

**O'ZBEKISTON ALOQA VA AXBOROTLASHTIRISH AGENTLIGI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI**

Himoyaga ruxsat etilgan
ATDT kafedra mudiri
professor Nishanov A.X _____
2011 yil « » _____

**DASTURLASH TEXNOLOGIYALARI FANI QISMIDA MULTIMEDIYALI
ELEKTRON AMALIYOT MASHG'ULOTLARINI YARATISH
(ZIYONET.UZ VA ATDT.UZ SAYTLARIDA HAMDA
TATU PORTALIGA JOYLASHTIRISH) MAVZUSIDA**

Bitiruv malakaviy ishi

Bitiruvchi _____ Movlonxo'jayeva S.M.
(imzo) F.i.o.

Rahbar _____ Nishanov A.X
(imzo) F.i.o.

Taqrizchi _____ Ergashev Sh.T.
(imzo) F.i.o.

TX va MM _____ Ishmuhamedova V.I
bo'yicha maslahatchi. (imzo) F.i.o.

MUNDARIJA

KIRISH

I BOB. MULTIMEDIYALI ELEKTRON RESURSLAR QIYOSIY TAHLILI

- 1.1. Multimedial ta'lim beradigan o'quv mahsulotlarining qiyosiy tahlili.
- 1.2. Multimedial o'quv qo'llanmalar yaratishning metodologik asoslari va bosqichlari
- 1.3. Instrumentariya. Interaktivlik
- 1.4. Bitiruv malakaviy ishining maqsad va vazifalari

I bob bo'yicha xulosa.

II BOB. ZAMONAVIY TEXNOLOGIYALARGA IZOH

- 2.1. "Visual Studio" dasturiy taminoti haqida
- 2.2. "CAPTIVATE" dasturiy taminoti haqida
- 2.3. "COURSE LAB" dasturiy ta'minoti haqida

II bob bo'yicha xulosa.

III- BOB. AMALIYOT ISHLARNING DASTURIY TA'MINOTI

- 3.1. Maqsadlarni aniqlash. "Masala qo'yilishi" hujjati.
- 3.2. Yashash sikl modellari.
- 3.3. Foydalanuvchi bilan muloqot.
- 3.4. "Texnik topshiriq" hujjati.
- 3.5. DT arxitekturasi. "Tashqi spetsifikatsiya" hujjati.
- 3.6. Kichik dasturlar standarti. "Ichki spetsifikatsiya" hujjati.
- 3.7. DTni testlash. Testlash prinsiplari va ularning usullari.
- 3.8. DTni ishlatish. Ishlatishga ko'rsatmalar.

III bob bo'yicha xulosa.

IV BOB. XAYOT FAOLIYATI XAVFSIZLIGI

4.1. Ishlab chiqarish muhitida mikroiklimni inson organizmiga ta'siri.

4.2. Yong'in xavfsizligi.

IV bob bo'yicha xulosa.

XULOSA

FOYDALANILGAN ADABIYOTLAR

ILOVA.

Annotatsiya

Ushbu bitiruv malakaviy ish Visual Studio, Captivate, Sound Forge, Course lab texnologiyalari asosida dasturlash texnologiyalari fani qismidagi multimediyali elektron amaliyot mashg'ulotlarini yaratish hamda masofaviy ta'limni tashkil etishga bag'ishlangan. Maqsadga erishish uchun Visual Studio, Captivate, Sound Forge, Courselab texnologiyalari va ko'rsatmalari masalasi ko'rib chiqildi va multimediyali elektron amaliyot mashg'ulotlari yaratildi.

Аннотация

Настоящая дипломная работа посвящена созданию в части предмета технологии программирования мультимедийных электронных практических занятий и организации дистанционного обучения на основании технологий Visual Studio, Captivate, Sound Forge, Course lab. Для достижения цели были рассмотрены технологии Visual Studio, Captivate, Sound Forge, Courselab и созданы мультимедийные электронные практические занятия.

The summary

The present thesis is devoted to creation regarding the subject of technology of programming of multimedia electronic practical training and the organization of remote training on the basis of technologies Visual Studio, Captivate, Sound Forge, Course lab. For purpose achievement technologies Visual Studio, Captivate, Sound Forge have been considered, multimedia electronic a practical training Courselab and is created .

KIRISH

“Dasturlash texnologiyalari” fani 5521900-Informatika va axborot texnologiyalari, 5523500-Axborot xavfsizligi, 5523600-Elektron tijorat,5811200-Servis(Axborot servisi), 5811300-Servis (Elektron va kompyuter texnikasi),5320200-Axborotlashtirish va kutubxonashunoslik, 5140900-Kasb-ta’limi (Informatika va AT), yo’nalishdagi bakalavriat 2 kurs talabalari uchun mo’ljallangan.

Ushbu uslubiy qo’llanma “Dasturlash texnologiyalari” fani bo’yicha amaliyot ishlarini nazariy va amaliy bajarilishining asosiy holatini o’rganishda talabalarga yordamchi bo’lib hisoblanadi.

Mavzular 5521900 informatika va axborot texnologiyalari ,5523500-Axborot xavfsizligi, 5523600-Elektron tijorat,5811200-Servis(Axborot servisi), 5811300-Servis (Elektron va kompyuter texnikasi),5320200-Axborotlashtirish va kutubxonashunoslik, 5140900-Kasb-ta’limi (Informatika va AT) yo’nalishdagi talabalar uchun o’qitiladigan kurs dasturi bilan mos ravishda berilgan.

Har bir amaliyot ishining mazmuni: ishning maqsadini, talabaning mustaqil ishiga topshiriqni, amaliyot ishini bajarish tartibini, ish bo’yicha hisobotnomani, talaba bilim darajasini aniqlash uchun test hamda nazorat savollarini, ishlatiladigan adabiyotlar ro’yxatini, nazariy materialni o’rganish bo’yicha uslubiy qo’llanmani o’z ichiga oladi.

Auditoriyadagi amaliyot ishidan oldin talaba keyingi darsga tayyorlab kelishi kerak bo’lgan mustaqil ish uchun topshiriqni oladi. Mustaqil ish bo’yicha hisobotnoma nazorat savollariga qisqa javoblarni hamda topshiriqni dasturi va yechimini o’z ichiga olishi kerak.

Auditoriyada o'qituvchi tomonidan talabning bilim darajasi tekshirilgandan keyin o'z varianti bo'yicha talaba topshiriqni bajaradi. Dars yakunida o'qituvchi talaba bajargan ishini tahlil qiladi va baholaydi.

Auditoriyadagi darsda material qanchalik chuqur o'rganilishiga e'tibor beriladi. Mavzuni mustaqil o'rganishda talabaga tushunarsiz bo'lgan savollar aniqlanadi va tushuntiriladi.

I BOB. MULTIMEDIALE ELEKTRON RESURLARNING QIYOSIY TAHLILI

1.1. Multimediale ta'lim beruvchi o'quv mahsulotlarining qiyosiy tahlili

Bu bo'limda multimedia resurslarini yaratishda foydalanilgan dasturiy ta'minotlarga va uning mazmuniga nisbatan qiyosiy tahlili keltirilgan. Elektron resurslarni yaratish yo'llari, vositalari va qurollari haqida fikr yuritilgan.



Multimedia resurslarini yaratishda foydalanilgan dasturiy ta'minotlar haqida

Ushbu qism bo'limda asosan ikkita dasturiy ta'minot haqida fikr va mulohazalar keltirildi.

Elektron resurslarni yaratish yo'lida, resurs qurishning o'rganilayotgan texnologiyasini ajralmas qismi hisoblangan umumiy tamoyillarga rioya qilish shart. Bunday tamoyillar pedagoglarni elektron resurs yaratish va foydalanishga tayorlash metodik tizimi tarkibiga kirishi lozim.

Kompyuter texnologiyalariga asoslangan ta'lim ahamiyatli darajada texnik infrastrukturaga tayanadi : kompyuterlar (o'quv axborotlarini joylashtirish va

taqdim qilish uchun qurol sifatida) va komyuter tarmoqlari (unga kirishga ruxsat vositasi sifatida).

Shuning uchun elektron resurslar yaratishda hisobga olinishi zarur bo'lgan tamoyillardan biri o'quv materialini taqsimlash tamoyil hisoblanadi.

Elektron resurs ishlab chiqishda hisobga olinishi kerak bo'lgan ikkinchi muhim tamoyil o'quv materialini interaktivligidir.

Interaktiv vositalar axborotni taqdim qilishning turli xil vositalarini - tekst, statik va dinamik grafiklar, video va audio zapislarni bir butun majmuaga birlashtirish imkoniyatini yaratadi, bu esa ta'lim oluvchini o'quv jarayonida faol ishtirokchi bo'lishiga yo'l beradi, modomiki, axborotlarni taqdim qilish ta'lim oluvchining harakatiga muvofiq javob sifatida yuz beradi.

Mulytmediadan foydalanish axborotni o'zlashtirishning o'ziga xos hususiyatlarini maksimal darajada hisobga olishga imkon beradi, bu pedagog tomonidan ta'lim oluvchiga kompyuter vositasida o'quv axborotlarini etkazib berishda juda ham muhimdir.

Shu tarzda, elektron resurslarni yaratishda hisobga olinishi lozim bo'lgan uchinchi tamoyil bu o'quv axborotlarini mulytmediali taqdim qilishdir.

Ta'limning istalgan yangi shakli psixologik-pedagogik asoslarni shakllantirishni talab qiladi, bularsiz o'quv jarayonining yutuqlari va samaradorligi xaqida gapirib bo'lmaydi.

Elektron resurslarni yaratish texnologiyalarini amalga oshirishda insonning psixo-fiziologik xususiyatlarini e'tiborga olish muammolari asosiy o'rinni egallaydi.

Ta'lim jarayonini optimallashtirish yo'lida asosiy muammo insonni yangi bilimlarni egallash jarayonidagi holatini baholash va yaxshilashdir. Bunda elektron resurslarni yaratishda hisobga olinishi lozim bo'lgan to'rtinchi tamoyil - ta'lim oluvchining shaxsiy hususiyatlariga moslashtirish tamoyilidir. Ta'limda

mustaqil ishlarni (elektron resurslarni qo'llagan holda) ahamiyati katta bo'lishiga qaramasdan, o'quv jarayonining asosiy sub'ektlari talaba va domla hisoblanadi. Ta'lim faoliyatida talabanning domla bilan baravar qatnashishi sifatli ta'lim shartlaridan biridir.

Elektron resurslarni yaratishning yuqorida ifodalangan tamoyillari elektron ta'lim vositalarini sifati va samaradorligini oshirishga imkon beradi.

1.2. Mulytimediali o'quv mahsulotlarining yaratishning metodologik asoslari va bosqichlari

Qaralayotgan bo'limda mulytimediali resurs ishlab chiqishning asosiy boqichlari va ularga metodologik yondashuvlar keltirilgan.

Yaratilayotgan ko'pchilik elektron ta'lim resurslari axborotlarning turli-tuman tiplari bilan ishlashga yo'naltirilgan. O'z ishlarini unumdorligini oshirish maqsadida pedagog-kadrlar ularga rasmlar, fotosur'atlar, tovushli fragmentlar va albatta, animatsiya va video kiritishga harakat qiladi. Resurs va texnologiyalarga muvofiq tavsiflanuvchi zamonaviy so'z "mulytimedia" ta'limda keng yoyildi va pedagoglar tomonidan o'z ishlarini amalga oshirishda foydalanilmoqda.

Mulytimedia tizimlarini yuzaga kelishi inson faoliyatining ko'pgina jabhalarida inqilob yasadi. Mulytimedia texnologiyalari o'zining eng keng qo'llanilish sohalaridan birini ta'lim doirasida qo'lga kiritdi. Mulytimediaga asoslangan axborotlashtirish vositalari bir qator imkoniyatlari bilan ta'lim samaradorligini sezilarli oshirishga qodir. O'quv materialini og'zaki bayonidan ta'lim oluvchi bir daqiqada mingtagacha axborot birligini, ko'rish a'zolarini qo'shganda esa yuz ming xuddi shunday axborot birligini qabul qilish va qayta ishlashga qodirligi ilmiy tajribalarda aniqlangan. Shuni tushinish muhimki,

boshqa ko'pgina so'zlar kabi, "mulytimedia" so'zi ham birvarakayiga bir necha turli ma'nolarga ega.

Mulytimedia- bu:

- Ishlab chiqarish tartibi, turli tipdagi axborotlarni qayta ishlash vositalarini ishlatish va tatbiq qilishni tavsiflovchi texnologiyalar;
- Turli tipdagi axborotlarni qayta ishlash va taqdim etish texnologiyalari asosiga qurilgan axborot resursi;
- Turli tipdagi axborotlarni qayta ishlash va taqdim etishga bog'langan holda ishlaydigan kompyuter dasturiy ta'minoti;
- Turli tipdagi axborotlar bilan ishlashga imkon beradigan kompyuter apparat ta'minoti.
- An'anaviy statik vizual (matn, grafika), shu bilan birga dinamik (nutq, musiqa, videofragmentlar, animatsiya va h.k) turli tipdagi axborotlarni o'zida birlashtiruvchi axborotning o'ziga xos umumlashgan ko'rinishi.

Shu taxlit, keng ma'noda "mulytimedia" termini foydalanuvchiga (bir vaqtning o'zida ham mutoala qiluvchi, ham eshituvchi, ham kuzatuvchi bo'lgan) yanada samarali ta'sir ko'rsatish maqsadiga yo'naltirilgan turli xil dasturiy va texnik vositalarni qo'llaydigan axborot texnologiyalari spektrini ifodalaydi.

Mulytimedia vositalari va texnologiyalari audiovizualy axborotlarni zamonaviy qayta ishlash usullarini tatbiq qilish bilan ta'limni unumdorligini oshirish va ta'lim oluvchilarning o'qishga bo'lgan qiziqishini orttirish imkoniyatlarini yaratadi.

Axborotlashtirish vositalarida mulytimedaning qo'llanilishi tufayli grafik, tovushli va video axborotlarni bir vaqtda ta'sir ko'rsatishi hisobiga bunday vositalar katta ta'sir kuchiga ega bo'lib, ko'ngil ochar vositalar sanoatida, turli xil idoralar kasbiy amaliyotida, uyda bo'sh vaqtda, ta'limda faol targ'ib etiladi.

1.3.Instrumentariya. Interaktivlik.

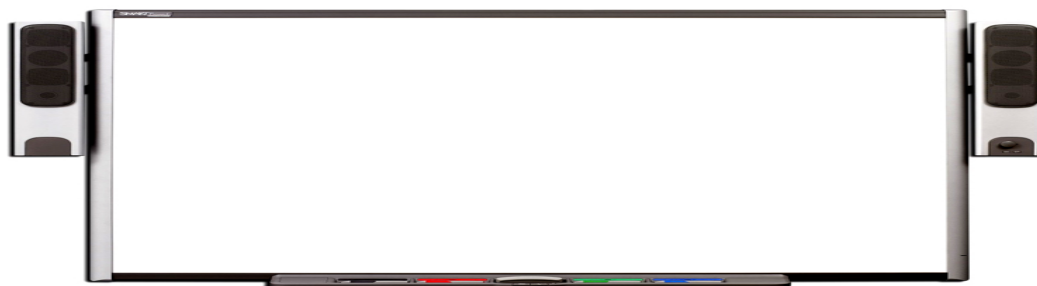
Interaktivlik - o'quv dialogi uchun bir ishtirokchi ta'limni axborotlashtirish vositasi bo'lish holatini anglatadi.

Interaktivlikni qo'lash mulytimedia- vositasini ijobiy tomonlaridan biri hisoblanadi.

Interaktivlik muayan chegaralarda axborotni taqdim qilishni boshqarishga imkon beradi: ta'lim oluvchilar individual sozlashni o'zgartirishi, natijalarni o'rganishi, shuningdek dasturning foydalanuvchiga aniq afzalliklari haqidagi so'rovlariga javob berishi mumkin.

Ta'lim oluvchilar materiallarni uzatish tezligini , takrorlashlar soni va individualy ta'lim talablarini qondiradigan boshqa parametrlarini o'rnatishi mumkin.

Bu multimedia texnologiyalarini egiluvchanligi haqida xulosa chiqarishga imkon beradi.



Interaktiv doska

Mulytimedia texnologiyalari axborotlarni ko'pgina ko'rinishlarini ma'no jihatidan va o'zaro monandlik asosida birlashtirishga imkon beradi.

Bu esa kompyuter yordamida axborotni ta'limda tez-tez foydalaniladigan quyidagi xilma-xil shakllarda taqdim qilishga imkon beradi:

- skanerlangan fotosur'atlar, chizmalar, xaritalar va slaydlarni o'z ichiga oladigan tasvirlar;
- tovush zapisi, tovushli effektlar va musiqa;
- video , murakkab videoeffektlar;
- animatsiya va animatsiyali o'xshatishlar.

