini, piktogramma va metafilelning rasmini to'rtburchakni to'liq to'ldiradigan qilib chizadi		
Quyi	Windows GDI funksiyalarini chaqirishda parametr sifatida qo'llanadi		

Rasmlar bilan ishlash: C++Builder muhitida bajariladigan grafik ishlarning asosiy mundarijasi asosingiz shaklida yoki unda joylashtirilgan boshqa komponentalarda bevosita rasm chizishdan iborat. C++Builder shuningdek tashqi tasvirlar - bit obrazlari, metafilellar, piktogrammalar, shu jumladan, palitralarni avtomatik boshqarishning qo'llab-quvvatlanishiga xizmat ko'rsatadi.

C++Builder muhitida rasmlar bilash ishlashda uchta muhim jihatni e'tiborga olish kerak.

Rasm, grafika yoki asos: C++Builder muhitida grafikaga aloqador bo'lgan uch xil ob'ekt mavjud:

-Asos shaklda, grafik komponentada, printerda yoki boshqa biron bit obrazida rasm chizish uchun mo'ljallangan yuzaning bit kartasidan iborat. Asos mustaqil ob'ekt emas, u hamma vaqt biron-bir boshqa grafik ob'ektning xususiyati bo'lib xizmat qiladi.

-Grafika: biron-bir fayl yoki zahiraning (bit obrazi, piktogramma yoki metafilel) ning satri tasviridan iborat. C++Builder dasturi TGraphic bazaviy sinfning hosilalari bo'lgan TVitmap, Ticon va TMetafile ob'ektlari sinflarini aniqlaydi.

Albatta, siz o'zingiz grafik ob'ektlarning shaxsiy sinflarini e'lon qilishingiz mumkin. Tgraphic sizning ilovangizda grafikaning barcha turlaridan foydalanish uchun minimal standart interfeys taqdim etadi.

-Rasm - grafika uchun konteyner bo'lib, u grafik ob'ektlarning har qanday sinflarini taqdim etadi. SHunday qilib, TRicture konteynerli sinf bit obrazi, piktogramma, metafile yoki foydalanuvchi tomonidan belgilangan boshqa biron grafik turga ega bo'lishi mumkin. Ilova esa «rasm» ob'ekti vositasida konteynerning barcha ob'ektlariga standartlashgan tarzda murojaat qilishi mumkin. Darhaqiqat, tasvirlarni boshqarish komponentalarining ko'pchiligi ob'ektlar turidagi TRicture ning Picture xususiyatiga ega bo'lib, u har xil turdagi grafik tasvirlarni taqdim etish imkoniyatiga ega.

Shuni ham ta'kidlab o'taylik: «rasm» ob'ekti hamma vaqt biron-bir grafikaga ega bo'lib, bu grafika o'z navbatida, asosga ega bo'lishga ehtiyoj sezadi (asosga ega bo'lgan yagona standart grafik sinf bu TVitmap dir). Odatda, rasm bilan ishlar ekansiz, siz grafik ob'ektning faqat TRicture konteyneri orqali kirish uchun ochiq bo'lgan qismi bilangina ish ko'rasiz. Agar sizga konkret grafik ob'ektga kirish huquqini ko'rsatish lozim bo'lib qolsa, ushbu rasmning Graphic xususiyatiga murojaat qiling.

Grafik fayllar: Ilovangiz ishlayotgan har bir daqiqada C++Builder standart formatdagi tasvirlar fayllarida rasmlar va grafiklarning yuklanishi va saqlanishini qo'llab-quvvatlaydi. Saqlanayotgan va dastlabki faylning nomlari bir-biriga mos kelishi hamda bir-biridan farq qilishi mumkin.

Tasvirni fayldan rasmga yuklatish uchun LoadFromFile rasm metodidan foydalaning. Faylda tasvirni saqlash uchun rasm metodidan foydalaning Ushbu metodlarning yagona parametri bu fayl nomi bo'ladi. LoadFromFile metodi fayl kengayishidan o'zi yaratadigan va saqlaydigan grafik ob'ekt turini aniqlash uchun foydalanadi. LoadToFile metodi saqlanayotgan grafik ob'ekt turiga mos keladigan kengayishli faylni saqlaydi.

Navbatdagi matnda yo'riqnoma (instruksiya) mujassam bo'lib, uni siz bit obrazini komponentali rasm ob'ektiga yuklash uchun kod moduli matniga yozib qo'yishingiz kerak.

```
void_fastcall TFormI::FormCreate(TObject *Sender) {  
    Image1->Picture->LoadFromFile("c:\\windows\\clouds.bmp");  
}
```

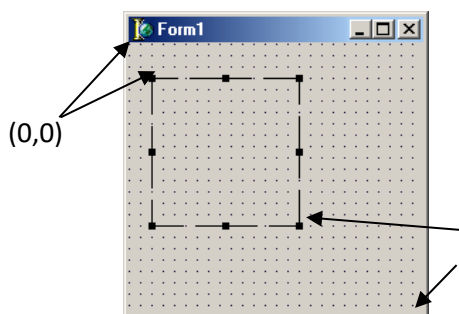
Rasm .bmp bit orazlari fayllarining standart kengayishini tanib oladi hamda o'z grafikasini TVitmap sinf ob'ekti sifatida yaratadi, keyin esa ko'rsatilgan ismli fayldan tasvirni yuklash LoadFromFile metodini chaqiradi.

Borland C++ dasturchiga turli xildagi sxemalar, chizmalar va illyustrasiyalar bilan ishlash imkoniyatlarini beradi. Dastur grafikani ob`ekt (forma yoki komponent Image) sirtida hosil qiladi. Ob`ekt sirti Canvas xususiyatiga mos keladi. Grafik element (to`g`ri chiziq, aylana, to`g`ri to`rtburchak va x.k.)larni ob`ekt yuzasida hosil qilish uchun Canvas dan foydalaniladi.

Masalan, `Form1->Canvas->Rectangle(10,10,50,50)` instruksiyasi dastur oynasida to`g`ri to`rtburchak hosil qiladi.

Chizma hosil bo`luvchi sirt.

Yuqorida aytib o`tilganidek, grafikani hosil qiluvchi sirt (yuza) Canvas xususiyatiga to`g`ri keladi. O`z navbatida Canvas xususiyati Tcanvas tipidagi ob`ektdir. Bu tip uslublari grafik primitivlarni (nuqta, chiziq, aylana va x.k.) hosil bo`lishini ta`minlaydi, xususiyati esa hosil bo`luvchi grafikani xarakteristikalarini: rangi, chiziq qalinligi va turi; bo`yaluvchi hududni rangi va ko`rinishini; harfni xarakteristikalarini beradi. Canvas «sirt», «chizish uchun yuza» sifatida tarjima qilinadi. Chizish yuzasi alohida nuqta – piksellardan tashkil topadi. Pikselni joylashuvi gorizontal (X) va vertikal (Y) koordinatalar bilan xarakterlanadi. CHap yuqoridagi nuqta koordinatasi (0,0). Koordinatalar yuqoridan pastga va chapdan o`ngga qarab o`sib boradi (15.1-rasm).



15.1-rasm. Chizish yuzasi nuqta koordinatalari.

Qalam va mo`yqalar.

Odatda rassom surat chizish uchun qalam va mo`yqalamdan foydalanadi. Borland C++ ning 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.

Qalam va mo`yqalam grafikani chizish yuzasida hosil qilishda mos ravishda `Pen`(qalam) va `Brush`(mo`yqalam) xususiyatlariga xosdir. Shu bilan birga ular `TPen` va `TBrush` tiplariga tegishlidir.

Qalam.

Qalamdan nuqta, chiziq, geometrik shakllar: to`g`ri to`rtburchak, aylana, ellips va h.k. larni chizishda qurol sifatida foydalaniladi. TPen ob`ekt xususiyati 5.1-jadvalda keltirilgan.

15.1-jadval

Xususiyat	Vazifasi
Color	CHiziq (kontur) rangi
Width	CHiziq qalinligi
Style	CHiziq ko`rinishi
Mode	Tasvirlash rejimi

Color xususiyati chizuvchi qalam rangini belgilaydi. Quydagi 5.2-jadvalda PenColor xususiyatlari keltirilgan:

15.2-jadval

Konstanta	Rang	Konstanta	Rang
clBlack	qora	clSilver	kumushrang
clMaroon	kashtanrang	clRed	qizil
clGreen	yashil	clLime	salatrang
clOlive	olivkoyiy	clBlue	ko`k
clNavy	to`q ko`k	clFuchsia	Fuchsia
clPurple	atirgularang	clAqua	yorug` ko`k
clTeal	Teal	clWhite	oq
clGray	kulrang		

Width xususiyati chizuvchi qalam qalinligini (pikselda) belgilaydi.

Masalan, Canvas->Pen->Width=2 chiziq qalinligi 2 pikselga teng bo`ladi.

Style xususiyati chiziluvchi chiziqning turini belgilaydi. Style komponentlari 15.3-jadvalda keltirilgan.

Konstanta	CHiziq 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
---------	------------------

Mo`yqalam.

Mo`yqalam (Canvas.Brush)dan yopiq sohalarni to`ldirish uchun foydalaniladi, masalan, geometrik shakllarni bo`yash va x.k. Mo`yqalam ob`ekt sifatida quyidagi ikki xususiyatni o`z ichiga oladi:

Color – bo`yaluvchi soha rangi

Style – to`ldiruvchi soha tipi

Masalan, konturning ichki sohasi bo`yalishi yoki shtrixlanishi mumkin.