Ko'pincha dialogda ikki tomon ishtirokidagi axborot almashinuvi nazarda tutiladi. Fanda dialogning kengroq tushunchasi uchraydi, uning asosiy alomati suhbatdoshlarni nutqiy xabarlar almashinuvi emas, balki bir necha fikrlarni mavjud bo'lishi hisoblanadi.

Ta'lim beradigan mulytimedia –resursi bilan ishlashning mukammallik ko'rsatkichi deb o'qituvchi bilan bitta ta'lim oluvchi dialogiga muvofiq keluvchi yagona o'zaro aloqani hisoblash mumkin.

Dialogni tavsiflaydigan asosiy psixologik omillar sifatida quyidagilarni keltirish mumkin:

- Dialog qurishning umumpsixologik tamoyillari,
 - Muloqat jarayonini tashkil qilish,
 - Tilshunoslik nuqtai-nazaridan (muloqat tilini tanlash, xabar matnini tuzish, uning shakli, o'lchami va h.k),
 - Muloqatning modalligi (axborotni bayon qilish va ta'lim oluvchilar javobini tipi)
 - Muloqatning mazmuni jihatidan.
-
- Reaktiv o'zaro ta'sir : ta'lim oluvchilar ularga taqdim qilingan holatga javob reaksiyasini namoyon qiladi. Topshiriqlar ketma-ketligi qat'iy qayd qilingan va mulytimedia-dasturini boshqarish imkoniyati uncha katta emas.
 - Faol o'zaro ta'sir: ta'lim oluvchilar mulytimelia- resursini nazorat qiladi. O'quvchilar topshiriqni qay tartibda bajarish, va mulytimedia-vositasi bilan ishlash doirasida materialni o'rganishni qaysi yo'lidan borishni o'zi hal qiladi.
 - Ikki tomonlama o'zaro ta'sir: ta'lim oluvchilar va mulytimedia-resurslari bir-biriga o'zaro moslashishga qodirdir.

Mulytimedia vositalarini interaktivligi o'qituvchi va ta'lim oluvchilar tomonidan ta'lim jarayoniga ta'sir etish imkoniyatlarining keng doirasini va o'quv materiallari tarkibini to'ldirish ko'zda tutadi.

1.5. Bitiruv malakaviy ishining maqsad va vazifalari

Zamonaviy ta'lim sharoitida talabalar mustaqil ishlash faolligini oshirish, ularning ijodiy qobiliyatlarini rivojlantirish ilg'or pedagogik texnologiyalaridan hamda yangi avlod o'quv adabiyotlaridan foydalanishni talab etadi.

Axborot va kommunikatsiya texnologiyalari ta'lim oluvchilar o'qitish darajasi samaradorligini oshirish ehtiyojlaridan kelib chiqib mazkur bitiruv malakaviy ishida quyidagi **maqsadlar va vazifalar** ilgari surildi:

Fanni o'qitishdan maqsad va vazifalari.

O'qitish maqsadi – Ta'lim jarayonida Axborot kommunikatsiya texnologiyalari yordamida mulytimediali elektron amaliyot mashg'ulotlarini yaratish ,dastur maxsulotlarini ishlab chiqish usullarini, ularni ishlatish bilan bog'liq masalalarni yechish, foydalanuvchi interfeyslarini ishlab chiqish, sozlash, testdan o'tkazish va ularni kuzatish masalalarini yechishni o'rgatishdan iborat. Dastur komponentalarini yaratishda programmalash texnologiyasining zamonaviy usul va uskunalari o'rgatish.

O'qitishning asosiy vazifasi dastur loyixalarini yaratish, hujjatlarni tayyorlash va amaliy programma ta'minotlarini to'g'ri ishlata olish ko'nikmalarini yaratishdan iborat.

Fan bo'yicha talabalarning bilim va ko'nikmasiga qo'yiladigan talablar:

Fanni o'rgangandan so'ng talaba quyidagilarni tassavur qilishi kerak:

- Zamonaviy dasturlash maxsulotlarini programmalash texnologiyasini;
- Masalalarni algoritmlash va modellashtirish usullarini;
- Optimal programmalash tilini tanlab olishni;
- Tayyor dastur maxsulotlarini programmalash, testdan o'tkazish, sozlash va xizmat ko'rsatish usullarini bilishi va ishlata olishi;
- Dasturlarni funksional xarakteristikalariga talablar qo'yishni;
- Dasturni ishlab chiqish bosqichlarini aniqlashni;
- Algoritmni blok-sxemalari va funksional sxemalari;
- Zamonaviy shaxsiy kompyuterlarni umumiy maqsadli programma maxsulotlarini.

Quyidagi ko'nikmalarga ega bo'lishi kerak:

- Dastur algoritmini blok-sxemasini ishlab chiqish;
- Dasturni tanlab olingan algoritmiga test misollarini ishlab chiqish;
- Dasturdagi xatoliklar topish va tuzatish.

Fanni o'quv rejasidagi boshqa fanlar bilan bog'liqligi:

Kurs materiali: «Oliy matematika», «Algoritmik tillar va dasturlash», «Informatika» kabi umumta'lim fanlardagi bilimlarga asoslanadi. Bu kursni o'zlashtirish talabalarni murakkab bo'lgan turli ilmiy va muxandislik masalalarini yecha olishga nazariy bilim sifatida asos bo'lib, «Modellashtirish», «Ma'lumotlar, bilimlar bazasi va ekspert tizimlari» fanlari uchun asosiy manba bo'lib hisoblanadi.

Fanni o'qitishdagi foydalaniladigan yangi pedagogik texnologiyalar

Har bir talabani bilimini mustaxkamlash va mustaqil fikrlash qobiliyatini oshirish uchun kursni o'qitish mobaynida zamonaviy o'qitish texnologiyalaridan foydalaniladi. Asosan o'qitish maxsus kompyuter sinflarida, hisoblash texnikasiga asoslangan holda olib boriladi.

I-bob bo'yicha xulosa

Shu kabi masalalarni o'rganish jarayonida shularga amin bo'ldikki, elektron ta'lim vositalari yordamida o'qitish texnologiyasining asosiy xususiyatlari va afzallik jihatlari quyidagilardan iborat ekan:

- ushbu texnologiya ta'lim oluvchining mustaqil fikrlashi va ijodiy qobiliyatlarini rivojlantirishga yo'naltiriladi;
- o'qituvchi ta'lim jarayonida maslahatchi sifatida qatnashadi;
- o'qitish jarayonida axborot vositalari va zahiralari faol integratsiyasi ta'minlanadi;
- o'qishga qiziqish ortadi;
- ta'lim olishning qulayligi va natijaviyligini oshadi;
- ta'lim oluvchining mustaqil ishlash va izlanish malakalari shakllanadi.

Kompyuterli o'qitishda har bir ta'lim oluvchi ma'lum bir vaqtda «virtual talaba» sharoitida bo'ladi. Demak talabaning virtual muhitga ko'nikib borishi ta'minlanishi kerak.

Elektron darsliklar maxsus fanning mazmuni va mohiyatiga diqqatni jalb etgan holda ko'p sondagi ma'lumotlarni qarab chiqish va ko'proq amaliy mashg'ulotlarni bajarishga imkon yaratadi. Murakkab hisoblashlar va almashtirishlardan xalos etadi. O'rganishning barcha bosqichlarida o'z-o'zini tekshirib ko'rish uchun keng imkoniyatlar yaratadi. Elektron darslik tajribali o'qituvchi vazifasini, cheklanmagan tushuntirishlarni, sanoqsiz takrorlashlarni, eslatishlarni taqdim etgan holda bajaradi.

Elektron darslik vositasida o'qitishda quyidagi qulay imkoniyatlar yaratiladi:

- ❖ kompyuterli qo'llab-quvvatlashlardan foydalangan holda, ta'lim oluvchilar katta miqdordagi topshiriqlarni bajarishga ulguradilar, echimlar va ularning grafik talqinini tahlil qilish uchun sarflanadigan vaqtni tejaydilar;
- ❖ kompyuter oldida mustaqil ish shaklida mashg'ulot o'tkazish imkonini beradi;
- ❖ ta'lim oluvchilar bilimi tez va samarali nazoratdan o'tkaziladi.
- ❖ o'qituvchiga nazariy va amaliy mashg'ulotlarda o'zining xohishi bo'yicha hajm jihatidan kichik, ammo tarkibi bo'yicha o'ta muhim bo'lgan materiallarni etkazishga, ta'lim oluvchilarning auditoriya mashg'ulotlari doirasidan tashqari o'rganish mumkin bo'lgan masalalarni echishda mustaqil shug'ullanishlariga sharoit yaratiladi;
- ❖ o'qituvchini uy topshiriqlari, turli hisoblashlar va nazorat ishlarini tekshirishdek mashaqqatli mehnatdan ozod etadi;
- ❖ ta'lim oluvchi bilan, ayniqsa, uy topshiriqlari va nazorat ishlari qismiga oid ishlashni individuallashtirish mumkin.

Elektron darsliklar vositasida o'qitish asosan ta'lim oluvchilarning kompyuter bilan ishlash bo'yicha tayorgarlik darajasiga va aqliy imkoniyatiga moslashtirilishi kerak.

Elektron darsliklar vositasida o'qitishni takomillashtirish uchun quyidagi tamoyillarga e'tiborni qaratish kerak:

- qo'shimcha elektron zahiralarni, ma'lumotlar va kutubxonalarni yaratish, tarmoqdan axborotni izlashni ta'minlovchi maxsus dasturiy ta'minotni ishlab chiqish kerak;
- o'qituvchilarning o'quv-uslubiy ishlarini takomillashtirish, internetdan foydalanish, axborot texnologiyalari va psixologiya sohalari bo'yicha mutaxassislar bilan hamkorlikni o'rnatish;
- elektron darslikni fan-texnika va texnologiyalarning so'nggi yutuqlari bo'yicha ma'lumotlar bilan muntazam to'ldirib borish;
- elektron darsliklar vositasida o'qitishda ilg'or pedagogik texnologiyalar va faol metodlarni qo'llash;
- elektron darsliklar vositasida o'qitish metodikasida bilimlarni baholash mezonlari muhim muammo hisoblanadi. Ushbu metodika asosan, mustaqil ta'limga yo'naltirilganligi sababli o'qituvchilarning baholashni tashkil etishdagi faol va mas'uliyatli ishtiroki talab etiladi. Chunki baholash jarayonida nafaqat testlar natijalari, balki o'quvchilarning faolligi hamda mustaqil ishlash qobiliyati ham inobatga olinishi kerak.
- maxsus fanning o'quv dasturi elektron darslik vositasida o'qitish metodikasiga moslashtirish lozim

II BOB. MA'LUMOTLARNI DASTURLASH TEXNOLOGIYASI KURSI ASOSIDA MUYTIMEDIALI ELEKTRON RESURS YARATISH

2.1. Microsoft Visual Studio 2005

Microsoft Visual Studio 2005 ish muhiti ko'p dasturlash tillarini jamlovchi va NET platformasi bilan ishlovchi ish muhiti bo'lib unda dasturlashning o'ziga hos qulayliklari mavjud. Masalan ko'p dasturlash tillarini mujassamlashganligida J#, C#, C/C++, Visual C++ , Visual Basic, ADO.NET, ASP.NET dasturlash tilarini bundan tashqari, SQL server , HTML redaktorlari va .NET Framework 2.0 versiyasi borligi dasturchini ishini yanada onsonlashtiriladi. Bu kompleksli ishlash muhitida malumotlar bazasi bilan ishlovchi, web saxifalar bilan ishlovchi dasturlarni ham yozish mumkin, ma'lumotlar bazasi bilan to'g'ridan to'g'ri aloqani o'rnatib unda jadvallar yaratish ularga o'zgartirishlar kiritish, web saxifalarni loyixalashtirish va ularni yaratish mumkin. Microsoft Visual Studio 2005 ish muhitining asosiy qulayligi unda sichqoncha bilan dasturlash mumkin bunda sichqoncha yordamida kerakli tugma, dialoglar va boshqalarni ekranga sichqoncha yordamida tashlash kifoya. Dev C++ ish muhitida bunday xususiyat yo'q va bu ish muhiti faqatgina C/C++ dasturlash tili bilan ishlaydi. Microsoft Visual Studio 2005 ish muhitida ishlash uchun kuchli kompyuterlar kerak tezkor xotirasi yuqori bo'lgan. Dev C++ ish muhitiga esa unchalar kuchli bo'lgan kompyuterlar kerak emas , bu ish muhiti kam resurs talab etadi.

Visual C++ dasturlash tili

Visual C++ dasturlash tilida dasturlash C/C++ tilida dasturlagan kabi lekin oynali formalar yaratganda o'ziga yarasha qulaylikka ega bunda C/C++ dasturlash tilida fagatgina **STANDARTNIY** funksiyalar to'plamidan va Win API funksiyalardan foydalanish mumkin. Visual C++ da esa **STANDARTNIY** funksiyalar to'plamidan va Win API funksiyalardan tashqari .NET Framework funksiyalaridan foydalanish

mumkin lekin bu dasturlashtilida yozilgan dasturlar .NET Frameworksiz ishlamaydi bu esa o`ziga yarasha noqulayliklar tug`diradi, biridan yutsak ikkinchisidan yutqizamiz. C/C++ da oynali dasturlar yordamida dasturlaganda har bir oynani va unda ko`rinadigan barcha narsani ko`rinishi va xususiyatlarini kiritish, ularni e`lon qilish, ketma-ketligini ko`rsatish, joy pozitsiyasini yozib kiritish kerak va ular qayerda qay holda joylashganini faqatgina tasavvur qila olamiz ya`ni dasturlagan vaqtimizda ularni ko`rinishini ko`ra olmaymiz. Visual C++ dasturlash tilida oynalar tayyor yaratilgan ularni kerakli ko`rinishga sichqoncha yordamida keltirib ko`rishimiz mumkin bu esa dasturchiga vaqtdan yutishda yordam beradi lekin dasturning ishlash tezligidan va shunga bog`liq ravishda uning hajmi oshib ketishiga sabab bo`ladi.

2.2.Captivate dasturiy ta`minoti haqida

Captivate shunday dasturiy ta`minoti yordamida siz osongina video yarata olishingiz mumkin. Navbatdagisi - Captivate ishlash muhitining qisqacha ma`lumoti. Shunga e`tibor berish kerakki: Siz shuningdek Captivateda ishlayotgan yo`lingizni sozlashingiz mumkin.

Tushunchalar.Captivate ikkita asosiy tushunchaga ega: videostudiyaning stsenariy bo`limi tushunchasi va tushunchani tahrir qiladi (Diafilm bilan).

Videostudiyaning ssenariy bo`limi: Bu tushuncha siz video start sahifasidan boshlab ochganingizda paydo bo`ladi. Video videoni yaratadigan slaydlardan iborat kichkina tasvirlar bilan ochiladi. Videostudiyaning ssenariy bo`limi tushunchasi barcha slaydlarni video doirasida birdaniga ko`rishning oson usulini ta`minlaydi.

Videoni yozib olganingizdan so`ng siz vaqtingini tahrir tushunchasida o`tkazasiz .

Captivate da video yaratish uchun sizning kompyuter tizimingiz apparat vositalari, dasturiy ta`minot va boshqa jihozdarni o`z ichiga olishi kerak.

Ishchi dastur uchun texnik talablar:

- 1.Tezkor xotira hajmi : 512 M bayt.
- 2.Processor : 2.4 G Hz
- 3.Video hotira : 128 M bayt

4.Doimiy hotira : 8 G baytni tashkil qilishi kerak.

2.3. “COURSE LAB” dasturiy ta’minoti haqida

Hozirgi vaqtda mulytimediali elektron resurslar yaratish uchun juda ko’plab dasturiy ta’minotlar mavjud. Shulardan eng qulay hisoblanuvchi dasturiy ta’minot bu “CourseLab”dasturiy ta’minoti hisoblanadi.

CourseLab- elektron o’quv kurslari tahrirlagichi .

Boshlang’ich shartlar:

CourseLab-bu kuchli va shu bilan birga foydalanishda oddiy bo’lgan, internet tarmog’ida, masofaviy ta’lim tizimida, kompakt-disk yoki boshqa har qanday tashuvchida foydalanishga mo’ljallangan interaktiv o’quv materiallari(elektron kurslar) ni yaratish vositasi (instrumenti)dir.