Color xususiyati sifatida Tcolor ning barcha o`zgarmaslaridan foydalanish mumkin. Style xususiyatlari 5.4-jadvalda keltirilgan.

15.4-jadval

Konstanta	Bo`yaluvchi soha tipi
bsSolid	to`liq
bsClear	Bo`yalmaydi
bsHorizontal	gorizontal shtrixlash
bsVertical	vertikal shtrixlash
bsFDiagonal	oldinga egilgan diagonal shtrixlash
bsBDiagonal	orqaga egilgan diagonal shtrixlash
bsCross	gorizontal-vertikal setkali shtrixlash
bsDiagCross	diagonal setkali shtrixlash

Matn hosil qilish.

Grafik ob`ekt sirtida matnni hosil qilish uchun TextOut dan foydalaniladi. TextOut ning yozilish formati quyidagicha:

Ob`ekt->Canvas->TextOut(x,y,Text);

Bu yerda

Ob`ekt – matn hosil bo`luvchi ob`ekt nomi;

x,y – matn boshlanuvchi koordinata (5.2-rasm);

Text – hosil bo`luvchi belgi kattalikdagi matn yoki satrli o`zgaruvchi.



15.2-rasm. Matn hosil bo'luvchi soha koordinatasi

Hosil bo'luvchi matn belgilari Canvas ob'ektiga muvofiq keluvchi Font xususiyati orqali ifodalanadi. Font xususiyati TFont ob'ektiga tegishli bo'lib, 5.5-jadvalda belgi xarakteristikalar va qo'llaniluvchi uslublari keltirilgan.

15.5-jadval

Xususiyat	Aniqlanishi
Name	Foydalaniluvchi shrift. Qiymat sifatida shrift nomi yoziladi, masalan, Arial Cyr
Size	punktlarda ifodalaniluvchi shrift o'lchami. Punkt-poligrafiyada qo'llaniluvchi o'lchov birligi bo'lib, u taxminan 1/72 dyuym1ga teng
Style	belgini yozish usuli, quyidagicha bo'lishi mumkin: oddiy, qalin, kursiv, ostiga chizilgan, ustiga chizilgan. Bular quyidagi konstantalar yordamida amalga oshiriladi: fsBold(qalin), fsItalic(kursiv), fsUnderline(ostiga chizilgan), fsStrikeOut(ustiga chizilgan). style bir nechta usullarni kombinasiya qilishi mumkin. Masalan, qalin kursiv holatini ifodalash: Ob'ekt->Canvas->Fon:=[fsBold, fsItalic]
Color	Belgi rangi. Qiymat sifatida TSolor konstantalaridan foydalanish mumkin.

Matn ekranda hosil bo'lgandan so'ng ko'rsatkich uning o'ng yuqori burchagiga siljiydi.

Ba'zida matndan so'ng biror ma'lumotni chiqarish kerak bo'lib qoladi. Agar matn uzunligi noma'lum bo'lsa ko'rsatkich turgan koordinatani aniqlash mushkul. Masalan «so'm» so'zini raqamdan keyin hosil qilish kerak bo'lsin. Bunday holatlarda ko'rsatkich turgan koordinatadan boshlab davom etish uchun PenPos dan foydalanishga to'g'ri keladi:

To'g'ri chiziq.

Borland C++ da to'g'ri chiziq hosil qilish uchun LineTo dan foydalaniladi. Uning yozilish formati quyidagicha:

Komponen->Canvas->LineTo(x,y)

LineTo to'g'ri chiziqni qalam (ko'rsatkich) to'rgan koordinatadan boshlab x,y – nuqtagacha chizadi. SHuning uchun chiziqning boshlang'ich nuqtasini kerakli joyga o'rnatib olish lozim bo'ladi. Bunda biz MoveTo ga murojaat qilamiz:

Komponen->Canvas->MoveTo(X0,Y0)

CHiziqning ko'rinishi (rangi, qalinligi va turi) Pen ob'ekti bilan ifodalanadi.

Aylana va ellips.

Ellipse uslubi ellips va aylana chizish uchun qo'llaniladi. Ellipse ning yozilish formati quyidagicha:

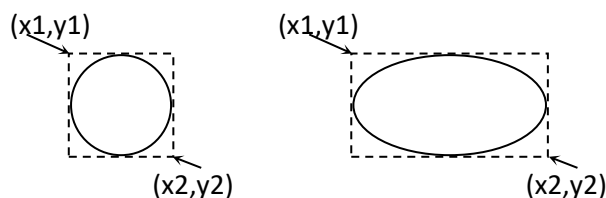
Ob'ekt->Canvas->Ellipse(x1,y1,x2,y2)

bu yerda,

ob'ekt – chizma hosil bo'luvchi ob'ekt nomi;

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 nuqtalarini koordinatalari (5.3-rasm).

Chiziqning ko'rinishi (rangi, qalinligi va turi) Pen ob'ekti bilan ifodalanadi.



15.3-rasm.

Yoy.

Yoy hosil qilish uchun Arc uslubidan foydalaniladi. Uning yozilish formati quyidagicha:

Ob'ek->Canvas->Arc(x1,y1,x2,y2,x3,y3,x4,y4)

bu yerda

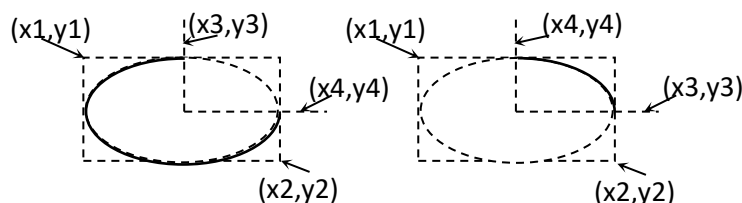
ob'ekt – yoy chiziluvchi ob'ekt nomi;

x1,y1,x2,y2 – hosil bo'luvchi yoyni davom ettirib hosil qilinuvchi ellips (aylana)ga tashqi chizilgan to'g'ri to'rtburchakning mos koordinatalari;

x_3, y_3 – yoyning boshlang'ich nuqtasi;

x_4, y_4 – yoyning tugash nuqtasi.

Shuni aytib o'tish lozimki, yoy soat strelkasi yo'nalishiga qarama-qarshi yo'nalishda chiziladi (5.4-rasm).



15.4-rasm.

Chiziqning ko'rinishi (rangi, qalinligi va turi) Pen ob'ekti bilan ifodalanadi.

To'g'ri to'rtburchak.

To'g'ri to'rtburchak hosil qilishda Rectangle uslubidan foydalaniladi. Uning yozilish formati quyidagicha:

Ob'ek->Canvas.Rectangle(x_1, y_1, x_2, y_2)

Bu yerda

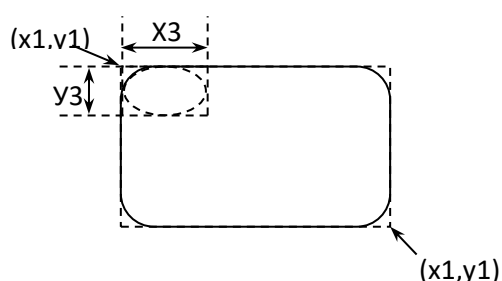
ob'ekt – tasvir hosil bo'luvchi ob'ekt nomi;

x_1, y_1, x_2, y_2 – to'g'ri to'rtburchakning mos ravishda yuqori chap va quyi o'ng burchak koordinatalari.

RoundRec uslubi ham to'g'ri to'rtburchak chizadi, faqat Rectangle dan farqi shundaki, uning burchaklari yumaloq (silliqlik) shaklda bo'ladi. YOzilish formati:

Ob'ekt->Canvas->RoundRec(x_1, y_1, x_2, y_2)

Bu yerda



ob'ekt – tasvir hosil bo'luvchi ob'ekt nomi;

x_1, y_1, x_2, y_2 – to'g'ri to'rtburchakning mos ravishda yuqori chap va quyi o'ng burchak koordinatalari;

x_3, y_3 – yumaloq hosil qilishda qo'llaniluvchi ellips o'lchamlari (15.5-rasm).

Ko'pburchak.

Polygon dan 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 esa shu nuqtalarni ketma-ket to'g'ri chiziqlar bilan tutashtirib chiqadi.

CHiziqning ko'rinishi (rangi, qalinligi va turi) Pen ob'ekti bilan ifodalanadi. Sektor.

Ellips yoki aylana sektorini hosil qilishda Pie uslubidan foydalaniladi. Pie ning umumiy yozilish formati:

Ob'ek->Canvas->Pie(x1,y1,x2,y2,x3,y3,x4,y4)

bu yerda

ob'ekt – yoy chiziluvchi ob'ekt nomi;

x1,y1,x2,y2 – hosil bo'luvchi sektorni davom ettirib hosil qilinuvchi ellips (aylana)ga tashqi chizilgan to'g'ri to'rtburchakning mos koordinatalari;

x3,y3 – sektorning boshlang'ich nuqtasi;

x4,y4 – sektorning tugash nuqtasi.