CourseLabning xossalari:

- WYSIWYG-doirasida o’quv materiallarini yaratish va tahrirlash, nima ko’rinsa , natijada shunga ega bo’linadi;
- O’quv materiali avtoridan HTML yoki qandaydir bir dasturlash tili bilishni talab qilmaydi;
- Ob’ektlilik yondoshish - amalda har qanday murakkab bo’lgan o’quv materiallarini bolalar kubiklaridek oson tuzishga imkon yaratadi;
- Stsenariylardan foydalanish murakkab ko’pob’ektlilik o’zaro bog’liqliklarni ancha soddalashtirish imkonini beradi;
- Mexanizm o’z ichiga test tuzishni ham oladi;
- Foydalanuvchi tomonidan yaratilgan ochiq kodli interfeys va shablonlar kutubxonasini oson kengaytirishga imkon beradi;

- Kursga Macromedia® G'lash®, Shockwave®, Java®, turli xil formatdagi videolardan iborat bo'lgan har qanday Rich-mediani kiritish imkoniyati borligi;
- Tovushni baravar birga olib borilishini kiritish va sinxronizatsiyalash oddiy mexanizmlari borligi;
- Microsoft® PowerPoint® formatidagi taqdimotli o'quv materialini import qilish imkoniyati;
- Harakatlarni tavsiflashning oddiy tili;
- Tahrirlagich tajribali foydalanuvchiga to'g'ridan-to'g'ri JavaScriptning qo'shimcha imkoniyatlaridan foydalanishni taqdim qiladi;
- Elektron o'quv kurslarini namoyish qilishda JavaScriptni mavjud bo'lishini talab qilmaydi.

Websoft CourseLab masofaviy ta'limda qanday o'rin egallaydi? Elektron o'quv kursi boshdan oxirigacha Websoft CourseLab tahrirlagichida ishlab chiqilishi mumkin. Tayyor o'quv kursi kursning tuzilmasi haqidagi ma'lumotlar saqlanadigan maxsus fayllar yaratish yordamida masofaviy ta'lim tizimiga import qilinishi mumkin.

I-bob bo'yicha xulosa

Dasturchi ishlash muhitini bilishi va dasturlash tilining eng oddiy fuksiyalari bilan tanishgan bo'lishi kerak. Bundan tashqari dasturchi dasturning ishlash algoritmini tuzishi kerak.

Dasturchi ishlash muhitini tanlashda o'ziga qulay va tushunarli- bunda dasturlash tili to'g'ri ishlashi va to'la imkoniyatlari qo'llana olinadigan muhitni tanlashi kerak.

Visual Studio 2005 (VS2005) dasturida dasturlash o'ziga yarasha qulayliklarni yaratadi. Shuning uchun biz shu muhitda dasturlash tilini visual C++ da dasturlashni amalga oshiramiz. Siz esa o'zingizga yoqqan muxitda, dasturlash tilida va albatta

o`qituvchining talablarini inobatga olgan holda ishlashingiz mumkin. Misol uchun Borland C++ , turbo C++, Dev C++, Visual C++, Turbo paskal, boorlend Delfy 2009 va Visual Studioning boshqa versiyalarida ishlashingiz mumkin.

VS2005 dasturini ishga tushiramiz va VS2005 dasturining asosiy oynasi va bosh sahifa ochiladi (1 - rasm).VS2005 dasturida ko`p dasturlash tillari yordamida dasturlash mumkin misol uchun C#, C++, J#, VB va boshqa dasturlash tillari yordamida dasturlash mumkin.

Captivate dasturiy ta`minoti o`ziga yarasha qulayliklarga ega. Foydalanuvchi uchun juda sodda hisoblanadi. Ushbu dasturiy ta`minot boshqa dasturiy ta`minotlarga (masalan camtasia studio) qaraganda sifatli hisoblanadi. Ishlashi bir tekisda.

III- BOB.AMALIYOT ISHLARNING DASTURIY TA'MINOTI

3.1 Dastur maqsadini aniqlash. «Masala qo'yilishi » hujjati.

Reja:

1. “Masala qo'yilishi hujjati”;
2. Algoritm tanlash;
3. Dastur maqsadini aniqlash;
4. Dasturlash strukturasi loyihalashtirishning asosiy prinsiplari.

Ishdan maqsad: DT fani, dasturiy ta'minot yaratishdagi muammo, maqsadlarini aniqlash va masalani qo'yilishini shakllantirish hujjatlarini yaratishdan iborat.

Nazariy qism

Muammoni yechish yoki maqsadga erishish uchun zarur bo'lgan talablarni (xususiyatlar, sifat va imkoniyatlar) ishlab chiqiladi.

O'z ichiga quyidagilarni oluvchi spetsifikatsiyalarni ishlab chiqish talab etiladi:

- dasturning maqsadi;
- sistema funksiyalarini tavsiflash;
- kiritiladigan va chiqariladigan ma'lumotlarning spetsifikatsiyalari;
- nazorat talablari (testlardan foydalagan holda nazorat qilish);
- hujjat turlari va soni.

Bu jarayonda, sistema tayyor holda ega bo'lishi lozim bo'lgan xususiyatlar aniqlanadi, sistemaning funksiyalari va interfeysning xarakteristikalarini tavsiflanadi.

Masalani yechishni boshlashdan avval uni aniq tariflash lozim. Birinchi navbatda - bu boshlang'ich va natijaviy ma'lumotlarni aniqlash hamda:

- a) ma'lumotlarni yig'ish;
- b) maqsadni aniqlash kerak.

Masalani qo'yilishini yanada batafsilroq ko'rib chiqishda quyidagilarni inobatga olish lozim:

- yechimni aniqlashni;
- ma'lumotlar yetarliligini hisobga olish kerakligini;
- ruxsat etilgan xatolarni va h.k.

“Masala qo'yilishi hujjati” foydalanuvchi tomonidan berilgan talablar spetsifikatsiya deb ataluvchi maxsus hujjatlarda rasmiylashtiriladi.

Masalani qo'yilishi va tarkibi.

Loyihalash vaqtida har - xil muammolarni yuzaga kelishini hisobga olish kerak. Asosan bu muammolar shu dasturni buyuruvchilarning aniq maqsadga ega bo'lmasliklaridan kelib chiqadi. Foydalanuvchi noaniq bergan so'rovga aniq va to'g'ri javob beruvchi dastur - bu eng yuksak dasturdir. Vazifa yaxshi yoki yomon deb aniqlanishi mumkin. Yomon shakllantirilgan vazifaga aniq bir dasturni loyihalab bo'lmaydi.

Aniq belgilangan vazifa esa loyihalash davomida ortiqcha ishdan vaqt sarflashdan voqif bo'ladi. Vazifa dasturning xajmiga qarab ixcham va tushunarli bo'lishi kerak. Agar dastur kichkina bo'lsa vazifa bir necha qatordan, katta bo'lsa butun - bir kitobdan iborat bo'lishi mumkin. Bu tavsiflar katta songa ega bo'lgan hujjatlarga asos bo'ladi. Keyinchalik vazifalarga kiritilgan o'zgartirishlar yoki qo'shimcha vazifalar dasturni buzibgina qolmay uni tugatishni umuman sekinlatib qo'yadi. Vazifani aniqlashda qo'yilgan loqaydsizliklar keyinchalik katta muammolar keltirib chiqaradi.

Algoritm tanlash.

Algoritmni to'g'ri tanlash bu dasturni to'g'ri va samarali bo'lishiga qo'yiladigan asosiy qoidadir. Bunda dasturni tilini va tarkibini to'g'ri tanlash mumkin. Shunday qilib yaxshi algoritm kerakli, lekin dastur uchun yetarli sharoit emas. Yaxshi algoritmni qanday tanlash kerak?

- Birinchidan - miyaga to'g'ri kelgan algoritmgaga dastur tuzma ;

- Ikkinchidan-bir necha variantlarni ko'rib chiqib, ularning ichidan eng yaxshisini tanla.

Ma'lumotlarni tavsiflash.

Yaxshi tavsiflangan ma'lumotlar dasturni ancha qisqartiradi.

- Demak, ma'lumotlarni massivni tashkillashda zarur bo'lgandagina qo'llash kerak;
- Boshqacha usulida tayanch va ko'rsatkichlardan foydalanish kerak;
- Ma'lumotlar haqida ko'rilayotgan vazifaga mos holda taassurotga ega bo'lish kerak.

Xisobda xar bir ma'lumotning tarkibini hamma tilga ko'chirish mumkin. Lekin siz mo'ljallangan ma'lumot tarkibini mujassamlagan dasturlash tilida foydalanishingiz maqulroq.

Barcha talablar tabiiy tilda keltirilib, aniq ko'rinishga ega bo'lishi shart. Masala qo'yilishi hujjati – bu hujjat eng birinchi yoziladi va quyidagi bo'limlardan iborat:

Maqsadlarni aniqlash

Maqsad sifatida dasturiy maxsulot uchun tanlab olingan aniq yo'nalish tushuniladi. Maqsadlarni aniqlash jarayoni bilan bir xil yo'nalishda olib boriladi. Mahsulotning maqsadi, ya'ni foydalanuvchi nuqtai nazaridan natijaviy yechimga olib kelish yo'llari. Bunga loyihalash grafigi, narxi, testlash jarayoni kiritiladi.

Dasturiy mahsulot maqsadini quyidagi guruhlarga bo'lish mumkin:

Ishonchlilik – bunga ma'lum bir vaqt mobaynida dasturiy mahsulot uzilishsiz ishlash mezonini kiritiladi.

Umumiylik – bu korxonaning quvvati, hodimlar soni bilan xarakterlanadi.

Tayyor tizimning psixologik omillari – bu dasturiy mahsulotning tushunuvchanligini, ishlatishga qulaylik, noto'g'ri ishlatishdan himoyalash, bularning hammasi dasturiy mahsulot hajmini bir muncha oshiradi, shu bilan birga dasturiy mahsulotning samaradorligini ham oshiradi.

Moslashuvchanlik – bu ko'rsatkich yordamida dasturiy mahsulotning kengaytirilishi amalga oshiriladi, ya'ni mahsulot unga zarur bo'lgan holda, oddiy

ravishda foydalanuvchining funksiyasi kiritilishi mumkin bo'lgan ko'rinishda yaratilishi lozim.

Kuzatib borish - bunda vaqt mezoni hisobga olinadi, shu bilan birgalikda dasturdagi xatoliklarni tuzatish uchun ketgan vaqt bunga kiradi.

Xavfsizlik - bunda zarur axborot saqlangan fayllar yoki papkalar boshqa fayllar tomonidan tasodifan o'chirilib ketilishi ehtimolligi ko'zda tutiladi.

Dasturiy mahsulotning tannarxi – bunda dastlabki ishlab chiqarishga va kuzatib borishga ketgan mablag'lar hisobga olinadi. “Odatda mahsulotning narxi undagi mavjud xatoliklarga teskari proporsional bo'ladi” – **Van Tassel**.

Shunga ko'ra, dasturiy mahsulot qanchalik ishonchli, umumiy, moslanuvchan, xavfsiz bo'lsa, uning narxi shuncha yuqori bo'ladi.

Mahsulot maqsadi

Rezyume – bunda ishlab chiqilayotgan mahsulotni umumiy yo'nalishi tavsiflanishi kerak.

Foydalanuvchini aniqlashda ishlab chiqilayotgan dasturiy mahsulot katta bo'lsa va uning foydalanuvchilari sifatida turli guruh foydalanuvchilari bo'lishi mumkin bo'lsa, u holda uning o'rni aniqlanmog'i lozim.

Foydalanuvchilar ketma-ketligi – bunda dasturiy mahsulot foydalanuvchi nuqtai nazaridan mahsulot tomonidan taminlanadigan barcha funksiyalar tavsiflanadi.

Kalendar reja hujjati – bunda dasturiy mahsulot yozish bosqichlari aniqlanadi, har bir bajargan bosqich to'g'risida ma'lumot beradi.

Samaradorlik – bunda ishlab chiqarish maqsadi tushuniladi, ya'ni vaqt, o'tkazuvchanlik imkoniyatlari, moslashuvchanlik vositalari hisoblanadi.

Dasturiy mahsulotning boshqalar bilan moslashuvchanligi – bunda davlat va xalqaro standartlarga moslashish talab qilinadi.

Dasturiy mahsulot konfiguratsiyasi – ya'ni yaratilayotgan dasturiy mahsulot biror-bir boshqa kattaroq tizimning tarkibiga kirishi va o'zi alohida ishlashi taminlanishi lozim.

Tasodifiy murojaatlardan himoyalanih – parol.

Xizmat ko'rsatish.

Dasturiy mahsulot ishini yo'lga qo'yish – o'rnatish, sozlash.

Loyiha maqsadi. Loyiha maqsadi bo'limida quyidagi savollarga to'liq javob berish shart:

- Loyihaning taxminiy narxi qanday?
- Loyihaning taqvimiy rejasi.
- Testlash jarayonining maqsadi.
- Moslashuvchanlik jarayonini maqsadi.
- Kuzatib borish jarayonidagi ro'y berish mumkin bo'lgan muammolar.
- Dasturlash bosqichini ishonchliligi.
- Loyihani ishlab chiqarishdagi ichki hujjatlar.

“Loyixani maqsadini” aniqlashda quyidagi ko'rinishdagi xatoliklar keng tarqalgan.

- Maqsadlar bir-biri bilan qarama-qarshilikda bo'ladi.
- Maqsadlar oshkor ravishda keltirilmaydi.
- Loyixa maqsadlari ko'zda tutilmaydi – G.Ivanova kitobidan.

Dasturlash mahsulotini loyihalash. Loyihalash so'zi lug'atda reja asosida biror-bir ko'rinishda bo'ladi. Dasturiy mahsulotni loyihalash ierarxik dekompozitsiya asosida olib boriladi. Bunda biror-bir murakkab ko'rinishga ega bo'lgan loyiha oddiy ko'rinishga ega bo'lgan kichikroq muammolarga bo'linadi.

Bu muammolar o'z navbatida yanada maydaroq qismga bo'linadi. Ierarxik dekompozitsiya tabiiy ko'rinishlarga ega bo'lib, amaliyotda juda ko'p ishlatiladi.

Ierarxik dekompozitsiya xosil qilish quyidagi qoidalarga bo'ysunadi:

- Iyerarxiyaning har bir sathida keltirilgan reja yoki loyiha bu sathda tugallangan ko'rinishga ega bo'lishi kerak;

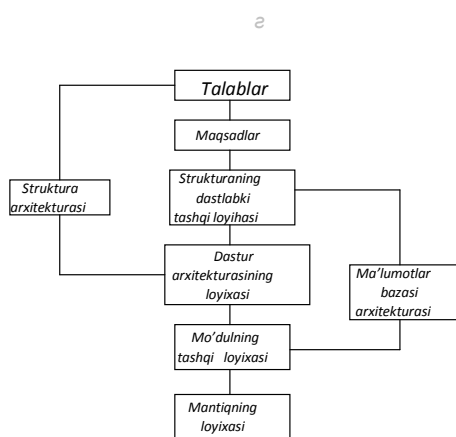
- Iyerarxiyaning ixtiyoriy sathida keltirilgan axborot ma'lumoti bir funktsiya yoki muammodan tashkil topgan bo'lib, shu sathga mos bo'lishi kerak;

Dasturiy mahsulotning ierarxik dekompozitsiyasi quyidagi yo'nalishlarda olib boriladi:

- Dasturiy maxsulotni yuqoridan pastga qarab loyixalash.
- “Dasturiy maxsulot loyixalash uni tashkil qilishda asosiy bosqichlaridan biri hisoblanadi. Yaxshi loyixalangan dasturni yomon yozish mumkin. Ammo yomon loyihalangan dasturni yaxshi yozish mumkin emas”. Van Tassil.

Dastlabki maqsadlar – aniqlashdagi xatoliklar keyinchalik faqat dasturiy ta’minotni emas, balki, uni ham katta o’zgartirishni taqozo etadi.

Shunga ko’ra dasturiy loyihalar 3ta guruhga bo’lish mumkin.



Dasturlash strukturasi loyihalashtirishning asosiy printsiplari quyidagi sxemada aks ettirilgan:

Nazorat savollari:

1. Dasturiy mahsulotga tegishli asosiy hujjatlar haqida ko’rsating.
2. “Masala qo’yilishi” hujjati nima?
3. Maqsadlarni aniqlashni ko’rsating.
4. Algoritm qanday tanlanadi?
5. Ma’lumotlarni qanday vaziyatlarda qo’llash kerak?
6. Ishonchlilik, umumiylik va moslashuvchanliklarni tariflang.
7. Foydalanuvchilar ketma-ketligi, kalendar reja hujjati va samaradorlikni ta’riflang.
8. Loyiha maqsadi haqida ma’lumot bering.

Tayanch iboralar: Masala qo’yilishi hujjati, dasturiy mahsulot, loyiha maqsadi, loyihalash, ierarxik dekompozitsiya, algoritm.

1 – topshiriq tavsifi:

1 – topshiriqda dasturchi ekranga chiqarish va klaviatura yo`rdamida qiymatlarni kiritish va eng oddiy arifmetik amallarni bajarishni o`rganadi. Bu - dasturchi keyinchalik dasturlash mobaynida kerakli ma'lumotlarni ekranga chiqarish va kerakli ma'lumotlar ustida amallar bajarishda kerak bo`ladi.

1- topshiriqni yechishdan avval dasturchi ishlash muhitini bilishi va dasturlash tilining eng oddiy fuksiyalari bilan tanishgan bo`lishi kerak. Bundan tashqari dasturchi dasturning ishlash algoritmini tuzishi kerak.

Dasturchi ishlash muhitini tanlashda o`ziga qulay va tushunarli- bunda dasturlash tili to`g`ri ishlashi va to`la imkoniyatlari qo`llana olinadigan muhitni tanlashi kerak.

Dasturchi dasturlash tilini tanlaganda o`zi biladigan va muallim talabiga ko`ra tanlashi kerak. Bu topshiriqni yechishni har qanday dasturlash tilida amalga oshirsa bo`ladi. Farqi , birida tezroq va onsonroq yo`l bilan natijaga erishish mumkin bu qay birini yaxshi bilishingiz va o`qituvchinig talabiga bog`liq.

1- topshiriqni yechish algoritmini ko`rib chiqamiz:

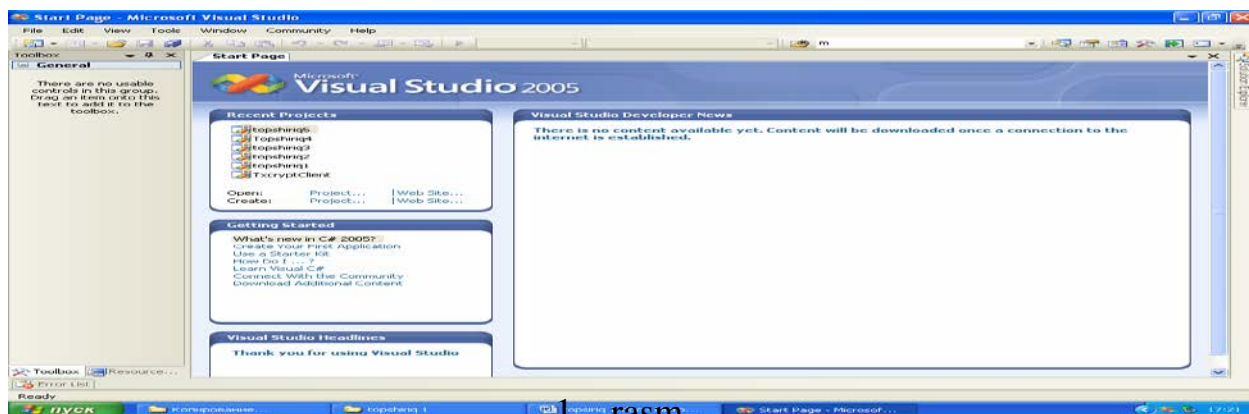
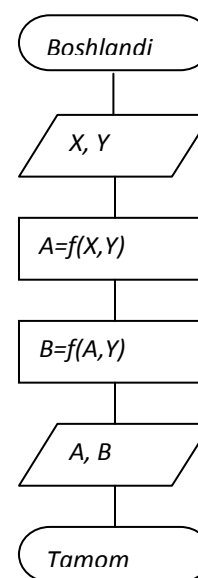
Har bir misolda ma'lum bir arifmetik amallarni bajarib 2 sonning qiymatini hisoblash kerak. Bu sonlarni hisoblashda 2 ta noma'lum son bor, noma'lum sonlarning qiymatini foydalanuvchi kiritishi kerak. Noma'lum sonlar kiritilgandan keyin arifmetik amallarni bajarib 2 sonning qiymatiga ega bo`lamiz va bu qiymatni ekranga chop ettirish kerak, ya'ni foydalanuvchi topshiriq natijalarini ko`rishi kerak va shu bilan dastur tugaydi. Endi bu ketma ketliklarni blok sxema ko`rinishida yozamiz. Misol uchun 1 – topshiriqning 1 – misolini yechib ko`ramiz.

Misolni yechishdan avval misolning maqsadini, nima so`rallyaptganini, nimaning hisoblash kerakligi, nimalar ma'lum va nimalar noma'lumligini baholaymiz. Baholab bo`lgandan keyin o`z imkoniyatlarimizni baholab qaysi dasturlash tilida, muxitda, proyekt turini va algoritmini aniqlab qaror qabul qilamiz. Ushbu qarorimiz asosida dasturni yozamiz, noma'lum qiymatlarni o`zlashtirish, hisob kitob qiymatlarini ekranda chop ettirish va foydalanuvchi bilan muloqotni o`rnatib, dasturni ishlashini tekshiramiz.

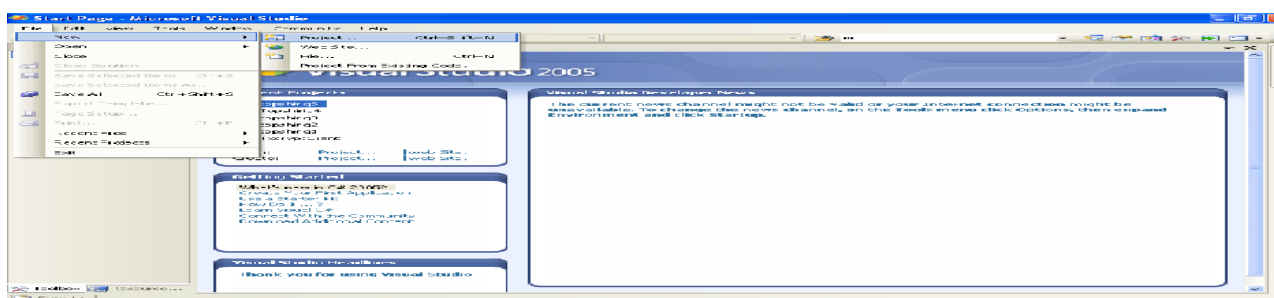
Visual Studio 2005 (VS2005) dasturida dasturlash o`ziga yarasha qulayliklarni yaratadi. Shuning uchun biz shu muhitda dasturlash tilini visual C++ da dasturlashni amalga oshiramiz. Siz esa o`zingizga yoqqan muxitda, dasturlash tilida va albatta o`qituvchining talablarini inobatga olgan holda ishlashingiz mumkin. Misol uchun Borland C++ , turbo C++, Dev C++, Visual C++, Turbo paskal, boorlend Delfy 2009 va Visual Studioning boshqa versiyalarida ishlashingiz mumkin. VS2005 dasturini ishga tushiramiz va VS2005 dasturining asosiy oynasi va bosh sahifa ochiladi (1 - rasm). VS2005 dasturida ko`p dasturlash tillari yordamida dasturlash mumkin misol uchun C#, C++, J#, VB va boshqa

dasturlash tillari yordamida dasturlash mumkin. Dasturlashni boshlashdan oldin yangi proyekt yoki yaratilgan proyektni ochib dasturlashimiz mumkin. Biz bu misolda yangi proyekt yaratib dasturlashni boshlaymiz.

Blok sxema:

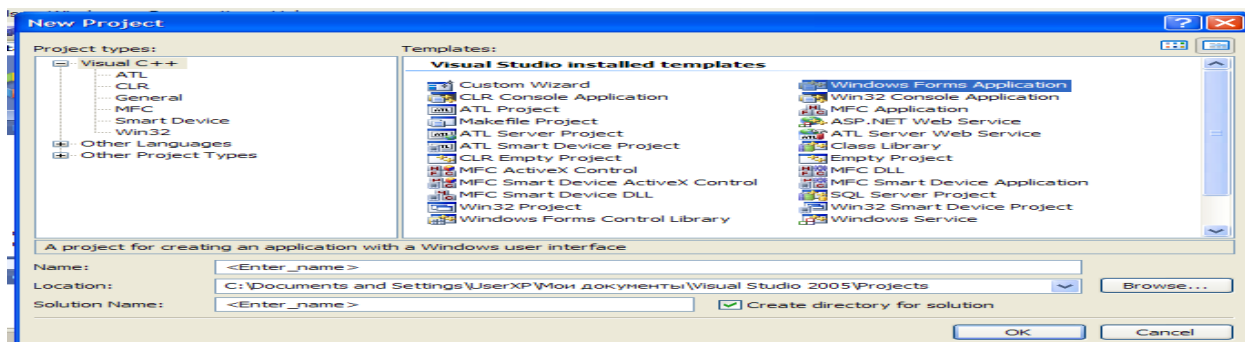


Yangi proyektni yaratish fayl menyusida yoki bosh sahifadan yangi projekt oynasini ochish orqali amalga oshiriladi. Fayl menyusida orqali ochish **File –>New –>Project** orqali amalga oshiriladi bu 2- rasmda ko`rsatilgan.



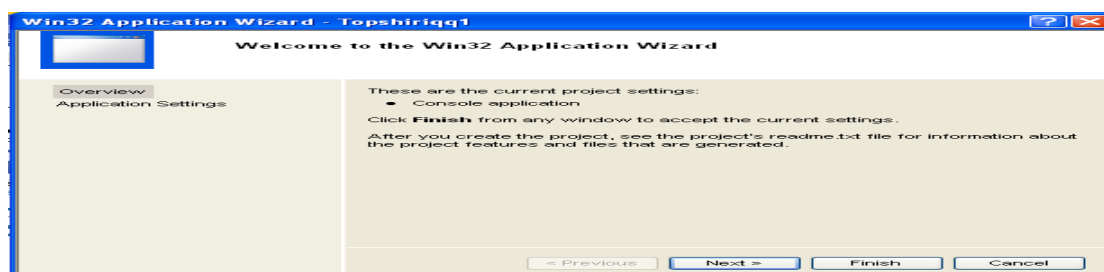
2 – rasm

Yangi projekt oynasida (3 - rasm) dasturlash tilini, projekt turini tanlash, projekt nomini va qayerda saqlanishini ko`rsatish kerak.



3 – rasm

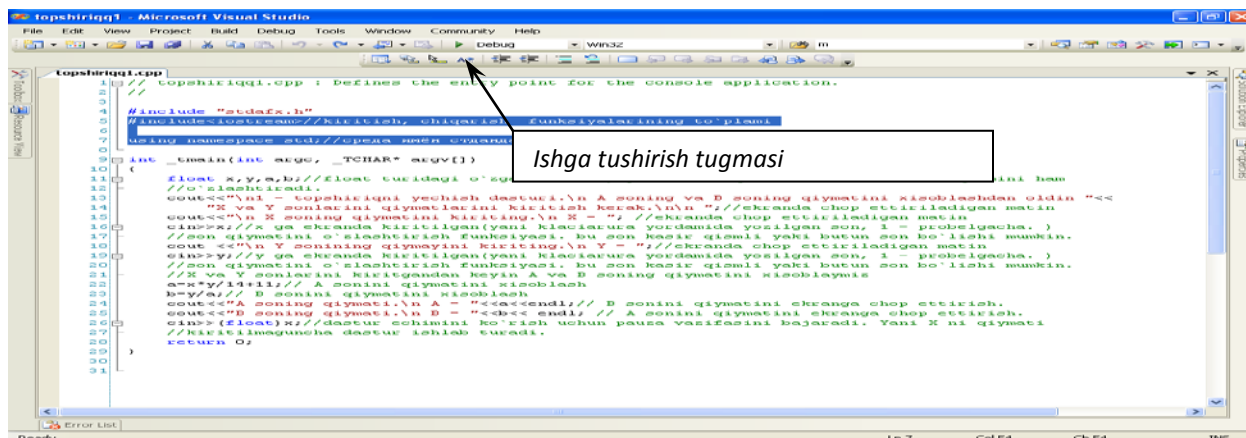
Biz dasturlash tilining Visual C++, projektini turini Win32 Console Application, nomini Topshiriq1, saqlash joyini ko`rsatamiz, **Ok** tugmasini bosamiz va shunda 4 – rasmdagi oyna ochiladi.



4 – rasm

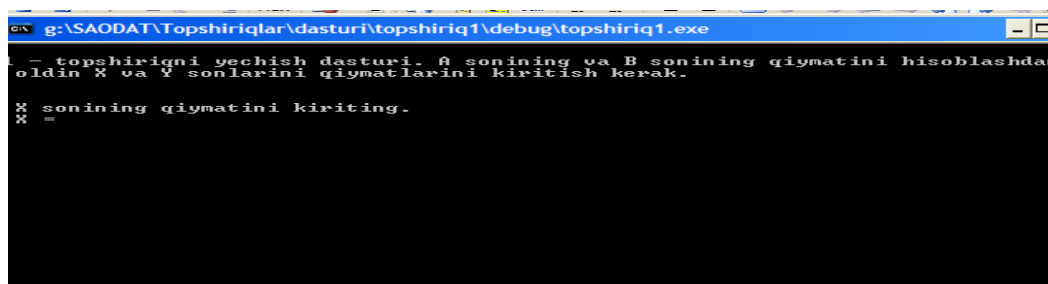
Bu oynada **Next >** tugmasini bosamiz va 5 – rasmdagi oyna ochiladi. Ochilgan oynada projektning hususiyatlari ko`rsatiladi, biz hech narsani o`zgartirmay **Finish** tugmasini bosamiz.

Algoritmini yozib bo'lgandan keyin shu algoritm asosida dasturiy qismini yozamiz (7 – rasm).



7 – rasm

Dasturiy qisimni yozib bo'lgandan so'ng dasturni kompilyatsiya qilish va ishga tushirish kerak. Kompilyatsiya qilish va ishga tushirish uchun asosiy oynadagi ishga tushirish tugmasi bosiladi va dastur ishga tushadi. Ekranda qora oyna ochiladi (8 – rasm) va foydalanuvchi o'rnida o'zim dasturni ishlashini tekshiraman. Ya'ni noma'lum sonlarning qiymatini klaviatura yordamida kiritaman. Bunda ekranda so'ralgan noma'lum (X) ning qiymatini klaviatura yordamida kiritaman va Enter tugmasini bosaman va keyingi noma'lum (Y) ning qiymatini va klaviatura yordamida kiritaman va Enter tugmasini bosaman. Noma'lum sonlar kiritilgandan so'ng ekranda A va B sonining natijalari ekranda chiqadi (9 – rasm) va bu natijalarni to'g'riligini tekshirib ko'raman (matematik usul bilan). Dastur yopilishi (tugashi) uchun ixtiyoriy sonni klaviatura yordamida kiritaman va klaviaturadagi Enter tugmasini bosaman. Shunda dastur o'z ishini tugatadi.



8 – rasm

```

g:\SAODAT\Topshiriqlar\dasturi\topshiriq1\debug\topshiriq1.exe
- topshiriqni yechish dasturi. A sonining va B sonining qiymatini hisoblashdan
oldin X va Y sonlarini qiymatlarini kiritish kerak.

X sonining qiymatini kiriting.
X = 2

Y sonining qiymatini kiriting.
Y = 7

A soning qiymati.
A = 12

B soning qiymati.
B = 0.583333

```

9 – rasm

1-amaliy mashg'ulot topshiriqlari:

- | | | |
|-----|---|-----------------------------------|
| 1. | $a = \frac{xy}{14} + 11$ | $b = \frac{y}{a}$ |
| 2. | $a = z^2 + \sqrt{z^{-1} + y^4}$ | $b = \frac{a}{z^2 + z}$ |
| 3. | $x + y + z + \frac{1}{2}$ | $b = a z^3 + x$ |
| 4. | $a = y^2 + \frac{1}{x}$ | $b = x^3 + 3a$ |
| 5. | $a = \sqrt[4]{x + y}$ | $b = \frac{a^3}{2x + 3y}$ |
| 6. | $a = \frac{z^4}{x^3} + 3x^{\frac{-3}{4}}$ | $b = \sqrt{a} + 2x$ |
| 7. | $a = \frac{x^4}{1 + x - y}$ | $b = \frac{a^{3/4+2x}}{y}$ |
| 8. | $a = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!}$ | $b = \frac{x}{a + 1}$ |
| 9. | $a = (1 + x^4 + 2x^3 - x)^{-1/2}$ | $b = (a + x) \frac{a + 2}{x - 1}$ |
| 10. | $a = \frac{1 + \frac{1}{x + y}}{2 - \frac{x}{2 - y}}$ | $b = a^{3/4} (x + y)$ |
| 11. | $a = (x + \sqrt{2x - y})^{3/4} - 2$ | $b = a x^3$ |
| 12. | $a = \sqrt{2^{x-y} + 3^{1/y}}$ | $b = \frac{(a + x)^{3/2}}{2!}$ |
| 13. | $a = x - \frac{x^3}{3!} + \frac{x^2}{2!}$ | $b = a^3 + x^3$ |
| 14. | $a = \left(\frac{x}{2} + \frac{8 - y}{3} \right)^2$ | $b = a - x^2$ |

15.	$a = \left(\frac{38}{x} + \frac{y}{9}\right)^2$	$b = a^2 + x^2$
16.	$a = 4\sqrt{x^2 - y^2}$	$b = \frac{x - y}{a}$
17.	$a = \frac{ -x + y }{14} + x^2$	$b = a^2 - 4x$
18.	$a = \frac{11x + 13y}{6} + 3$	$b = \frac{2}{5}a$
19.	$a = (x - y)^3 - 3x + y$	$b = a^{2/3}$
20.	$a = 12^{x+y} - \sqrt{x}$	$b = 3 \times a$
21.	$a = \left(x - \frac{2x}{y}\right) 14y$	$b = x^a$
22.	$a = x^x - 118 + y$	$b = \frac{\sqrt{a}}{\sqrt{x}}$
23.	$a = \frac{9y - x}{\sqrt{x}} + 2y$	$b = a^{\sqrt{x}}$
24.	$a = \left(\frac{3xy}{5}\right)^2 + x$	$b = 3\frac{2}{xy} + a$
25.	$a = \frac{x^y}{18y} + 4$	$b = \frac{12^a}{x}$
26.	$a = \frac{32 - x}{7 + x} - 11x^2$	$b = \frac{3}{a} + y^2$
27.	$a = \frac{4y + x^2}{y^2 - 5} + 7$	$b = \frac{18}{x} + \sqrt{a}$
28.	$a = 3x - \sqrt{8y} + \frac{3}{y}$	$b = \frac{7}{3}a$
29.	$a = \left(\frac{5}{x} - \frac{y}{7}\right) + 3x$	$b = \frac{a}{x + y}$
30.	$a = \frac{18y}{x + y} + 3x$	$b = a - \frac{8}{x}$

1-AMALIY MASHG'ULOTGA TESTLAR:

1. Nuqtalar o'rnini to'ldiring:

..... - foydalanuvchi tomonidan berilgan talablar spetsifikatsiya

deb ataluvchi maxsus hujjatlarda rasmiylashtiriladi.