15.6-jadval

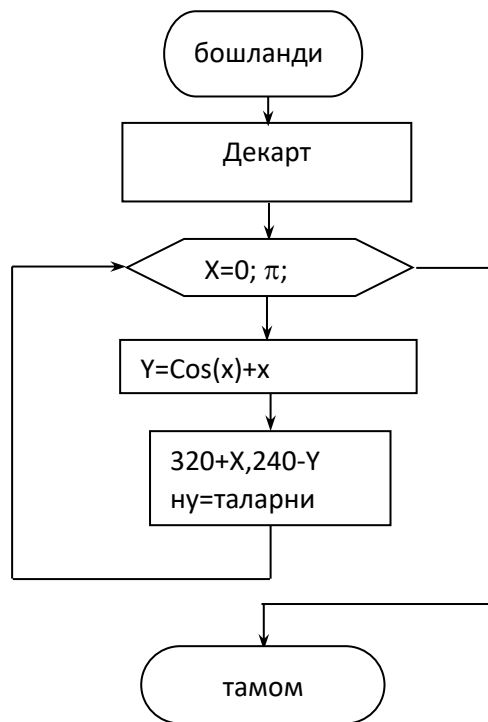
№	y=f(x) funksiyaning ko'rinishi	Argumentning o'zgarish oralig'i		Grafikdagi nuktalar sonin
		A	B	
	Sinx	$-\pi/2$	$\pi/2$	30
	Cosx	0	$3\pi/2$	40
	[Sinx]+[Cosx]	0	π	40
	[Sinx]-[Cosx]	0	π	40

Topshiriklar bajarish na'munasi.

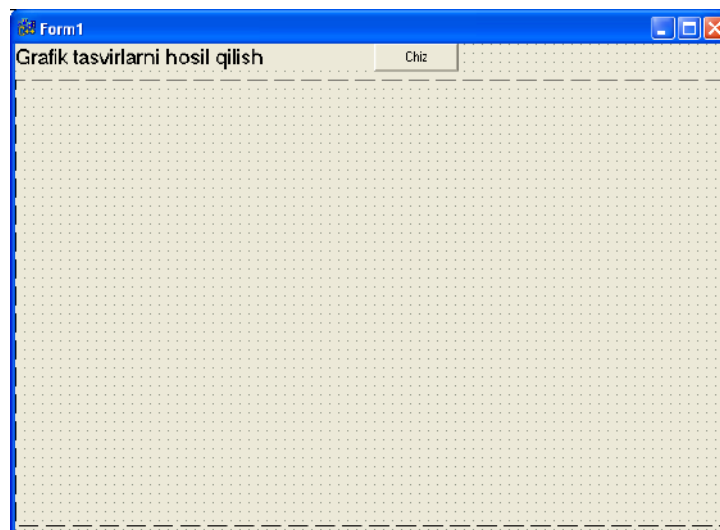
Bizga quyidagi funksiyaning grafigini yasash berilgan bo'lsin.

$U = \cos x + x \quad x \in [0; \pi]; \quad n=80,$

Avvalom bor biz bu masalaning algoritm blok-sxemasini tuzib olamiz. Ya'ni u quyidagicha bo'ladi:



Borland C++ tilidagi forma va dastur ko`rinishi



```

//-----
#include <vcl.h>
#include <math.h>
#pragma hdrstop
#include "Unit1.h"
//-----

#pragma package(smart_init)
#pragma resource "*.dfm"
  
```

```

TForm1 *Form1;

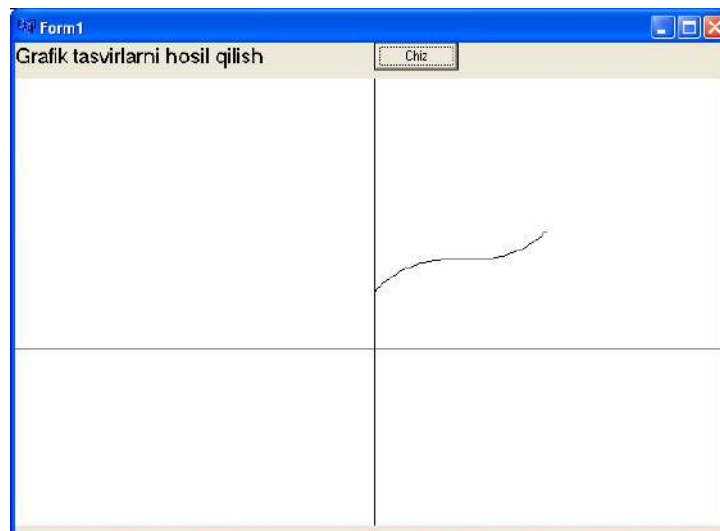
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
    : TForm(Owner)
{
}

//-----

void __fastcall TForm1::Button1Click(TObject *Sender)
{
    float y,x,e;
    int z,w;
    Image1->Canvas->MoveTo(320,0);
    Image1->Canvas->LineTo(320,480);
    Image1->Canvas->MoveTo(0,240);
    Image1->Canvas->LineTo(640,240);
    z=320;
    w=240;
    for(x=0;x<=3.14;x+=3.14/60)
    {
        y=cos(x)+x;
        Image1->Canvas->MoveTo(z,w);
        Image1->Canvas->LineTo(320+x*50,240-y*50);
        z=320+x*50;
        w=240-y*50;
    }
}

//-----

```



O'zbekiston Respublikasi bayrog'i girafigi

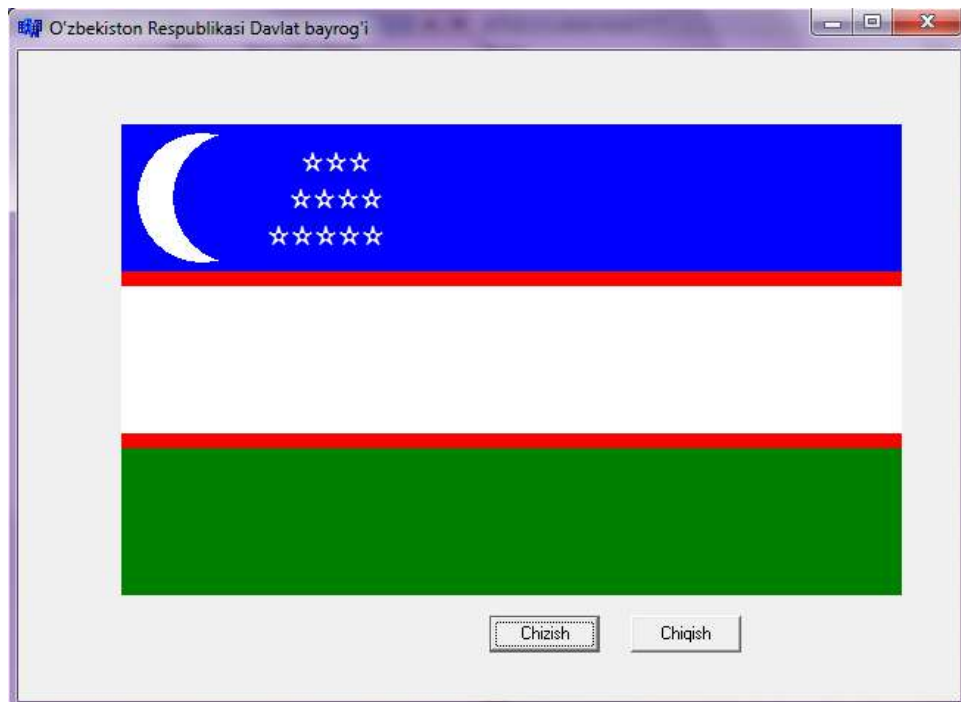
```
//-----
#include <vcl.h>
#pragma hdrstop
#include "Unit1.h"
//-----
#pragma package(smart_init)
#pragma resource "*.dfm"
TForm1 *Form1;
//-----
//-----
__fastcall TForm1::TForm1(TComponent* Owner)
    : TForm(Owner)
{
}
//-----
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    Canvas->Pen->Color = clBlue;
    Canvas->Brush->Color = clBlue;
```

```

Canvas->Rectangle(70,50,600,150);
// oy chizish
Canvas->Pen->Color = clBlue;
Canvas->Brush->Color = clWhite;
Canvas->Ellipse(80,55,170,145);
Canvas->Pen->Color = clBlue;
Canvas->Brush->Color = clBlue;
Canvas->Ellipse(105,55,195,145);
// yulduz chizish
Canvas->Font->Name = "Monotype Corsiva";
Canvas->Font->Color = clWhite;
Canvas->Font->Size = 28;
Canvas->TextOut(180,60," ***");
Canvas->TextOut(180,85,"*****");
Canvas->TextOut(165,110,"*****");
Canvas->Pen->Color = clRed;
Canvas->Brush->Color = clRed;
Canvas->Rectangle(70,150,600,160);
Canvas->Pen->Color = clWhite;
Canvas->Brush->Color = clWhite;
Canvas->Rectangle(70,160,600,260);
Canvas->Pen->Color = clRed;
Canvas->Brush->Color = clRed;
Canvas->Rectangle(70,260,600,270);
Canvas->Pen->Color = clGreen;
Canvas->Brush->Color = clGreen;
Canvas->Rectangle(70,270,600,370);
}

```

```
//-----
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    Close(); //dasturdan chiqish.
}
//-----
```



Nazorat savollari:

1. C++ tilida GDI nima?
2. GDI funksitayasiga murojat.
3. C++ Builder muhitida rasmlar bilan ishlashda nimalarga e'tibor qaratish kerak?
4. Asos nima?
5. C++ tili grafika tushunchasi.
6. C++ tilida rasm tushunchasi.
7. Grafik fayllar.
8. C++ tilida palitrage xizmat ko'rsatish.
9. C++ yilida bit obrazidan nusxa ko'chirish.
10. Grafik yartish va xizmat ko'rsatish.

Ma'ruza №16

Mavzu: Texnik tizimlarda integrallashgan muxitda mavjud bo'lgan ma'lumotlar bazasini yaratish va qayta ishlash.

Reja:

1. Ma'lumotlar bazasi tushunchasi.
2. Borland C++ Builder 6 integrallashgan sohasida ma'lumotlar bazasi.
3. Borland C++ Builder 6 muhitida ma'lumotlar bazasi va uni qayta ishlash.
4. SQL tilini qo'llash. Ttable, TADOTable, TDBGrid, TDBNavigator komponentalari yordamida ma'lumotlar bazasini qayta ishlash.