A. Malumotlarni tavsiflash.

B. Masalani berilish hujjati.

C. Maqsadlarni aniqlash.

2. Maxsulotning maqsadi tarkibiga qaysi jarayonlar kiradi?

A. loyihalash grafigi, narxi, testlash.

B. sifati, narxi, xajmi.

C. standartga javob berishi xamda narxi.

3. Dastur samaradorligi xamda dastur xajmining o'zaro munosabati qanday ?

A. o'zgarishsiz.

B. samaradorligi oshsa – xajmi kamayadi.

C. samaradorligi oshsa –xajmi oshadi.

4. Dasturiy maxsulot maqsadining qaysi guruhida vaqt me'zoni xisobga olinadi?

A. Kuzatib borish.

B. Xavfsizlik.

C. Dasturiy maxsulotning tan narxi.

5. Dasturiy loyixalar necha guruhga bo'linadi?

A. 2

B. 3

C. 5

6. “Dasturiy maxsulot loyixalash uni tashkil qilishda asosiy bosqichlaridan biri hisoblanadi. Yaxshi loyixalangan dasturni yomon yozish mumkin. Ammo yomon loyixalangan dasturni yaxshi yozish mumkin emas”. –degan tushuncha kim tomonidan aytilgan?

A. Ivanova

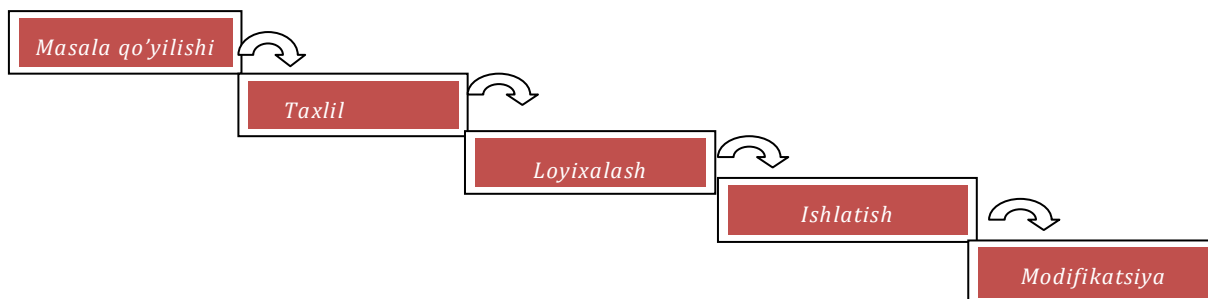
B. Van Tassil

C. G. Mayers

2. Oraliq nazorat modeli;
3. Spiralli model hamda,
4. Umumlashtirilgan model.

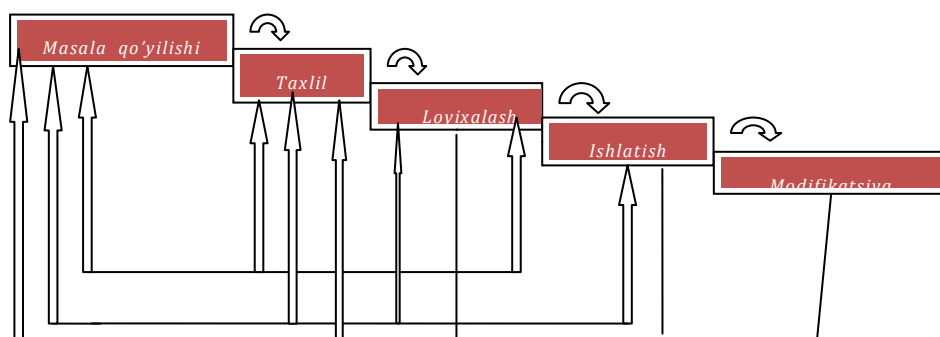
Dasturiy ta'minot yashash siklining 1-modeli kaskadli model hisoblanadi.

Kaskadli model bo'yicha har bir keyingi bosqichga o'tish faqatgina oldingi bosqich to'la tugallangandan so'ng boshlanadi. Ushbu modelni quyida rasm orqali batafsilroq tushuntiramiz:



Dasturiy ta'minot yashash siklining 2-modeli oraliq nazorat modeli hisoblanadi.

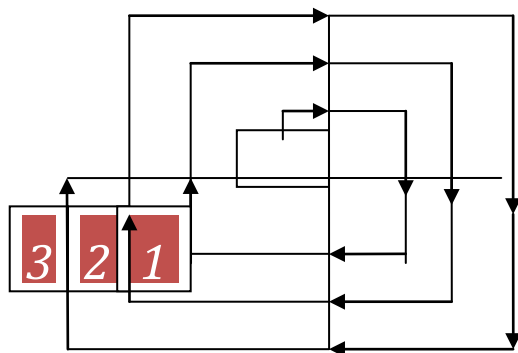
Oraliq nazorat modeli asosida iteratsiya usuli qo'yilgan, ya'ni har bir bosqichdan ixtiyoriy bosqichga qaytish mumkin. Ushbu modelni quyida rasm orqali batafsilroq tushuntiramiz:



Dasturiy ta'minot yashash siklining 3-modeli spiralli model hisoblanadi.

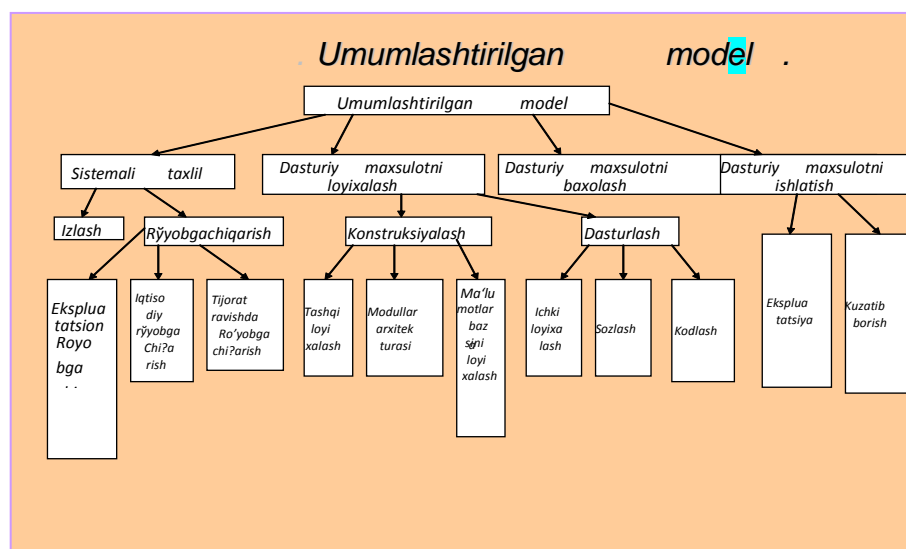
Spiralli model asosida prototipirovaniye usuli qo'llaniladi.

Prototip- dasturiy mahsulot bo'lib, u ishlab chiqarilayotgan dasturiy ta'minotning maxsus funksiyalari va tashqi interfeysini ishlatishga yordam beradi. Ushbu modelni quyida rasm orqali batafsilroq tushuntiramiz:



Dasturiy ta'minot yashash siklining 4-modeli umumlashtirilgan model hisoblanadi.

Umumlashtirilgan modelda yuqorida keltirilganlar bilan birgalikda berilgan masalaning maqsadi aniqlanadi. Juda ko'p hollarda bu bosqich ikki xil yo'nalishda : izlanish ,ro'yobga chiqarish yo'nalishlarida olib boriladi.



Umumlashtirilgan model sxemasi quyoda tushuntiramiz:

1.Dasturiy maxsulot yashash siklining birinchi bosqichi **sistemali taxlil** deyiladi. Bu bosqichda dasturiy maxsulotga bo'lgan talab uning asosiy funksional xarakteristikalarini aniqlaydi. Maxsulotni ishlab chiqarish uchun ketishi mumkin bo'lgan mablag' hisoblanadi, maxsulotni ishlatilish samaradorligi aniqlanadi. Bu bosqichda yuqorida keltirilganlar bilan birgalikda berilgan masalaning maqsadi aniqlanadi. Juda ko'p hollarda bu bosqich ikki xil yo'nalishda: izlanish ,ro'yobga chiqarish yo'nalishlarida olib boriladi.

1.1. **Izlanish**- bosqichi, ishlab chiqarish rahbari dasturiy maxsulotga bo'lgan talablarni aniqlab ularning ketma-ketligini tartibga soladi.

1.2. **Ro'yobga chiqarish** - bosqichi, izlanish bosqichining texnik qismi hisoblanadi. Bu bo'limda hosil qilingan mahsulotni baholash imkoniyati tekshiriladi.

Ekspluatatsion ro'yobga chiqarish-ya'ni maxsulotni amaliy ravishda ishlatish qulay bo'ladimi yoki yo'qmi degan savollarga javob beradi. Iqtisodiy ro'yobga chiqarish bosqichida ishlab chiqarilayotgan maxsulotning tannarxi foydalanuvchiga qulay bo'ladimi va iqtisodiy jihatdan samarali maxsulot bo'ladimi yoki yo'qmi degan savollarga javob izlaydi.

2. Dasturiy maxsulotni loyihalash. **Loyiha** so'zi lug'atda reja asosida biror bir ko'rinishga keltirish kabi aniqlangan.

2.1. **Konstruksiyalash** bosqichi ma'lum bir maqsadlar aniqlangandan so'ng boshlanadi.

2.1.1. Tashqi loyihaning bu bosqichida tashqi loyihalashning foydalanuvchi bilan muloqoti aniqlanadi.

2.1.2. Modulning arxitekturasini aniqlash jarayonida butun loyixalarni tashkil qiluvchi dasturlar, modullar, komponentalar va ular orasidagi bog'liqlik aniqlanadi.

2.1.3. Bu bosqich natijasida ma'lumotlar bazasi loyixalanadi

2.2. **Dasturlash**. Bu bosqich konstruksiyalash bilan birgalikda olib boriladi.

2.2.1. **Ichki loyixalash** -bunda dasturiy mahsulotni tashkil qiluvchi modullarni ichki loyixasi xosil qilinadi.

2.2.2. **Dasturni sozlash**.

2.2.3. **Dasturni kodlash**, ya'ni algoritmni tanlab olingan dasturlash yoki mashina tilida yozish.

3. Dasturiy maxsulotni baxolash va sinash. Bunda dasturiy maxsulot sistemali sinovdan o'tkaziladi. Bosqich dasturiy ta'minotni tashkil qiluvchi barcha modullar yozilib, sozlab bo'lgandan so'ng boshlanadi.

4. Dasturiy maxsulotni ishlatish. Bu bosqich dasturiy maxsulot taqsimlash sistemasiga o'tkazilganidan so'ng boshlanadi.

4.1. **Ekspluatatsiya** bu dasturni belgilangan vaqtda ishlatish.

4.2. Dasturiy maxsulotni belgilangan vaqtda kuzatib borish.

Nazorat savollari:

- 1.Kaskadli model haqida ma'lumot bering.
- 2.Spiralli model ta'rifi?
- 3.Umumlashtirilgan model tarifi?
- 4.Oraliq nazorat model ta'rifi?
5. Izlanish bosqichida qanday amallar bajariladi?
6. Ro'yobga chiqarish bosqichida qanday amallar bajariladi?
7. Loyiha so'zining lug'aviy ma'nosi?
- 8 . Dasturiy maxsulotni baxolash va sinash haqida ma'lumot bering.

Tayanch iboralar: prototip, sistemali taxlil ,izlanish, ro'yobga chiqarish,ekspluatatsion ro'yobga chiqarish, ichki loyihalash, ekspluatatsiya.

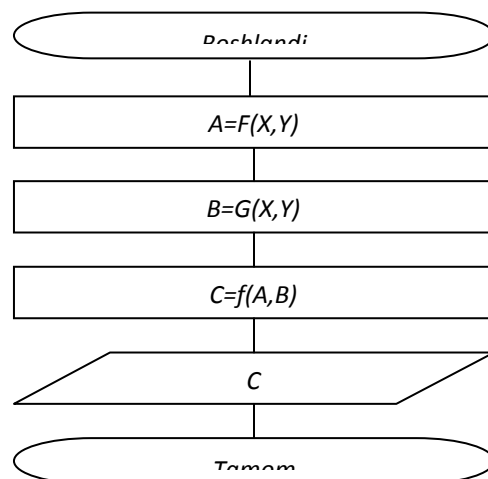
2 – topshiriq tavsifi.

2 – topshiriqda dasturchi ekranga chiqarish, eng odiy arifmetik amallarni bajarish, murakkab matematik funksiyalarni hisoblash va algaritm ketma-ketligini o`rganadi.

2– topshiriqni yechish algoritmini ko`rib chiqamiz.

Har bir misolda ma'lum murakkab matematik funksiyalarni bajarib 2 sonning qiymatini hisoblash kerak. Bu sonlarni hisoblagandan avval hisoblash algaritmini tuzish kerak so'ngra murakkab arifmetik ketma – ketlikni hisoblash kerak. Hisoblash algaritmining arifmetik amallari qavslar yordamida guruhlash orqali amalga oshiriladi. Guruhlangan ariifmetik amallar bajarilganidan so`ng ketma – ketlik qiymatiga ega bo`lamiz va bu qiymatni ekranga chop ettirish kerak- ya'ni foydalanuvchi topshiriq natijalarini ko`rishi bilan dastur tugaydi. Endi bu ketma - ketliklarni blok sxema ko`rinishida yozamiz. Blok sxemada hisob-kitob algaritmi ko`rsatilmaydi .

Blok sxema:



Misol uchun 2 – topshiriqning 1 – misolini yechib ko`ramiz.

Biz bu misolda yangi projekt yaratib dasturlashni boshlaymiz. Yangi projektни yaratish fayl menyusi yordamida yoki bosh sahifadan yangi projekt oynasini ochish orqali amalga oshiriladi. Fayl menyusi orqali ochish **File** –>**New** –>**Project** orqali amalga oshiriladi. Yangi projekt oynasida dasturlash tilini, projekt turini tanlash, projekt nomini va qayerda saqlanishini ko`rsatishi kerak. Bunda dasturlash tilining Visual C++, projektни turini Win32 Console Application, nomini Topshiriq1, saqlash joyini ko`rsatamiz, **Ok** tugmasini bosamiz . Bu oynada **Next** > tugmasini bosamiz va oyna ochiladi. Ochilgan oynada projektning xususiyatlari ko`rsatiladi biz esa xech narsani o`zgartirmay Finish tugmasini bosamiz. Bunda VS2005 ning asosiy oynasi va dasturiy qismi (kodi) saqlanadigan fayl ochilgan saxifa ham ochiladi (6 – rasm). Bu sahifaga misolning dasturiy qismi yoziladi va shu faylda saqlanadi. Dasturlashga o`tishdan avval unig algoritmini tuzish kerak. Dasturlash algoritmi yuqorida ko`rsatilgandek bo`ladi. 1 – misolning dasturlash algoritmini ko`rib chiqamiz.

$$1 - \text{misol. } \left(\frac{2-b}{b-1} + 2 \frac{a-1}{a-2} \right) \div \left(b \frac{a-1}{b-1} + a \frac{2-b}{a-2} \right); A = \sqrt{2} + 0,8; B = \sqrt{2} - 0,2;$$

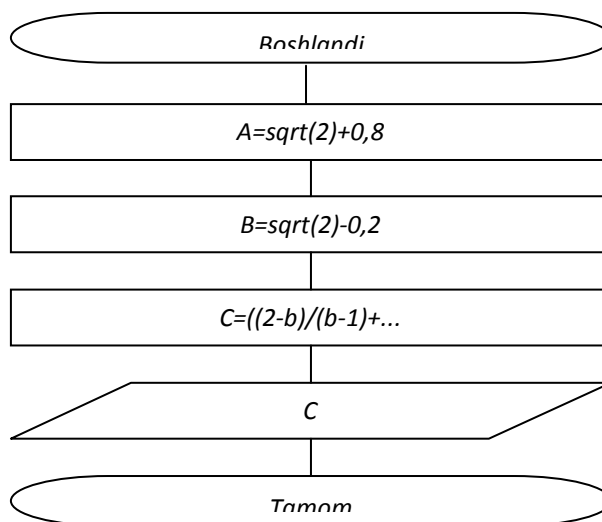
Arifmetik amallarni bajarib A va B sonining qiymatini hisoblash kerak. Bu sonlarni hisoblashda murakab matematik funksiya (ildiz osti) ni hisoblash kerak buning uchun matematik murakkab funksiyalar to`plami (math.h) kerak. Ular yordamida murakkab funksiyalarni hisoblash mumkin. A va B sonning qiymatini hisoblagandan keyin murakkab arifmetik ketma-ketlikni hisoblash kerak. Bunda murakkab ketma – ketlikni dasturlash tiliga moslab hisoblash algoritmini yozamiz.

Murakkab ketma-ketlikni hisoblash algoritmi:

$$C = \left(\frac{(2-b)}{(b-1)} + 2 \frac{(a-1)}{(a-2)} \right) \div \left(b \frac{(a-1)}{(b-1)} + a \frac{(2-b)}{(a-2)} \right)$$

Qavslar tagidagi ko`rsatilgan sonli ketma – ketlik bo`yicha hisoblashni amalga oshiramiz. Hisoblash ketma-ketligi amalga oshirilgandan keyin qiymatni ekranga chop ettirish kerak, ya'ni foydalanuvchi topshiriq natijalarini ko`rsin va shu bilan dastur tugaydi. Endi bu ketma ketliklarni blok sxema ko`rinishida yozamiz:

Blok sxema:



Algoritmini yozib bo`lgandan keyin shu algoritm asosida dasturiy qismini yozamiz.

Dasturiy qisimni yozib bo`lgandan keyin dasturni kompilyatsiya qilish va ishga tushirish kerak. Kompilyatsiya va ishga tushirish asosiy oynadagi ishga tushirish tugmasi orqali amalgam oshiriladi va dastur ishga tushadi. Ekranda qora oyna ochiladi (8 – rasm), C sonning natijasi ekranda chiqadi va bu natijani to`g`riligini tekshirib ko`ramiz (matematik usul bilan). Dastur yopilishi (tugashi) uchun ixtiyoriy sonni klaviatura yordamida kiritamiz va klaviaturadagi Enter tugmasini bosaman. Shunda dastur o`z ishini tugatadi.

```

g:\SAODAT\Topshiriqlar\dasturi\topshiriq2\debug\topshiriq2.exe
2 - topshiriqni yechish dasturi.
C soning qiymati.
c = 1

```

8 – rasm

2-amaliy mashg'ulotga topshiriqlar:

1. $\left(\frac{2-b}{b-1} \div 2 \frac{a-1}{a-2}\right) \div \left(b \frac{a-1}{b-1} + a \frac{2-b}{a-2}\right)$ $a = \sqrt{2} + 0.8$ $b = \sqrt{2} - 0.2$
2. $\frac{\sqrt{x}+1}{x\sqrt{x}+x+\sqrt{x}} \div \frac{1}{x^2+\sqrt{x}}$ $x=11$
3. $\left(\frac{(a+b)^{3/4}+c^{1/2}}{a^{3/4}-b^{-1/2}}\right)^{3/4} \div \left(\frac{b^3-c^4}{(a+b^{1/4}b^3)}\right)$ $a=2$ $b=3$
4. $\frac{2x^{-1/6}}{x^{1/4}-3x^{-1/4}} - \frac{x^{2/3}}{x^{1/3}-x^{2/4}} - \frac{x+1}{x^2-4x+3}$ $x=5$

5. $t \frac{1 + \frac{1}{\sqrt{t+1}}}{2 - \sqrt{t+4}} - \sqrt{t+4} - \frac{4}{\sqrt{t+4}}$ $t=10$
6. $\frac{2^{1.4} + 27y^{1/4}}{\sqrt{2} + 3\sqrt[4]{y}} + 3\sqrt[4]{32y^2} - 2$ $y=4$
7. $\frac{x-1}{x + \sqrt{x+1}} \div \frac{\sqrt{x+1}}{x-1} + \frac{2}{x}$ $x=12$
8. $\left(\left(\frac{1}{\sqrt{a} + \sqrt{a+1}} + \frac{1}{\sqrt{a} - \sqrt{a+1}} \right) \right) \div \left(1 + \sqrt{\frac{a+1}{a-1}} \right)$ $a=2$
9. $\frac{x-1}{x^{3/4} + \sqrt{x}} \cdot \frac{\sqrt{x} + \sqrt[4]{x}}{\sqrt{x+1}} \cdot \sqrt[4]{x+1}$ $x=17$
10. $\sqrt[4]{6x(5 + 4\sqrt{6})} \cdot \sqrt{3\sqrt{2x} - \sqrt{3x}}$ $x = \frac{11}{12}$
11. $\sqrt[6]{4x(11 + 4\sqrt{6})} \cdot \sqrt{2\sqrt{3x} - 4\sqrt{2x}}$ $x=2$
12. $\frac{x + \sqrt{2-x} * \sqrt{1-x\sqrt{2-x}}}{\sqrt{1-x}}$ $x=0.12$
13. $\sqrt{\frac{\sqrt{2}}{a} + \frac{a}{\sqrt{2}} + 2} - \frac{a^2\sqrt{2x} - 2\sqrt{a}}{a\sqrt{2a} - \sqrt[4]{8a^4}}$ $a = \frac{7}{16}$
14. $\frac{(m^2 - n^2) * (n + m^{-1})^{n-m}}{(n^2 - m^2) * (m - n^{-1})^{m-n}}$ $m=3.1$ $n=2.2$
15. $\frac{(2p - q)^2 + 2p^2 - 3pq}{2p^{-1} + q^2} \div \frac{4p^2 - 3pq}{2 + pq^2}$ $p=0.78$ $q = \frac{7}{25}$
16. $\frac{\sqrt[4]{7\sqrt[3]{54} + 15\sqrt[3]{128}}}{\sqrt[3]{4\sqrt[4]{32} + \sqrt[3]{9\sqrt[4]{162}}}}$
17. $\sqrt[4]{32\sqrt[3]{4}} + \sqrt[4]{64\sqrt[3]{0.5}} - 3\sqrt[3]{2\sqrt[4]{2}}$
18. $\frac{3a^2 + 2ax - x^2}{(3x + a)(a + x)} - 2 + 10 * \frac{ax - 3x^2}{a - 9x^2}$ $a = \sqrt{2}$ $x=2$

19. $\left(x^2 + 2x - \frac{11x-2}{3x+1}\right) \div \left(x+1 - \frac{2x^2+x+2}{3x+1}\right)$ $x=7.(3)$
20. $\left(\frac{a\sqrt{a}+b\sqrt{b}}{\sqrt{a}+\sqrt{b}} - \sqrt{ab}\right) \div \left(\frac{\sqrt{a}+\sqrt{b}}{a-b}\right)^2$ $a=28$ $b=17$
22. $\frac{\sqrt{x}+1}{x\sqrt{x}+x+\sqrt{x}} \div \frac{1}{x^2+\sqrt{x}}$ $x=13$
23. $\frac{2x^{-1/6}}{x^{1/4}-3x^{-1/4}} - \frac{x^{2/3}}{x^{1/3}-x^{2/4}} - \frac{x+1}{x^2-4x+3}$ $x=2$
24. $\frac{2^{1.4}+27y^{1/4}}{\sqrt{2}+3\sqrt[4]{y}} + 3\sqrt[4]{32y^2} - 2$ $y=1$
25. $\frac{x-1}{x^{3/4}+\sqrt{x}} \cdot \frac{\sqrt{x}+\sqrt[4]{x}}{\sqrt{x}+1} \cdot \sqrt[4]{x}+1$ $x=1$
26. $\sqrt[6]{4x(11+4\sqrt{6})} \cdot \sqrt{2\sqrt{3x}-4\sqrt{2x}}$ $x=4$
27. $\frac{(m^2-n^2) \cdot (n+m^{-1})^{n-m}}{(n^2-m^2) \cdot (m-n^{-1})^{m-n}}$ $m=3$ $n=2$
28. $\frac{\sqrt[4]{7\sqrt[3]{54}}+15\sqrt[3]{128}}{\sqrt[3]{4\sqrt[4]{32}}+\sqrt[3]{9\sqrt[4]{162}}}$
29. $\frac{3a^2+2ax-x^2}{(3x+a)(a+x)} - 2 + 10 \cdot \frac{ax-3x^2}{a-9x^2}$ $a=\sqrt{2}$ $x=1$
30. $\left(\frac{a\sqrt{a}+b\sqrt{b}}{\sqrt{a}+\sqrt{b}} - \sqrt{ab}\right) \div \left(\frac{\sqrt{a}+\sqrt{b}}{a-b}\right)^2$ $a=2$ $b=11$
-

2-AMALIY MASHG'ULOTGA TESTLAR:

1. Dasturiy taminot yashash siklining nechta modeli mavjud?
A. 4 B. 2 C. 6
2. Dasturiy taminot yashash siklining qaysi modelida keyingi bosqichga o'tish faqatgina oldingi bosqich to'la tugallangandan so'ng boshlanadi?
A. Spiralli model. B. Kaskadli model C. Umumlashtirilgan model.
3. "iteratsiya usuli" qanday ma'noni anglatadi?
A. har bir bosqichdan faqat boshlang'ich bosqichga qaytish mumkin
B. har bir bosqichdan xoxlagan bosqichga qaytish mumkin.
C. har bir bosqichdan faqat oxirgi bosqichga qaytish mumkin
4. Prototip nima?
A. Dasturiy maxsulot B. Dasturiy maxsulot loyixasi C. Ma'lumotlar bazasi
5. Qaysi modelning maqsadi -izlanish xamda ro'yobga chiqarish yo'nalishlarida olib boriladi?
A. Spiralli model
B. Kaskadli model
C. Umumlashtirilgan model
6. Qaysi bosqichda ishlab chiqarish raxbari dasturiy maxsulotga bo'lgan talablarni aniqlab ularning ketma-ketligini tartibga soladi?
A. Izlanish bosqichi
B. Ro'yobga chiqarish bosqichi
C. Konstruksiyalash bosqichi
7. Qaysi bosqichda bo'limda xosil qilingan mahsulotni baholash imkoniyati tekshiriladi.
A. Konstruksiyalash bosqichi
B. Izlanish bosqichi
C. Ro'yobga chiqarish bosqichi
8. Dastur ekspluatatsiyasi nimani anglatadi?
A. Dasturni sinash.
B. Dasturni belgilangan vaqtda ishlatish.

C.Dasturni kodlash.

Javoblari: 1-A 2-B 3-B 4-A 5- C 6-A 7- C 8-B

Dastur kodi BMIning ilova qismi 2-ilova bandida keltirilgan.

3.3. FOYDALANUVCHI BILAN MULOQOT

REJA:

1. DT ning yo'nalishlari;
2. Dasturlash texnologiyasining mahsuloti;
3. Foydalanuvchining to'liq qayta ishlangan interfeysi;
4. CASE texnologiyasi.

Ishdan maqsad: Dasturiy vosita xamda foydalanuvchi o'rtasidagi muloqotni tashkillashtirish.

Nazariy qism

Dasturlash texnologiyasi dasturiy ta'minotning rivojlanishida turli xil bosqich turli ahamiyat kasb etadi. Kompyuter imkoniyatlarining oshishi, rivojlanish vositasi va dasturlash metodologiyasining (uslubining) o'sishi kompyuterlarda murakkab masalalarni hal qiluvchi dasturlash texnologiyalariga bo'lgan e'tiborning oshishiga olib keldi. Kompyuter qiymatining keskin arzonlashuvi va ayniqsa, kompyuter

tashuvchilarda ma'lumotlarning saqlanish qiymati shunga olib keldiki, inson faoliyatining hamma sohasida amaliy kompyuterlarni keng qo'llash, dasturlash texnologiyasining muhim ko'zlangan maqsadini o'zgartirdi. Tashkillashtirish yetarlicha puxta, sifatli DV tushunchasiga ega, shu bilan birga DV foydalanuvchi ishi uchun qulaydir. Kompyuter tarmoqlarining qo'llanilishi tezkor samarali rivojlanishi ma'lum masofaga mo'ljallangan ma'lumotga (axborotga) kirish va elektron orqali odamlar orasida ma'lumotlarni almashtirishga olib keldi. Kompyuter texnikasi alohida masalalar yechimi vositasida haqiqiy (real) va o'ylangan (ideal) dunyo informatsion modellash vositasiga, oddiygina odamlarni qiziqtirgan savollariga javob berish vositasi tobora o'zgarib boradi. Insoniyat jamiyatida chuqur va to'liq axborotlashtirish (malumotlar) bosqichi boshlanyapti. Bularning hammasi texnologiyani dasturlash oldida yangi va yetarlicha qiyin muammolarga olib kelyapti:

70-yillarda to'satdan kompyuter tizimlarining malumotlarini saqlashga qiziqish ortib ketdi. Dasturlash texnologiyasi samarali rivojlanishining quyidagi yo'nalishlari boshlandi:

- dalillar bilan isbotlash va dastur taminotining tuzilishi va ishlov berishga mos kelmaslikni keng joriy qilish;

- rivojlanishning mavhum ma'lumotlarining turi va dasturlashning moduli (hususan, g'oyalar paydo bo'lishining o'ziga hos taqsimlanishi va ma'lumot strukturalarini berkituvchi modullardan foydalanish);

- DV ning ishonchliligini tekshirish va safarbar qilish muammosini ta'minlash;

- DV jamoaviy ishlov berishning metodini (uslubini) boshqarishni barpo qilish;

- Dasturlash texnologiyasiga yordamlashish vositasi instrumental (asbob yasaydigan) dasturlarning yuzaga kelishi (dastur instrumentlari (asboblari));

80-yillar inson faoliyatining barcha sohalarida maxsus kompyuterlarni keng suratda qo'llash, shunday kompyuterlarni ko'proq yaratish va DV dan foydalanuvchi turli xil sohadagi kishilar bilan xarakterlanadi. Bu interfeysdan foydalanishning shiddatli rivojlanishi va DV konseptsiyasi sifatining aniq yaratilishiga olib keldi.

Dasturlash texnologiyasining talabini hisobga oluvchi dasturlash tili ko'ridi. Texnologiya jarayonining nusxasi qizg'in harakatda boshlandi va hammasidan avval bu jarayonda hujjatlar paydo bo'ldi. DV ga ishlov berish munosabati ilg'or pozitsiya (mavqe) obyektidan chiqishdir. DV ni uzatish bilan birga har xil instrumentallar orasida ishlash (ishlov berish) yuzaga keldi. Kontseptsiya kompyuterlarining tarmoqlari rivojlandi;

90-yillar shunisi bilan ahamiyatliki, unda jamiyat orasida xalqaro kompyuter tarmoqlari keng jalb etildi. Shaxsiy kompyuterlar unga xuddi terminal sifatida ulanmoqdi. Bu kompyuter tarmoqlari informatsiyasiga bir qator muammolarni tartibga solishga kirishishni yuzaga keltirdi (xuddi texnologik, shu kabi yuridik va etik (odob, axloq) xarakter asosida). Kompyuter informatsiyasining muammolarini himoya qilish keskin boshladi va axborot tarmog'iga o'tdi. DV ga ishlov berish va kompyuter texnologiyasi qizg'in rivojlana boshladi (CASE texnologiyasi) va dasturlash spetsifikatsiyasining formal metodlari (uslublari) bilan birga bog'langan. Butun axborotlashtirishni kompyuterlashtiruvchi bosqich (davr) boshlandi.