Ma'lumotlar bazasi.

Odatda, ma'lumotlar bazasi bu kompyuterda ishlatiladigan ma'lumotlarni aniq bir mavzuda to'plash yoki biznes ilovada ishlatish usulidir. Ma'lumotlar bazasi tegishli ma'lumotlarni mantiqiy shaklda izlash va kiritishga yordam beradi. Ayrim ma'lumotlar bazasini boshqarishning eski tizimlari, ayrim jadvallarni bayon qilishda ishlatiladi. Biroq xozirgi vaqtda ma'lumotlar bazasi termini barcha ma'lumotlarsiz tizimlarda xam ishlatiladi. Ma'lumotlar bazasi nafaqat kompyuterlar uchundir. Quydagi ma'lumotlar bazasi xam mavjud bo'lib, ayrim xollarda biz ularni ma'lumotlar bazasini kulda uzatish tizimi yoki ma'lumotlar bazasini kulda boshqarish tizimi deb ataymiz. Bu tizimlar odatda odamlardan, xujjatlardan, papka va shkaflardan iborat bo'lib ma'lumotlar bazasining mexanik tizimining kaliti xisoblanadi. Real ma'lumotlar bazasining mexanik tizimida, siz extimol ayrim formal uzatish usullardan foydalanarsiz. Siz ma'lumotlarga qo'l orqali shkafni ochib, papkani olib erisharsiz. Foydalanuvchilar kirish uchun qogoz shakllarni to'ldirib, balki klaviatura orqali ma'lumotlarni blankalarga kirib erishar. Siz ma'lumotlarni qo'lda izlaganda xujjatlarni nusxa ko'chirish orqali ko'p ishlardan sortirovka qilib, yangi varaqga ko'chirib (yoki Excel elektron jadvaliga) olarsiz. Siz elektron jadvallardan yoki kalkulyatordan foydalanib ma'lumotlarni taxlil qilganingizda yoki yangi qiziqarli usullarda aks ettirganingizda foydalanishingiz mumkin.

Ma'lumotlar bazasiga kirish bu qoqozni tizimdan olishning avtomatlashtirilgan usulining o'zidir. Microsoft Access ma'lumotlar bazasini aniq bir strukturada saqashning usulidir. Access jadvallar ko'p turdagi ma'lumotlarni saqlashning ya'ni oddiy qatordagi matndan (masalan ismi, adresi) murakkab ma'lumotlarni, tovush, ko'rinish, video tavar va boshqalarni saqlash usulidir. Ma'lumotlarni ma'lumotlar bazasida ani? formatda sa?lashning (MBBT), kirish ikoniga, ya'ni ma'lumotlarni foydali ma'lumotlarga aylantirish ega bo'lishdir.

Jadvallar Access ma'lumotlar bazasida ma'lumotlarni saqlovchi asosiy zaxiraxona vazifasini bajaradi. So'rovnomalar, shakllar, xisobotlar ma'lumotlarga kirishning asosiy yo'li bo'lib, ular ma'lumotlarni qulay xolda qo'shish, o'zgartirish, ajratib olish imkonini beradi. Ko'p ishlatuvchilar ilova kodiga (VBA) makroslarni yoki Visual Basicni qo'shish orqali, Access ilovasini o'zlari uchun qulay ko'rinishga keltirishadi.

Uchta asosiy model turlarining farqlari: ierarxik, tarmoq va relyasion.

1. **Ierarxik** tarmoq elementlar majmuidan iborat **bo'lib** unda birinchi pogana elementlari boshqa pogana elementlariga bo'ysunadi. Elementlar orasidagi bogliqlik daraxt ko'rinishida bo'ladi. Bunday bazada xar bir tashkil etilgan element faqat bitta oilaga masub bo'ladi. Bunday bazaga misol qilib ixtiyoriy oila shajarasini keltirish mumkin.

2. **Tarmoq** modeli shu bilan xarakterlanadiki berilganlarning xar bir elementi boshqa ixtiyoriy elementlar bilan boglangan bo'ladi. Bunda xar bir yaratilgan element birdan ortiq element yaratishi mumkin. Bunday bazaga misol qilib turli to'garak mashgulotlarida shugullanuvchi o'quvchini keltirishimiz mumkin. Bu xolatda bir o'quvchi bir necha turli to'garakka qatnashishi shuningdek bir qancha o'quvchilar bir to'garaka qatnashishi mumkin.

3. **Relyatsion** modelda ob'ektlar va ularning o'zaro ikki o'lchovli jadval ko'rinishida tasvirlanadi. Ma'lumotlarning bunday ko'rinishda tasvirlanish ob'ektlari o'zaro aloqalarni yaqqol tasvirlashda asos bo'ladi. Shuni takidlash lozimki hozirgi vaqtda deyarli barcha foydalanuvchilar va komp'yuterlar asosan relyatsion modellar asosida tashkil qilinmoqda. Ma'lumotlar bazasining bunday jadvallar bilan ishlashi relyatsion dyiladi.

Ma'lumotlar bazasining modellari.

Relyatsion ma'lumotlar bazasi.

Ma'lumotlar bazasining yadrosi bo'lib, ma'lumotlar bazasining modeli xisoblanib, u ma'lumotlavr strukturasini ifodalab, usullardagi kelishuvlarni tasavvuriga ega bo'lish, xamda ularni manipulyatsiya qilishdir. Ierarxik, manbaali va relyatsion modellar mavjuddir. Ierarxik model bu bir darajadagii ma'lumotlarni boshqa bir darajadagi ma'lumotlarga bo'ysinishidir, ular daraxt ko'rinishiga egadir. Ma'lumotlarning manbaa (set) modeli bunda ma'lumotlarning xar bir elementi boshqa bir element bilan boglanganligi, xar bir yangi yaratilayotgan element birdan ortiq yaratuvchi elementga ega ekanligini tavsiflaydi. Relyatsion model ma'lumotlarning ixtiyoriy ko'rinishini oddiy ikki o'lchamli jadval ko'rinishida tasavvur etish imkonini beradi. Ma'lumotlarni tasavvur etishning bunday turi, eng foydalanuvchi va kompyuter uchun qulayidir.

MicrosoftAccess ma'lumotlarni ishlab chiqishning relyatsion tizimi xisoblanib, tegishli jadvallarda saqlanadi (masalan klientlar) va u boshqa jadvallar bilan (masalan buyurtmachi) bilan boglangan bo'ladi. Access o'zaro boqlangan jadvallarni qo'llab quvvatlaydi, bu esa ularga klientlarni va ularning buyurtmalarini ma'lumotlarni yo'qotmagan xolda ajratib olish imkonini beradi, yana bir tomoni shundan iboratki, klientlarga tegishli bo'lmagan ma'lumotlarni , buyurtma yozuvlarini xam olisho' imkonini beradi. Biz endi jadvalning "klientlar" yoki "tblCustomers jadvalini" ko'ramiz. Birinchi xolda "klient" jadvali klientning ma'lumotlar bazasiga tegishli bo'lib, u klient xaqidagi ma'lumotlarga egadir, bu vaqtda tblCustomers jadvali ma'lumotlar bazasidagi tblCustomers nomli jadvalga murojaat qiladi. Turli ishlab chiqaruvchilar obektlarni turlicha nomlashi mumkin. Masalan, ma'lumotlar bazasida tblCustomersni klientlar jadvali deb ishlatishi mumkin, biroq boshqa odam, uni klientlar nomi bilan jadvalni nomlab ishlatishi mumkin. ma'lumotlar bazasi bilan ishlaganda, qaysi obekt nomlanishiga, qaysi obekt esa bayoniga ishlatilishini bilish muximdir. Bir qancha jadvallarni ma'lumotlarni ortiqcha va ko'proq kirish maqsadida soddalashtiriladi. Ilova uchun ikkita jadvalni klientlar xaqidagi ma'lumotlarni saqlashda ishlatish uchun aniqlab olib, masalan, klient xar safar detalni sotib olganda uning nomini, manzilini xar safar saqlashning xojati yo'qdir. Oxir oqibat, siz ma'lumotlar bazasini saqlash uchun Exceldan foydalanganingizda, ma'lumotlar bazasini Access jadvalida saqlash uchun foydalanish xam qulay ekanligini bilib olasiz. Bitta katta jadval klient xaqidagi barcha ma'lumotlarni saqlash uchun qiyindir. Siz klientning xar bir sovdosi uchun ma'lumotlarni kiritishingiz, (ismi va manzili xaqida xar safar takroriy ma'lumot kiritish) qiyindir. Bu esa xar bir sotib olingan maxsulot uchun xam zarurdir, ya'ni qachon klient bir savdolashish natijasida qancha maxsulot sotib olganini ko'rsatish uchun zarurdir. Ma'lumotlarni bitta jadvalda saqlash samarasizdir, sababi xar bir savdoda ayrim maydonlarning bo'lishi kerakmasdir va jadvalda ko'p bo'sh maydonlar qolishi mumkin. Biz shunday jadvallarni xosil qilmoqchimizki, ular minimum ma'lumotlni saqlagan xolda, bir vaqtning o'zida foydalanganda qulay va kengaytirish uchun maqul bo'lishi kerak. Bu maqsadga erishish uchun bitta jadvaldan ko'p bo'lgan jadvalni kiritib, xar bir jadvallar ma'lum bir mavzudagi maydonga ega bo'lsin. So'ngra jadval tayyor bo'lgandan so'ng, ulardagi foydali ma'lumotlarni tanlab olish xolatiga keltiriladi. Vaxolanki bu jarayon fakticheski elektron relyatsin ko'rinishda realizatsiya qilingan bo'lib, amalda juda murakkab bo'lib tuyuladi. Bunday Access tizimlarlarda ma'lumotlar jadvallarda saqlanadi. Masalan, xodimlar xaqidagi ma'lumotlarni (ismi, manzili) saqlagan jadval, ularning (oylik maoshi, maoshi miqdori) xaqidagi jadval bilan boglangan bo'ladi. So'rovnomalar foydalanuvchiga murakkab savollarni (Djeynga 2012 yili berilgan maoshlarning summasi qancha) berishi mumkin. Bu boglangan jadvallardan javoblar ekran ko'rinishida yoki pechat qilingan javoblar ko'rinishida beriladi. Qo'l va relyatsion tizimlarning farqi shundan iboratki, relyatsion tizimlar bitta aloxida

olingan shaxs yoki predmetlar uchun ma'lumotlar bazasini aloxida jadvalda saqlashda foydalaniladi. Masalan patsientni boshqarish tizimi, ismi, manzili, va boshqa kontaktlashish ma'lumotlarni boshqa jadvalda saqlash uchun, patsientni davolashni olib borish xaqidagi ma'lumotlarni boshqa jadvalda saqlanishi mumkin. Asl xolda esa davolash ma'lumotlari jadvali barcha patsientlarning davolanishi xaqidagi barcha ma'lumotlarni saqlashi kerak.

Access ma'lumotlar bazasi u bilan boglangan barcha obektlarning ma'lumotlarini saqlovchi umumiy konteyner xisoblanadi. Bu esa jadvallar majmuasidan xam yu?oriro?dir. Access bir vaqtning o'zida barcha ma'lumotlar bazasi qamrab olgan baza xisoblanadi. Ma'lumotlar bazasiga kirishni ochgandan sung (jadvallar, suravnomalar va boshkalar), ob'ektlar ma'lumotlar bazasida sizning ishlaringiz takdimot kilinadi. Siz Accessning bir vaktning uzida kancha nusxalari ochiladi va ma'lumotlar bazasi zarur bulgan xolatda amalga oshiriladi. Kup ma'lumotlar bazasida Access yuzlab va minglab jadvallar jamlangan shakllar, surovnomalar, makrosalar, xisobotlar, modullardan iborat buladi. Ayrim xolatlarda истисно тарикасида Access маълумотлар базаси барча объектлар бита кенгайтирилган файлда .accdb, .accde ёки .adp да булади.

C++da ma'lumotlar bazasi bilan ishlaganda, dastlab **Alias** hosil qilinadi. **Alias** ma'lum bir guruhga tegishli bo'lgan jadvallarni qysi toifaga tegishliligini (Jadvalar hosil qilgan drayverlarni) jadvallar yo'lini o'zida saqlaydi. C++ **Builderda Alias** hosil qilishining bir necha usullari mavjud.

1.Menyuning **DataBase** bo'limidan **SQL Explorer** tanlanadi.

2. **Object->New** [Ctrl+N]tanlanadi.



3.Hosil bo'lgan darchadan drayver nomini tanlab, **standart Paradox** tanlanadi.

4.**Path** bo'limidan jadvallar saqlanadigan katologlar tanlanadi.

5.Hosil bo'lgan **Alias** ni saqlash uchun sichqonchanning o'ng tugmasni bosib, **Apply** tanlanadi.

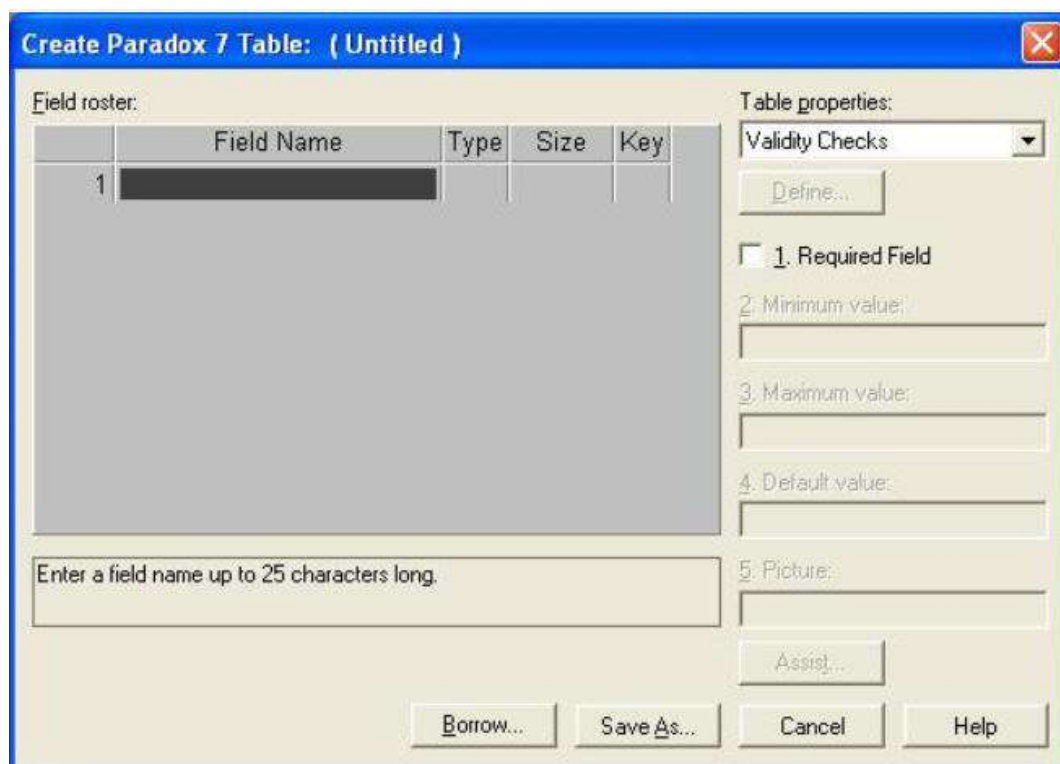
Jadvallar hosil qilish uchun quyidagi amallar ketma ketligi bajarailadi.

1.C++ **builder** menyusining **Tools** bo'limidan **Database Desktop** tanalanadi.

2. **Database desktop**dan **File->New->Table** tanalanadi.



3. Jadval toifasi tanlanadi.



4. **Field Name**-Jadval maydoni nomi

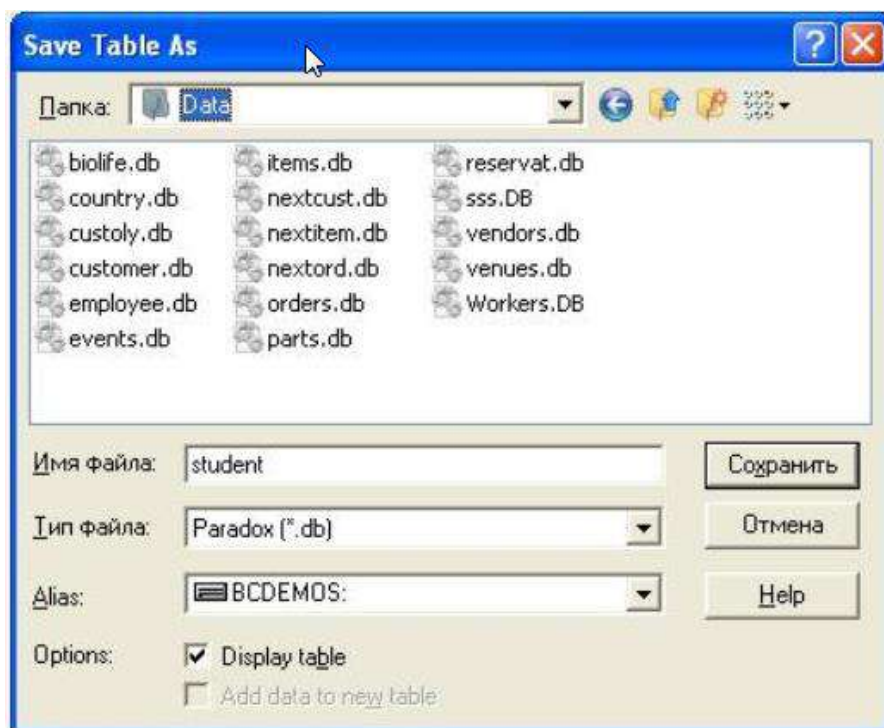
Type-maydon toifasi;

Size-hajmi;

Key-kalit maydonini bildirish bo'limi;

1. Jadval maydonlari kiritilgandan so'ng **Save as..** tugmasi bosiladi.
2. **Alias** bo'limidan o'zimiz hosil qilgan alias nomini tanlaymiz.
3. Jadvalga nom beriladi.
4. Options bo'limidan **Display Table** ga bayroqcha o'rnatilsa, **Soxranit** tugmasi bosilgandan keyin, **Database desktop** da jadval ochiladi.

5. Bayroqcha o'rnatilmasa(yoki ixtiyoriy jadvalni ochish uchun) **File->Open->Table** tanlanadi. **Alias** bo'limidan jadval saqlangan alias tanlanadi. So'ngra ochilishi lozim bo'lgan jadval nomi tanlanib **Otkrit** tugmasi bosiladi.



Maydonlar haqida ma'lumot

Maydon nomi 25 ta simvoldan iborat bo'lishi mumkin. Birinchi simvol probel bo'lishi mumkin emas.