Dasturlash texnologiyasining mahsuloti

Dasturning mazmuni talab qilinuvchi funksiyani to'ldiruvchi, DV dasturlash texnologiyasining mahsuloti bo'ladi. Bu yerda «dastur» («programma») da tez-tez to'g'ri dasturni ya'ni, xatolarni saqlaydigan dastur tushuniladi. Biroq, dasturda dasturlovchilar orasida xatolar tushunchasi ko'p marotaba talqin etilgan. Agarda u foydalanuvchilarni yaxshi tushinib, uni to'ldirsa, dasturda xato bo'lishi hisobga olinadi. Bu dastur hujjatlar asosida qo'llaniladi va «fahmlab kutish» foydalanuvchilardan talab etiladi. Demak, dasturdagi xatoni tushunish rasman muhim (kerakli) emas. DV da dastur va hujjatlar o'zaro shunday bog'liqlik, ular ayrim yaxlitlikni tashkil etadi. Shuning uchun dasturda bo'lmagan xatolar haqida emas, balki, butun DV haqida gapirilsa to'g'ri bo'ladi: foydalanuvchilarni oqilona kutishini, agar u to'ldirilmasa, DV da xato bor deb hisoblaymiz. Ayrim hollarda, DV dagi xatolar turi DV dasturi orasida kelishmovchilik kelib chiqadi va hujjatlar ularning qo'llanishi bo'yicha ishlatiladi. Dastur o'zining funksional spetsifikatsiyasiga to'g'ri kelmasa, alohida tushunchali ayrim hodisa xatolari paydo bo'ladi (tariflanadi, etap

bo'yicha ishlab chiqiladi, dasturlashda oldin o'rtacha xatolik yuz beradi). Ko'rsatilgan ishda bunday xato dastur defekti deb ataladi. Lekin, bunday turli xatolarni ajratilishi, alohida tushunchalar, xatolarning sababi dastur emas, funksional spetsifikaning o'zi bo'lishi mumkin degan fikrga olib keladi.

Foydalanuvchining to'liq qayta ishlangan interfeysi. Yangi interfeys sharofati bilan Windows da dasturlarni ishga tushirish, hujjatlarni ochish va saqlash, disklar va arxivlar bilan ishlash ancha oson va yengil. Quyida foydalanuvchining yangi interfeysining asosiy hususiyatlarini ko'rib chiqamiz:

- Ishga tushurish tugmachasining menyusi (Start Menu, ruscha versiyada - Bosh menyu) siz so'nggi paytlarda ishlagan dasturlar va hujjatlarga, boshqaruv paneli (Control Panel)ga, printerlarga va tizim utilitlardan foydalanish, ularga yo'llashning osonlashishi ta'minlanadi;

- Bir dasturdan boshqasiga o'tishga soddalashtirilgani;

- Oldingi versiyalardagi fayllar dispecheri (G'ile manager)ni almashtirib kelgan Windows o'tkazuvchisi (Windows Explorer)ning kuchli dastur ekanligi;

- Serverlar ko'rib chiqishni va tarmoq fayllariga xuddi lokal qattiq diskdagidek oson ishlov berish imkoini beruvchi tarmoq o'rami nishoni (Network Neighborhood)ning borligi;

- Tez-tez ishlatiladigan dasturlar, panellar va hujjatlar uchun yorliqlar yaratish imkoni;

- Obyektlarning tizimini sozlashni yengillashtiruvchi hususiyatlarining ro'yxati;

- Hujjatga u yaratilgan ilovani ishga tushurmay nazar tashlash imkonini beruvchi «tez ko'rib chiqish» vositalari to'plami;

- Sizning kompyuteringizda bo'lganidek, xuddi shuningdek tarmoq serveridan ham ixtiyoriy axborotni tanlashga yordam beruvchi kuchli izlash dasturi;

- Siz tasodifan kerakli bo'lgan ma'lumotni chiqarib tashlaganingizda sizga uni tiklashda qo'l keladigan savatcha (Recycle Bin);

- Printerlar va shiriftlar bilan ishlash uchun oddiy asboblar;

- Tizimni o'rnatish va sozlashda foydalanuvchining kuzatuvchi «ustalar» jamoasi (Wizards).

- Unda kontekstli izlash imkoni paydo bo'lgan takomillashtirilgan yordam tizimining mavjudligi.

Yangi tizimga o'tish Windowsga Windowsning oldingi versiyalarining qobiqlari bo'lgan dasturlar dispecheri (Program Manager) va fayllar dispecheri ham kirganliklari tufayli ham osonlashadi. Shunisi ham borki, ulardan kimdir uzoq vaqt foydalanishi (agar umuman foydalansa) ehtimoldan xoli.

CASE texnologiyasi

Case - texnologiyalari turli mutaxassislar : tizimli taxlilchilar, loyixachilar va dasturchilar ishtirok etadigan , odatda jamoaviy safarbarlikni talab etuvchi murakkab axborot tizimlarini yaratishda qo'llaniladi.

Case texnologiyalar deganda AT predmet soxasiningm taxlili uslubiyati, loyixalashtirilishi, dasturlashtirilishi va foydalanilishini o'z ichiga olgan axborot tizimlarini ishlab chiqarishni avtomatlashtirish vositalari yig'indisi tushuniladi.

Case - texnologiyalarining asbobsozlik vositalari tizim siklining barcha hayotiy bosqichlarida (taxlil va loyixalashdan to tatbiq etilguncha) qo'llaniladi, ular yuzaga kelgan vazifalarni xal etishni ancha soddalashtiradi.

Case - texnologiyalardan foydalanishda tizim ishlanmasini tayyorlovchi kishi detallarga chalg'imag yuqori darajada loyixalashtirish bilan shug'ullanadi.

Bu boshidanoq xato qilmaslik, ancha mukammal dasturiy maxsulotlar olish imkonini beradi. Bu texnologiyalar kompaniyalar uchun rejalashtirish, moliyalashtirish, ta'lim kabi vazifalarni juda yaxshi xal etishga yordam beradi. Shunday qilib, Case -texnologiyalari kompaniyalarga u yoki bu loyixani oqilona amalga oshirish yoki biznes umumiy samaradorligini oshirishda tub o'zgartirishlarni amalga oshirish imkonini beradi.

Xozirgi paytda Case - texnologiyalari-o'zida yuzlab kompaniyalarni birlashtirgan, informatikaning eng tezkor rivojlanayotgan sohalaridan biridir. Bozorda mavjud Case - texnologiyalardan Knowledge Ware firmasinig Application Development Workbech(ADW),BPwin(Logic Works-firmasi), CDEZ Tods(Oracle

-firmasi) maxsulotlarini misol qilish mumkin. Zamonaviy Case – texnologiyalari turli sinfdagi ATlar: banklar, moliyaviy korporatsiyalar, yirik firmalarni barpo etishda muvaffaqiyatli qo'llaniladi. Ular odatda anchagina qimmat turadi va ATni yaratish barcha jarayonlarini tubdan qayta tashkil qilish va uzoq o'qitishni talab etadi. Shunga qaramay, uni qo'llashning iqtisodiy samarasi juda ahamyatli va ko'pgina zamonaviy, jiddiy dasturiy loyixalar aynan zamonaviy Case – texnologiyalari yordamida amalga oshiriladi.

Nazorat savollari:

1. DT ning yo'nalishlarini sanab bering?.
2. Foydalanuvchi bilan muloqot deganda nimani tushunasiz?
3. 70-yillarda DT rivojlanishining qanday yo'llari boshlandi?
4. 80-yillarda DT soxasida qanday o'zgarishlar yuz berdi?
5. 90- yillarda DT soxasida qanday o'zgarishlar yuz berdi?
6. Dasturlash texnologiyasining mahsuloti xaqida gapirib bering.
7. Foydalanuvchining to'liq qayta ishlangan interfeysi qanday?.
8. Case texnologiyasi haqida gapiring.

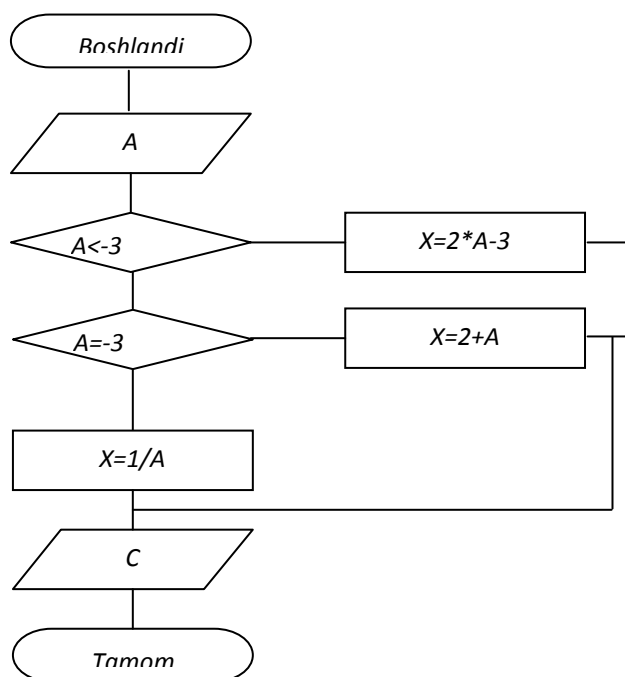
Tayanch iboralar: case, case texnologiyalari, dasturlash texnologiyasining mahsuloti.

3 – topshiriqda dasturchi ekranga chop ettirishni , eng oddiy arifmetik amallarni bajarishni va logik funksiyalardan foydalanishni o`rganadi. Bu dasturchi keyinchalik dasturlash mobaynida kerakli ma'lumotlarni ekranga chop ettirishda va kerakli ma'lumotlar ustida logik amallarni bajarishda kerak bo`ladi. Dastur ishlatishi va murakkab funksiyani hisoblash algaritmini tuzishi kerak.

3– topshiriqni yechish algoritmini ko`rib chiqamiz.

Har bir misolda noma'lum sonni solishtirishni amalga oshirish kerak va shu solishtirish natijasida X ni xisoblash kerak. Birinchi navbatda noma'lum sonni qiymatini o`zlashtirish kerak va solishtirish amalini bajarib , X ni qiymati hisoblanadi. X ni qiymatini ekranga chop ettirish kerak, ya'ni foydalanuvchi topshiriq natijalarini ko`rsin va shu bilan dastur tugaydi. Endi bu ketma - ketliklarni blok sxema ko`rinishida yozamiz.

Blok sxema:



Misol uchun 3 – topshiriqning 1 – misolini yechib ko`ramiz.

Bu misolda yangi projekt yaratib dasturlashni boshlaymiz.

Yangi projektни yaratishni fayl menusi yordamida yoki bosh saxifadan yangi projekt oynasini ochish orqali amalga oshiriladi. Fayl menyusi orqali ochish

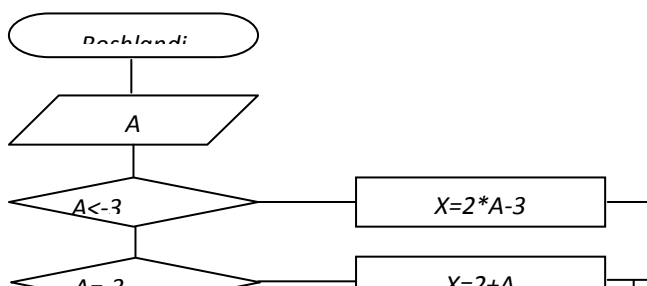
File – >New – >Project orqali amalga oshiriladi .

Yangi projekt oynasida dasturlash tilini, projekt turini tanlash, projekt nomini va qayerda saqlanishini ko`rsatish kerak. Men dasturlash tilini Visual C++, projektни turini Win32 Console Application, nomini Topshiriq1, saqlash joyini kursataman, **Ok** tugmasini bosaman. Bu oynada **Next >** tugmasini bosamiz va 5 – rasmdagi oyna ochiladi. Ochilgan oynada projektning xususiyatlari ko`rsatiladi undagi xech narsani o`zgartirmay **Finish** tugmasini bosamiz.

Bunda VS2005 ning asosiy oynasi va dasturiy qismi (kodi) saqlanadigan fayl ochilgan saxifa ham ochiladi. Bu saxifaga misolning dasturiy qismi yoziladi va shu faylda saqlanadi. Dasturlashga o`tishdan oldin unig algoritmini tuzish kerak. Dasturlash algoritmi yuqorida ko`rsatilgandek bo`ladi. 1 – misolni dasturlash algoritmini ko`rib chiqamiz.

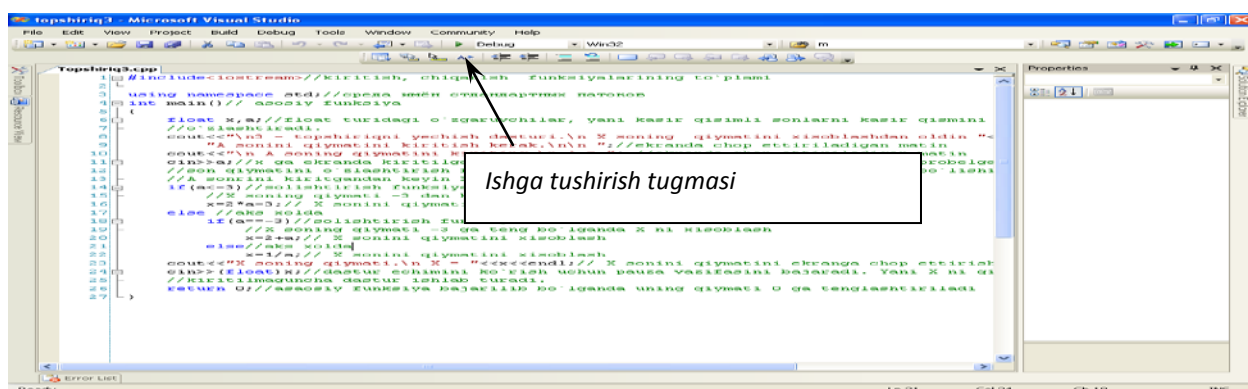
$$1 - \text{misol. } X = \begin{cases} 2A - 3 & ; A < -3 \\ 2 + A & ; A = -3; \\ 1/A & ; A > -3 \end{cases}$$

Nomalum A sonni qiymatini foydalanuvchi klaviatura yordamida kiritadi. A soning qiymatini solishtirishni amalga oshirish kerak va shu solishtirish natijasida X ni hisoblash kerak. X ni qiymatini ekranga chop ettirish kerak, ya`ni foydalanuvchi topshiriq natijalarini ko`rsin va shu bilan dastur tugaydi. Endi bu ketma - ketliklarni blok sxema ko`rinishida yozamiz.



Blok sxema:

Algoritmini yozib bo`lgandan keyin shu algoritim asosida dasturiy qismini yozamiz (7 – rasm).



Dasturiy qismini yozib bo`lgandan keyin dasturni kompilyatsiya qilish va ishga tushirish kerak. Kompilyatsiya va ishga tushirishni asosiy oynadagi ishga tushirish tugmasi bosiladi va dastur ishga tushadi. Ekranda qora oyna ochiladi (8 – rasm) A sonining qiymati klaviatura yordamida kiritiladi. X sonning natijasi ekranda chiqadi (9 – rasm) va bu natijani to`g`riligini tekshirib ko`ramiz (matematik usul bilan). Dastur yopilishi (tugashi) uchun ixtiyori sonni klaviatura yordamida kiritamiz va klaviaturadagi Enter tugmasini bosamiz. Shunda dastur o`z ishini tugatadi.

```
GA f:\SAODAT\Topshiriqlar\dasturi\topshiriq3\debug\topshiriq3.exe
3 - topshiriqni yechish dasturi.
X soning qiymatini xisoblashdan oldin A sonini qiymatini kiritish kerak.

A soning qiymatini kiriting.
A =
```

8 – rasm

```
GA f:\SAODAT\Topshiriqlar\dasturi\topshiriq3\debug\topshiriq3.exe
3 - topshiriqni yechish dasturi.
X soning qiymatini xisoblashdan oldin A sonini qiymatini kiritish kerak.

A soning qiymatini kiriting.
A = 2
X soning qiymati.
X = 0.5
```

9 – rasm

3-amaliy mashg'ulotga topshiriqlar:

$$1. x = \begin{cases} 2a - 3, & \text{agar, } a < 3 \\ 2 + a, & \text{agara} = -3 \\ 1 / a, & \text{agara} > 3 \end{cases}$$

$$2. a = \begin{cases} x^2 + 4x + 6, \text{ agar, } x \leq 9 \\ \frac{1}{x^2 + 4x}, \text{ agar, } x > 9 \end{cases}$$

$$3. a = \begin{cases} 0, \text{ agar, } x \leq 0 \\ y, \text{ agar, } 0 < y < 1 \\ y^2, \text{ agar, } y \geq 1 \end{cases}$$

$$4. c = \begin{cases} -2, \text{ agar, } a \leq 0 \\ a^2 - a, \text{ agar, } a > 0 \end{cases}$$

5. Quyidagi kvadrat tenglamani yeching : $ax^2+bx+c=0$.

$$6. a = \begin{cases} z^2, \text{ agar, } z < 2 \\ 2, \text{ agar, } z = 2 \\ 1 - z, \text{ agar, } z > 2 \end{cases}$$

$$7. 1 = \begin{cases} \sqrt{2p+3}, \text{ agar, } p > 0 \\ 1 - 2p, \text{ agar, } p = 0 \\ 3p^2 + 1, \text{ agar, } p < 0 \end{cases}$$

8. $a=b+c$, $b=2-c$ bo'sa $ax^2+bx+c=0$ kvadrat tenglama yechimga egami?