Eslatma: Maydon nomini yozishda probeldan umuman foydalanmagan ma'qul. Chunki, SQL so'rovlardan foydalanganda muammo chiqishi mumkin. Zarurat bo'lsa, shift minus "_" belgisidan foydalaning. Apostrovdan (') ham foydalanmang!

Maydon toifasini tanlash uchun Type maydoniga o'tib, sichqonchani o'ng tarafi bosiladi yoki probel bosiladi. Paradox uchun maydon toifalari quyidagicha bo'lishi mumkin.

1. A 1-255 Alpha Satrli maydon. ASCII kodini barcha simvollarida qabul qiladi.
2. N-number – 10307 -10308 butun son
3. \$ - money – pul birligini bildiruvchi musbat yoki manfiy son.
4. S – short -32767 ... 32767 oralig'idagi butun sonlar.
5. I – long integer -2147483648 ... 2147483648
6. # - 0 – 32 BCD Binary Coded Decimal formatdagi son.
7. D – Date - sanani bildiruvchi maydon.
8. T- Time - vaqt.
9. @ - Time Stamp - vaqt va sanani bildiruvchi maydon.
10. M – 1-240 Memo cheklanmagan satr ma'lumotlarini saqlash uchun mo'lallangan.
11. F – 0-240 Formatted cheklanmagan format. Memo ulangan satrni saqlash uchun mo'ljallangan.
12. G – graphic tasvir ma'lumotlarini saqlash.
13. O – OLE tasvir, ovoz, dokument va xakozolarni saqlash.
14. L – logical mantiqiy maydon.
15. + Autoincrement avtomat ravishda yozuvni bittaga oshirib boradi. Yozuvlar o'chirilsa oldindagilar o'zgarishsiz qoladi.

Misol: Lesson nomli alias hosil qiling. Quyidagi maydonlarni o`z ichiga oluchi talaba nomli jadval hosil qiling.

Nº	Field Name	Type	Size
1	ID	+	
2	Familiya	A	20
3	Ism	A	20
4	Otasi	A	20
5	Guruh	A	5
6	Tug_kun	D	

Tug_kun maydonida tug'ilgan kuni sanasi yoziladi.

Bu jadvalni **C++ Builder** bilan bog'lashni ko'rib chiqamiz.

1. **File** → **New** → **Application** tanlanadi;
2. Komponentalarning **BDE** bo'limidan **Table** komponentasi formaga qo'yiladi;
3. **Table** komponentasining **DatabaseName** hususiyatida **Alias** ko'rsatiladi. (yani **Lesson**);
4. **TableName** hususiyatida jadval nomi keltiriladi. (talaba);
5. **Active** hususiyati **true** ga o'zgartiriladi;
6. Komponentalarning **Data Access** bo'limidan **DataSource** komponentasi formaga qo'yiladi;
7. **DataSet** hususiyati **Table1** qilinadi;
8. Komponentalarning **Data Controls** bo'limidan **DBNavigator** komponentasi formaga qo'yiladi;
9. **DataSource** hususiyati **DataSource** qilinadi;
10. Komponentalarning **Data Controls** bo'limidan **DBGrid** komponentasi formaga qo'yiladi;
11. **DataSource** hususiyati **DataSource** qilinadi;
12. Formaga **Edit** va **Button** komponentalari qo'yiladi. **Button** komponentasi ustida sichqoncha ikki marta bosiladi va quyidagilar kiritiladi:

```
TLocateOptions qidiruv_turi;  
qidiruv_turi << loPartialKey << loCaseInsensitive;  
  
if (!table1->Locate("Familiya",Edit1->Text,qidiruv_turi))  
ShowMessage("Bunday yozuv yo`q");
```

Bu yerda **qidiruv_turi** turidagi **loPartialKey** qidirilayotgan familiya qisman kiritilsa ham qidiruvni amalga ochirishni bildiradi. Masalan: Abdurahimov familiyasini qidirish lozim bo'lsa, Abdu deb yozilsa ham qidiruv amalga oshirilishini bildiradi.

loCaseInsensitive esa, qidirilayotgan ma'lumotning katta yoki kichik yozilganiga ahamiyat bermaslikni bildiradi. Masalan: Abdurahimov familiyasini qidirish lozim bo'lsa, abdurahimov deb yozilsa ham qidiruv amalga oshirilishini bildiradi.

Bu parametrlarni alohida ishlatilsa ham bo'ladi.

```
qidiruv_turi << loCaseInsensitive;
```

yoki

```
qidiruv_turi << loPartialKey;
```

Maydonlar bo'yicha filterlash

1. Formaga **Button** va **Edit** komponentalarini qo'ying

2. **Button** komponentasiga quyidagilar yoziladi

```
Table1->Filtered = false;  
Table1->FilterOptions << foCaseInsensitive;  
Table1->Filter = "Familiya='" + Edit1->Text + "'";
```

3. **Table** ning **FilterOptions** xususiyatining **foCaseInsensitive** xususiyati **True** qilinadi.

Yoki **Edit** komponentasining **Onchange** hodisasiga quyidagilar yoziladi:

```
Table1->Filtered = false;
```

```
// Edit1 bo'sh bo'lsa funksiyadan chiqib ketiladi  
if (Edit1->Text == "")  
return ;
```

```
Table1->FilterOptions << foCaseInsensitive;  
Table1->Filter = "Familiya='" + Edit1->Text + "'";  
Table1->Filtered = true;
```

DIQQAT: Filterlashni bu turi faqat Paradox uchun o'rinli

Maydonlarga quyidagicha murojaat qilish mumkin.

```
Table1->FieldByName ("Maydon_nomi")->Value;
```

```
Table1->FieldByName ("Narxi")->AsCurrency;  
Table1->FieldByName ("Soni")->AsInteger;
```

Maydonga biror qiymatni o'zlashtirish quyidagicha amalga oshiriladi:

```
Table1->Edit; // Jadvalni o'zgartirishga ruxsat berish
Table1->FieldByName("Ism")->AsString = Edit1->Text;
Table1->Post(); // Saqlash
```

Biror maydon yig'indisini hisoblash quyidagicha bo'lishligi mumkin.

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    float s;
    Table1->First;
    while(!Table1->Eof)
    {
        s += Table1->FieldByName("Jami")->Value;
        Table1->Next; // keyingi yozuvga o'tish
    }
    Button1->Caption = FloatToStr(s);
}
```

-- . --

SQL tili (Structured Query Language – strukturalashgan so'rovlar tili) sintaksisi juda oson. SQL tilida katta – kichik harflar farqlanilmaydi. Ya'ni Select operatorini SELECT, Select, select shakllarida yozish mumkin. Agar bir nechta operatoridan foydalanilsa operator oxirida “,” qo'yiladi. Faqat bitta operator ishlatilsa operator oxirida “,” qo'yish shart emas. Izoh yozish ba'zi sistemalarda / * <IZOH> * shaklida, ba'zi sistemalarda esa { <IZOH> } shaklida bo'ladi.

SQL so'rovlar tilidan deyarli barcha (MS SQL Server, MySQL, PostgreSQL, Oracle, Informix, Paradox Interbase, FireBird) ma'lumotlar bazasida foydalanish mumkin. SQL so'rovlari ma'lumotlar bazasiga qarab qizman o'zgarishi mumkin.

Select tanlash operatori

Select operatorining umumiy ko'rinishi

```
SELECT <maydon nomlari ro'yhati>
FORM <jadvallar>
WHERE <izlash sharti>
GROUP BY <guruhlanishi lozim bo'lgan maydonlar>
HAVING <izlash sharti>
ORDER BY <maydon nomlari ro'yhati>;
```

Quyidagi maydonlarni o'zida saqlovchi namuna nomli jadval hosil qiling va uni Lesson aliasiga saqlang.

No	Field Name	Type	Size
1	ID	+	
2	Familiya	A	20
3	Ism	A	20
4	Otasi	A	20
5	Guruh	A	5
6	Tug_kun	D	
7	Tug_yil	S	
8	Jinsi	A	5

1. Bu jadvaldagi ma'lumotlarni ko'rish quyidagicha bo'ladi:

```
SELECT *  
FROM namuna
```

2. Jadvaldagi ba'zi maydonlarni ko'rish uchun shu maydon nomlari ro'yhati select dan keyin keltiriladi:

```
SELECT Familiya, Ism, Otasi  
FROM namuna
```

3. Familiya bo'yicha tartiblash

```
Select familiya, ism, otasi  
From namuna  
Order By Familiya, Ism
```

4. Tug'ilgan yili bo'yicha kamayib borish tartibida tartiblash

```
Select familiya, ism, otasi, tug_yil  
FROM namuna  
Order By tug_yil DESC
```

5. Select operatoridan keyin faqat maydon nomi emas, ixtiyoriy arifmetik amal (+, -, *, /) ishlatish mumkin. Misol uchun, yuqoridagi jadvaldagi tug'ilgan yilni joriy yildan ayirib, talabning yoshini aniqlaymiz:

```
Select familiya, ism, otasi, (2013-tug_yil) AS Yoshi  
FROM namuna
```

6. Agar maydon nomini ruscha shriftda chiqarish talab etilsa quyidagicha so'rov yoziladi: (Maydon nomidan keyin **AS** xizmatchi so'zi bilan maydon sarlavhasi keltiriladi. Faqat "E" xarfida muammo bo'lishi mumkin)

```
Select familiya AS Фамилия, ism AS Имя  
FROM namuna
```

6.1. Familiya va Ism maydonlarini bitta ustunda chiqarish uchun quyidagicha yozish mumkin.

6.1. Familiya va ism maydonlarini bitta ustunda chiqarish uchun quyidagicha yozish mumkin.

```
Select familiya + ' ' + ism AS "Фамилия Имя"  
FROM namuna
```

MySQL da yuqoridagi so'rov sal boshqacha amalga oshiriladi.

Where operatoridan keyin tanlash shartida mantiqiy amallar **AND**, **OR**, **NOT**, arifmetik amallar va munosabat amallari:

=	Teng
>	Katta
>=	Katta yoki teng
<	Kichik
<=	Kichik yoki teng
!= yoki <>	Teng emas
LIKE	belgilar ketma - ketligi
between ... and	Qiyamatlar oralig`i
IN	To`plamga tegishlilikni aniqlash

7. Dastlabki 6 ta amal tushunarli. Misol uchun jinsi ayol bo'lgan, 1990 yildan keyin tug'ilganlar ro'yhatini chiqaramiz.

```
Select Familiya, Ism, Otasi, tug_yil, (2013-tug_yil) AS yoshi  
Select Familiya, Ism, Otasi, tug_yil, (2013-tug_yil) AS yoshi  
FROM namuna  
Where jinsi="Ayol" and tug_yil>1990
```

8. Like amali sintaksisi quyidagicha:

<maydon> **LIKE** "<belgilar ketma - ketligi>"

Bu amal satrli maydonlar uchun qo'llaniladi va izlanayotgan ma'lumot topilsa true qiymat qaytaradi. Belgilar ketma – ketligini to'liq kiritish yoki "%" belgisi bilan tugatish mumkin. Bu belgi ixtiyoriy belgilar ketma – ketligini bildiradi.

Misol:

```
Select familiya, ism, otasi  
FROM namuna  
Where Familiya LIKE "A%"
```

Amali "A" harfidan boshlanuvchi barcha familiyalarni bildiradi.
(Adambaev, Abdurahimov, Alimov, Azamov, Asqarov, Azizov, ...)

```
Select familiya, ism, otasi  
FROM namuna  
Where Familiya LIKE "Abdu%"
```

Amali "Abdu" harflaridan boshlanuvchi barcha familiyalarni bildiradi.
(Abdurahimov, Abdugʻaʼniyev, Abduqayyumov, Abdullaev)

```
Select familiya, ism, otasi  
FROM namuna  
Where Ism LIKE "%im%"
```

Amali ketma – ketlikda "im" qismi bor boʻlgan ismlarni bildiradi.
(Karimboy, Abdukarim, Rahimjon, Hakimboy, Salimjon, ...)

9. **between ... and** amali sintaksisi quyidagicha:

```
<maydon> between <qiymat> and <qiymat>
```

1985 va 1990 yillar oraligʻida tugʻilganlar roʻyhatini chiqarish

```
Select familiya, ism, otasi  
FROM namuna  
WHERE tug_yil BETWEEN 1985 AND 1990
```

10. **IN** amali sintaksisi quyidagicha:

```
<maydon> IN (< toʻplam >)
```

Maydondagi maʼlumot qiymati, toʻplam elementlariga tegishli boʻlganlarini chiqarib beradi. Masalan,

```
SELECT *  
FROM namuna  
WHERE Ism IN ('Azamat', 'Qudrat', 'Ne'mat')
```

Tugʻilgan yili 1980, 1982, 1989 boʻlganlar roʻyxati esa quyidagicha:

```
SELECT familiya, ism, otasi, tug_yil  
FROM namuna  
WHERE tug_yil IN (1980, 1982, 1989)
```

11. **DISTINCT** xizmatchi soʻzi koʻrsatilgan maydon boʻyicha bir xil qiymatli maʼlumotlardan faqat bittasini olishni bildiradi

```
SELECT DISTINCT familiya  
FROM namuna
```

9. **Count** koʻrsatilgan maydon boʻyicha, yoki butun jadvaldagi yozuvlar sonini aniqlash uchun ishlatiladi

```
Select count (*)  
FROM namuna
```

95-99 guruhidagi talabalar sonini aniqlash:

```
Select count (*) as soni  
FROM namuna  
WHERE quruh="95-99"
```

Yuqorida keltirilgan so`rovlarni C++Builder da qo`llanilishini ko`rib chiqamiz.

1. **File** → **New** → **Application** tanlanadi;
2. Komponentalarning **BDE** bo`limidan **Query1** komponentasi formaga qo`yiladi;
3. **Query1** komponentasining **DatabaseName** hususiyatida **Alias** ko`rsatiladi. (yani Lesson);
4. **Query1** komponentasining **SQL** hususiyatiga quyidagilar kiritiladi:

`select * from namuna`
5. **Query1** komponentasining **Active** hususiyati **true** ga o`zgartiriladi;
6. Komponentalarning **Data Access** bo`limidan **DataSource** komponentasi formaga qo`yiladi;
7. **DataSet** hususiyati **Query1** qilinadi;
8. Komponentalarning **Data Controls** bo`limidan **DBNavigator** komponentasi formaga qo`yiladi;
9. **DataSource** hususiyati **DataSource1** qilinadi;
10. Komponentalarning **Data Controls** bo`limidan **DBGrid** komponentasi formaga qo`yiladi;
11. **DataSource** hususiyati **DataSource1** qilinadi;
12. Formaga **Label**, **Edit** va **Button** komponentalari qo`yiladi. Button komponentasiga quyidagilar kiritiladi:

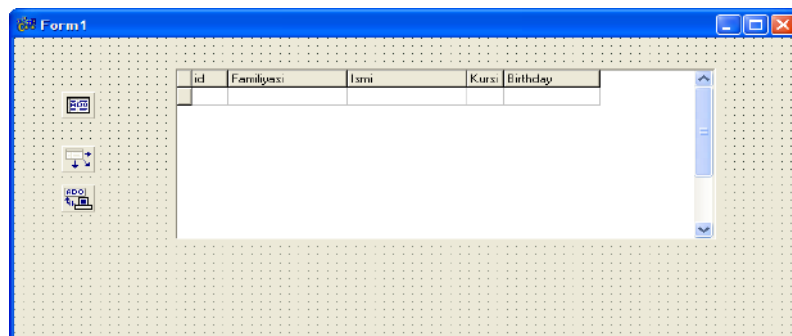
```
Query1->Close;  
Query1->SQL->Clear();  
Query1->SQL->Add("Select * from namuna");  
Query1->SQL->Add("Where Familiya=\""+Edit1.Text+"\"");  
Query1->Open();
```

C++ builderda Access dasturi yordamida ma'lumotlar bazasini yaratish.

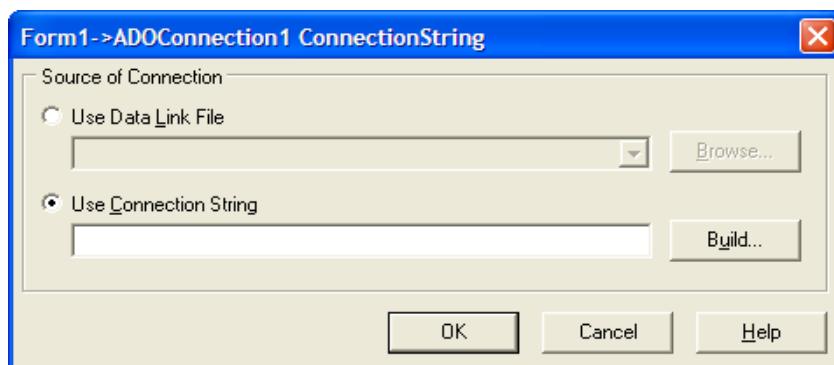
Ma'lumotlar bazasi bilan ishlovchi dasturlar judayam ko'p. Ularni ishlatilish sohasi bo'yicha bir nechta guruhlariga bo'lish mumkin. Bu yerda biz uchun qulay bo'lgan Access 2000 da Talaba.mdb fayl yaratamiz va undagi Talaba jadvalini Builder muhitida ulashni ko'rib chiqamiz.

Avvalambor, Talaba.mdb faylini yaratib unda jadval yaratamiz va bu ma'lumotlar bazasini Builderdagi loyihamiz yaratilgan papkaga saqlaymiz.

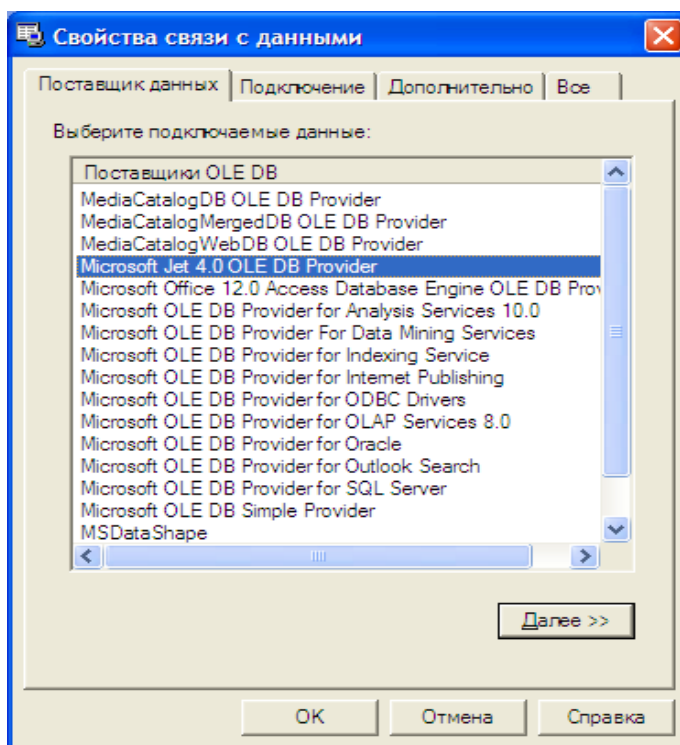
1. Builderda Access bilan ulanish uchun Standart panelidagi ADO panelidan ADOConnection1 va ADOTable1 komponentasini, DataAccess panalidan DataSource1 ni, DataControl panelidan DBGrid1 komponantasini tanlaymiz va ularni formaga tashlaymiz.



2. ADOConnection1 ni ustiga sichqonchani ikki marta bosish orqali uningConnectionString xususiyati ochiladi.

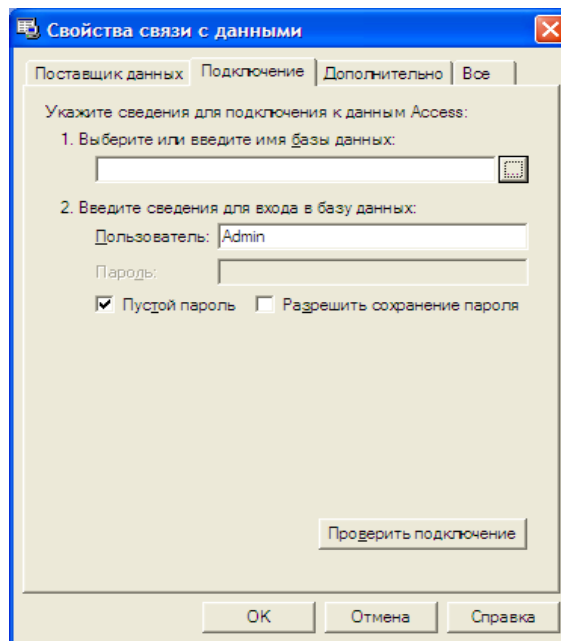


3. Undan Build tugmasini bosamiz.



4. Microsoft.Jet.OLEDB.4.0 Provider ni tanlaymiz.

5. So'ngra (...) tugmasi orqali Accesda yaratilgan fayl ko'rsatiladi hamda uning ulanganligi "Проверить подключение" tugmasi bilan tekshiriladi. Agarda "Проверка подключение выполнена" natijasi chiqsa ulangan sanaladi.



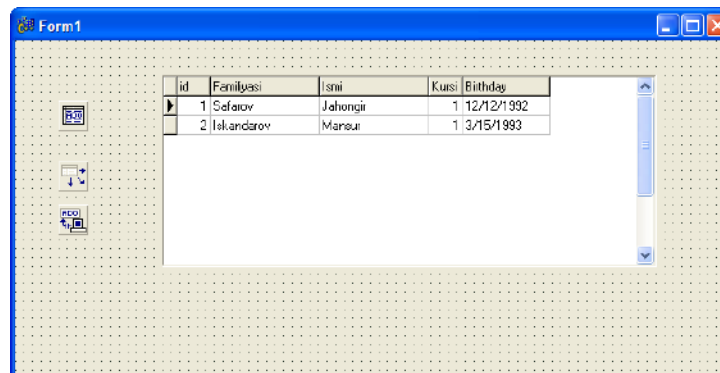
6. Endi ma'lumotlar bazasidagi Talaba jadvalini ulash lozim. Buning uchun ADOTable1 ning Connection xususiyatiga ADOConnection1 tanlanadi.

7. So'ngra TableName xususiyatidan kerakli jadval (talaba) tanlanadi. Agarda UserName va Password so'ralsa OK deyish lozim.

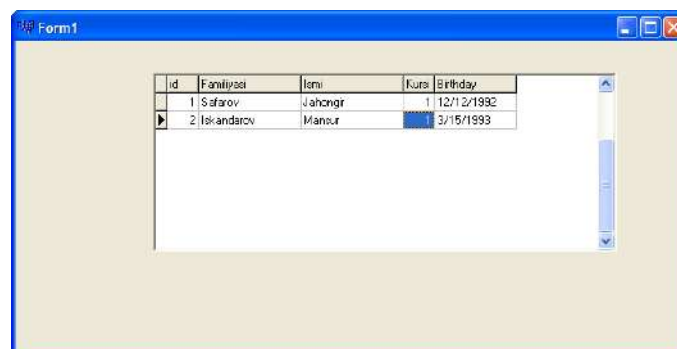
8. Albatta ADOTable1ni Active xususiyatini True qilish lozim.

9. Endi DataSource1 ni DataSet xususiyatidan ADOTable1ni tanlaymiz.

10. Eng oxirida DBGrid1 ni DataSource xususiyatiga DataSource1 ni tanlaymiz hamda barcha bajarilgan amallarni saqlaymiz.



11.F9 tugmasi yordamida loyihani ishga tushiramiz.



16.1-jadvalda berilgan masalalarni Accessda ma'lumotlar bazasini tuzib Builder bilan bog'lang. Jadvalga kamida 20 ta satr kiriting hamda har bir ustun bo'yicha saralash amallarini Builderda bajaring (Masalan, har bir ustun uchun alohida tugmacha qo'yib saralashni amalga oshirish mumkin).

Nazorat savollari

1. Ma'lumotlar bazasi deb nimaga aytiladi ?
2. MBBT qanday tushuncha?
3. Accessda jadvallar qanday yaratiladi?
4. MBni loyixalash nechta turi mavjud?
5. C++da qanday MB yaratiladi?
6. SQLning ma'nosi nima?
7. 2.SQLda jadvallar qanday yaratiladi?
8. 3.SQLda so'rovlar qanday amalga oshiriladi?
9. 4.Borland C++da ma'lumotlar bazasi qanday yaratiladi?

FOYDALANILADIGAN ADABIYOTLAR RO'YXATI:

1. Asosiy adabiyotlar

1. Kadirov M.M. Axborot texnologiyalari. O'quv qo'llanma, 1-qism. -T.:Sano-standart, 2018. - 320 b.
2. Kadirov M.M. Axborot texnologiyalari va jarayonlarni matematik modellashtirish. Darslik, 2-qism. -T.:O'zbekiston faylasuflari milliy jamiyati, 2019. -306 b.
3. 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, - 552 b.
4. Kenjabayev A.T., Ikromov M.M., Allanazarov A.Sh. Axborot-kommunikatsiya texnologiyalariyu. O'quv qo'llanma. – T.: O'zbekiston faylasuflari milliy jamiyati, 2017. - 408 b.

2. Qo'shimcha adabiyotlar

1. O'zbekiston Respublikasi Prezidentining 2017 yil 7 fevraldagi PF-4947-son «O'zbekiston Respublikasini yanada rivojlantirish bo'yicha Harakatlar strategiyasi to'g'risida»gi farmoni.
2. Nazirov Sh.A., Qobulov R.V., Bobojonov M.R., Raxmanov Q.S. C va C++ tili. T.: Voris-nashriyot, 2013. - 488 b.
3. Kenneth C. Laudon, Jane. P. Laudon. Management Information Systems: Managing the Digital Firm, 13th Edition, Pearson Education, USA 2014. P 621.
4. Kunwoo Lee. Principles of CAD/CAM/CAE: The Computer Aided Engineering Design Series. 5st Edition. Addison Wesley Longman, USA, 2015.
5. Alex Allain. Jumping into C++. USA, 2014. p 340.
6. Azimdjanova M.T., Muradova M.T., Pazilov M.S. Informatika va axborot texnologiyalari. O'quv qo'llanma. –T.: O'zbekiston faylasuflari milliy jamiyati, 2013. -176 b.
7. Aripov M., Dattoyev S., Fayziyeva M. Web texnologiyalari. O'quv qo'llanma. –T.: O'zbekiston faylasuflari milliy jamiyati, 2013. -280 b.
8. Ganiyev S.K., Karimov M.M., Tashev K.A. Axborot xavfsizligi. Darslik. – T.:Fan va texnologiya, 2017. - 372 b.

3. Axborot manbaalari

9. www.lex.uz – O'zbekiston Respublikasi Qonun hujjatlari ma'lumotlari milliy bazasi.
10. www.ziynet.uz – O'zbekiston Respublikasi ta'lim portali.

Mundarija:

	Kirish	
1	Axborot texnologiyalari va jarayonlarni matematik modellashtirish faniga kirish	
2	Intellectual boshqaruv tizimlari maqsadi, funktsiyalari va uslubiyati. Ekspert tizimlari.	15
3	Zamonaviy avtomatlashtirilgan loyihalash tizimlari	19
4	Matematik modellashtirish, sonli tahlil usullarini tizimlarda amalga oshirish. Hisobli eksperiment	24
5	Grafik modellashtirish. Loyihalash jarayonida amaliy dasturlarni grafik imkoniyatlaridan foydalanish.	35
6	Geoinformatsion tizimlar	56
7	Tarmoq ma'lumot bazasi. Bulutli texnologiyalar.	62
8	Texnik va texnologik jarayonlarida axborot xavfsizligini ta'minlash	72
9	Texnik tizimlardagi axborotlarni kriptografik himoyalash	76
10	Zamonaviy dasturlash texnologiyalari. Ob'yektga yo'naltirilgan dasturlash tillari.	81
11	C++ dasturlash tilining asosiy konstruksiyalari va tizimda qo'llash xususiyatlari	96
12	C++ dasturlash tilida oddiy dastur tuzilishi.	100
13	Mantiqiy dasturlash texnologiyasi.	108
14	Dasturlashda funksiya va modullarni qo'llash.	126
15	Grafika va multimedia dasturlash tizimlarida qo'llash.	145
16	Texnik tizimlarda integrallashgan muxitda mavjud bo'lgan ma'lumotlar bazasini yaratish va qayta ishlash.	161