9. Tengliklarni yeching: $ax=b$; $a=3x-1$; $b=\frac{1}{2x+1}$.

$$10. h = \begin{cases} a + b, \text{ agar, } x < 2 \\ a - 1, \text{ agar, } x \geq 2 \end{cases}$$

$$11. g = \begin{cases} (3 - 2x), \text{ agar, } -2 \leq x < 2 \\ 0, \text{ agar, } x < -2 \\ x/(x+3), \text{ agar, } x \geq 2 \end{cases}$$

$$12. a = \begin{cases} x^2, \text{ agar, } x > 1 \\ 8, \text{ agar, } x = 1 \\ 1 - x, \text{ agar, } x < 1 \end{cases}$$

$$13. a = \begin{cases} 3x^2, \text{ agar, } x > 0 \\ x + 4, \text{ agar, } 0 \leq x \end{cases}$$

$$14. a = \begin{cases} x, \text{ agar, } x < 0 \\ 5, \text{ agar, } x = 0 \\ x^3, \text{ agar, } x > 0 \end{cases}$$

$$15. a = \begin{cases} \frac{10}{x}, \text{ agar } 1 \leq x < 4 \\ \sqrt{x}, \text{ agar } x < 1 \\ x, \text{ agar } x \geq 4 \end{cases}$$

$$16. a = \begin{cases} x, \text{ agar } x > 10 \\ \sqrt{x} + 2, \text{ agar } 0 \leq x \leq 10 \\ x + 10, \text{ agar } x < 0 \end{cases}$$

$$17. a = \begin{cases} 25x, \text{ agar } x > 1 \\ x^2 + 3, \text{ agar } 0 < x \leq 1 \end{cases}$$

$$18. a = \begin{cases} x^2 + x + 3, \text{ agar } 0 < x < 2 \\ (x - 2)^2, \text{ agar } x < 0 \end{cases}$$

$$19. a = \begin{cases} 3x + 2, \text{ agar } x > 0 \\ \sqrt{3x + 2}, \text{ agar } x < 0 \end{cases}$$

$$20. a = \begin{cases} x^2 + 1, \text{ agar } a < b \\ -(x^2) + 1, \text{ agar } a < b \end{cases}$$

$$21. a = \begin{cases} 2x + 7, \text{ agar } 0 < x < 3 \\ \frac{2}{x}, \text{ agar } x > 3 \\ \frac{x}{2}, \text{ agar } x \leq 0 \end{cases}$$

$$22. a = \begin{cases} \sqrt{3x} + 1, \text{ agar } x \geq 0 \\ (3x)^2 = 1, \text{ agar } x < 0 \end{cases}$$

$$23. a = \begin{cases} x + x^2, \text{ agar } x < 10 \\ x - x^2, \text{ agar } x > 10 \end{cases}$$

$$24. a = \begin{cases} \frac{x^2}{10}, \text{ agar } x > 0 \\ x, \text{ agar } x \leq 0 \end{cases}$$

$$25. a = \begin{cases} 3x^2, \text{ agar } x > 0 \\ x + 2, \text{ agar } x < 0 \\ -x + 10, \text{ agar } x = 0 \end{cases}$$

$$26. a = \begin{cases} \frac{2x}{3}, \text{ agar, } x \geq 40 \\ \frac{3x}{2}, \text{ agar, } x < 10 \end{cases}$$

$$27. a = \begin{cases} \sqrt{x}, \text{ agar, } x > 5 \\ x^2, \text{ agar, } x = 5 \\ x + 25, \text{ agar, } x < 5 \end{cases}$$

$$28. a = 28x^2 + 3 \quad b = 23/(x-3) \quad f = \begin{cases} a + b \\ a - b \end{cases}$$

$$29. a = \begin{cases} \frac{2}{3}x, \text{ agar, } x > 0 \\ 0, \text{ agar, } x = 0 \\ x + 3, \text{ agar, } x < 0 \end{cases}$$

$$30. a = \begin{cases} 3x + 2, \text{ agar, } x > 0 \\ \sqrt{3x + 2}, \text{ agar, } x < 0 \end{cases}$$

3 -amaliyot mashg'uloti uchun test

1. Nechanchi yilda dasturlash texnologiyasi samarali rivojlanishining yo'nalishlari boshlandi?

A.70

B.80

C.90

2. Nechanchi yilda texnologiya jarayonining nusxasi haqidagi hujjatlar paydo bo'ladi?

A. 70

B. 80

C.90

3. Nechanchi yilda kompyuter informatsiyasining muammolarini himoya qilishni keskin boshladi va axborot tarmog'iga o'tildi?

A. 90

B. 70

C.80

4. Bir dasturdan boshqasiga o'tishga soddalashtirish hususiyati qaysi interfeys orqali amalgam oshiriladi?

A. Dastur interfeysi B. Buyurtmachi interfeysi C. Foydalanuvchi interfeysi

5. Yangi tizimga o'tish Windowsga Windowsning oldingi versiyalarining qobiqlari bo'lgan dasturlar dispetcheri (Program Manager) va fayllar dispetcheri ham kirganliklari tufayli qanday xolga keladi?.

A. qiyinlashadi B. osonlashadi C. o'zgarmas qoladi

6. Case – texnologiyalarida qaysi soha mutaxassislari ishtirok etadilar ?

A. . tizimli taxlilchilar, foydalanuvchilar B. tizimli taxlilchilar, loyixachilar va dasturchilar C. dasturchilar xamda buxgalterlar

Javob: 1. A 2. B 3. A 4. C 5. B 6. B

Dastur kodi BMIning ilova qismi 3-ilova bandida keltirilgan.

4.4. “TEXNIK TOPSHIRIQ ” HUJJATI.

Reja:

1. “Texnik topshiriq” hujjati.
2. Dasturiy maxsulotga talablar.
3. Texnik topshiriq bo'limlari.
4. Texnik topshiriqqa tegishli hujjatlar:

Ishdan maqsad: “Texnik topshiriq” hujjati haqida amaliy ko'nikma hosil qilish.

Nazariy qism

Texnik topshiriq dasturiy maxsulotni asosiy hujjati hisoblanadi. Texnik topshiriq hujjatini yozishdan avval dasturiy mahsulotga quyidagi ekspluatatsion talablar ro'yxatini tuzish zarur:

1. Dasturiy cheklanishlar – bunda dastur ishlashi lozim bo'lgan operatsion tizim va ishlash jarayonida zarur bo'lgan dasturlash vositalari keltiriladi.

2. Ergonomik xarakteristikalar – sifatida foydalaguvchilar va operatorlar tomonidan ishlab chiqarish quvvatini oshiruvchi xususiyatlar ko'riladi.

“**Ergonomika**” grekcha “mehnat va qoida” manosida ishlatilib mehnat uchun eng yaxshi sharoit yaratish deganidir.

3. Ishonchlilik – dasturiy maxsulotning ishonchliligi deganda ma'lum bir dasturiy yoki qurilmalardagi uzilishlar mavjudligidan dasturiy maxsulotning ishini qayta tiklash imkoniyatiga aytiladi.

4. Moslashuvchanlik – dasturiy maxsulotni turli xil EXMLar bilan birgalikda ishlash imkoniyati.

5. Tekshiruvchanlik – ya'ni natijalarni tekshirish imkoniyatlari.

6. Universallik – har- xil turdagi ma'lumotlar kiritilganda dasturni to'g'ri ishlash extimolligi.

7. Effektivlik – texnika vositalari resursidan minimal sonidan foydalanish. Masalan: operativ xotirani minimal hajmini ishlatish.

8. Adaptatsiya – tez-tez o'zgartirish imkoniyatlari.

9. Dasturiy maxsulotning tannarxi.

Gost. 19.201-78 “Texnicheskoe zadanie. Trebovanie k sodержaniyu i oformleniyu”.

Bu standart bo'yicha texnik topshiriq quyidagi bo'limlardan iborat:

1. Kirish qismi – bunda dasturiy maxsulotning nomi kerakligi, aktualligi, shunga o'xshash dasturlar bilan solishtirish natijasi va agar yaratilgan dastur boshqa maxsulotning qismi sifatida ishlab chiqarilgan bo'lsa, uning faqat yangi xususiyatlari keltiriladi.

2. Ishlab chiqarishga asos – bunda dasturiy maxsulot qaysi hujjat asosida ishlab chiqilayapti, buyruq yoki shartnoma nomerlari.

3. Yo'llanma – bunda dasturiy maxsulotning asosiy maqsadi yoziladi.

Quyidagi dasturga texnik topshiriq yozish.

Maktab o'qituvchilari uchun mo'ljallangan bir argumentlik funktsiya $y=f(x)$ dan grafik tuzing.

Ushbu dastur hisoblash jadvalini ko'rsatish va berilgan intervalda berilgan formula bo'yicha argument qadamini o'zgartirish kerak.

Texnik topshiriq.

1. Kirish. Ushbu texnik topshiriq bitta argumentli funktsiya qiymatlarini va grafigini chizuvchi dasturga tegishli. Ushbu dastur yuqori sinfdagi maktab o'quvchilari uchun mo'ljallangan.

Maktabning elementar algebra kursida grafiklarni taxlil qilish va hisoblash masalasi muxim masala hisoblanadi.

Bu masalani matematik usulda yechish jarayonida o'quvchilar bir o'zgaruvchili funktsiya grafigini chizib, qiymatlar jadvalini tuzib, berilgan intervalda, berilgan qadam bilan funktsiyaning qiymatini topishi kerak.

Bu jarayonda maktab o'quvchilari funktsiyaning tashqi nuqtalarini, nuqtalar xarakteristikalarini, ildizlarini, birinchi va ikkinchi uzilish nuqtalari ta'rifini va formulalarini ishlatish kerak.

Dasturiy maxsulotga talablar.

1. Funktsional xarakteristikalar.

Dastur quyidagi funktsiyalarni bajara olishi kerak:

a) bir o'zgaruvchili funktsiyaning analitik ko'rinishini kiritish va uzoq vaqt davomida uni tizimda saqlab turish;

b) funktsiyaning aniqlash intervalini kiritish va o'zgartira olish;

v) argument qadamini kiritish va o'zgartirish.

Ko'rsatilgan oraliqda funktsiya uzilish nuqtalariga ega emas shartiga asosan shu intervalda funktsiya qiymatlar jadvalini hisoblash va grafigini tuzish.

2. Boshlang'ich ma'lumotlar:

a) funktsiyaning analitik qiymatlari;
b) funktsiyaning aniqlash intervali;
s) berilgan intervalda argument qadamini o'zgartiradigan holda nuqtalar sonini aniqlash.

3. Ishonchlilik:

a) kiritilayotgan ma'lumotlarni tekshirish imkoniyati;
b) dastur bilan ishlayotgan vaqtda foydalanuvchini noto'g'ri xarakatlaridan himoyalash.

4. Texnik parametrlarga talablar:

a) tizim shaxsiy kompyuterlar tizimida ishlashi kerak;
b) minimal konfiguratsiya: protsessor turi – Pentium va undan yuqori. Operativ hotiraning xajmi – 32Mg va undan yuqori

Dasturiy maxsulotning xarakteristikalarini – funksional xarakteristika: - kirish va chiqish ma'lumotlarining ko'rinishi va tartibi; effektivlik koefitsienti; operativ hotiraning maksimal xajmi.

5. Texnik topshiriqqa tegishli hujjat – sistemali dasturchiga ko'rsatma.

Texnik topshiriqqa tegishli hujjatlar:

a) ishlab chiqarilgan dastur listinglari shartli ravishda izohlarni o'z ichiga olishi kerak;

b) ishlab chiqarilayotgan dastur asosiy matematik terminlar va ma'lumotlar uchun axborotni o'z ichiga olishi kerak;

s) 25-30 varaqdan iborat tushuntirish xati;

d) sistemali dasturchiga ko'rsatma.

Nazorat savollari:

1. Texnik topshiriq hujjatini ko'rsating.

2. Obyektga yo'naltirilgan loyixalash texnologiyalari nima?

3. Tizimning obyektli modeli nima?

4. Ergonomik xarakteristikalar deganda nima tushiniladi?

5. Dastur universalligi nima?

6. Dastur qanday funktsiyalarni bajara olishi kerak?
7. Texnik parametrlarga qanday talablar qo'yilishi kerak?
8. Texnik topshiriqqa tegishli hujjatlarni sanab o'ring.

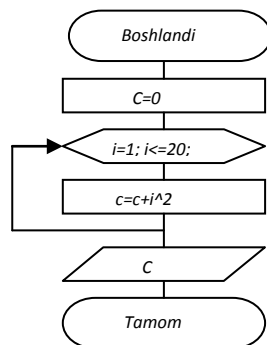
Tayanch iboralar: ergonomika, dasturiy cheklanish, adaptatsiya, ergonomik xarakteristika, ishonchlilik, moslashuvchanlik, effektivlik.

4 – topshiriq tavsifi.

4 – topshiriqda dasturchi ekranga chop ettirish, eng odiy arifmetik va murakkab amallarni bajarish, logik funktsiyalardan foydalanish siklik jaroyonlarni qurishni o'rganadi. Bu dasturchi keyinchalik dasturlash mobaynida kerakli ma'lumotlarni ekranga chop ettirish va kerakli ma'lumotlar ustida logik amallar bajarish, siklik jarayonlardan foydalanishni bilishi kerak bo'ladi.

4– topshiriqni yechish algoritmini ko'rib chiqamiz.

Ma'lum bir qatorning qiymatini hisoblash kerak va qiymatni ekranga chop ettirish kerak. Ushbu misolning blok sxemasi quyida ko'rsatilgan: Blok sxema:

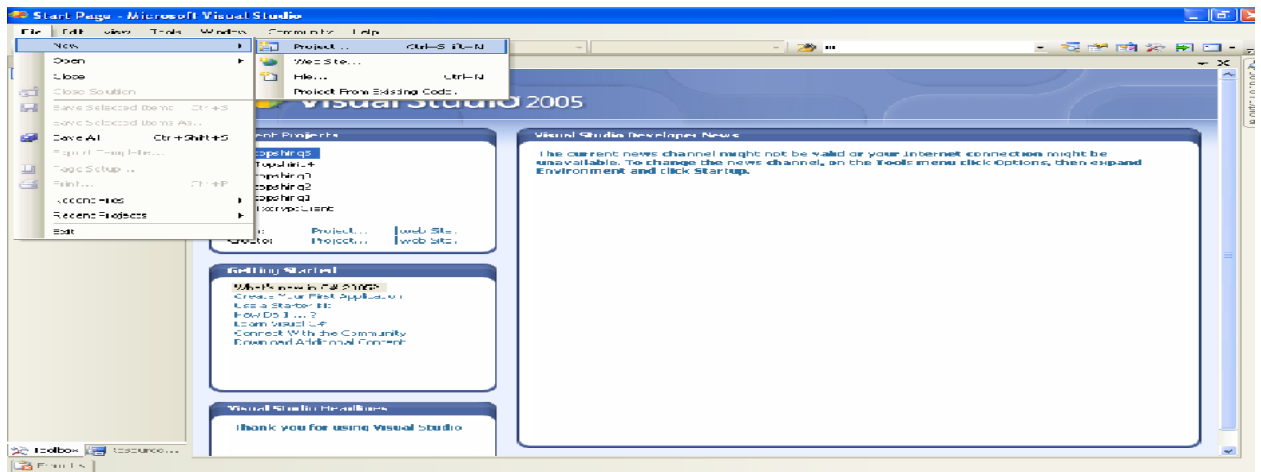


Misol uchun 4 – topshiriqning 1 – misolini yechib ko'rsatamiz.

Bu misolda yangi projekt yaratib dasturlashni boshlaymiz.

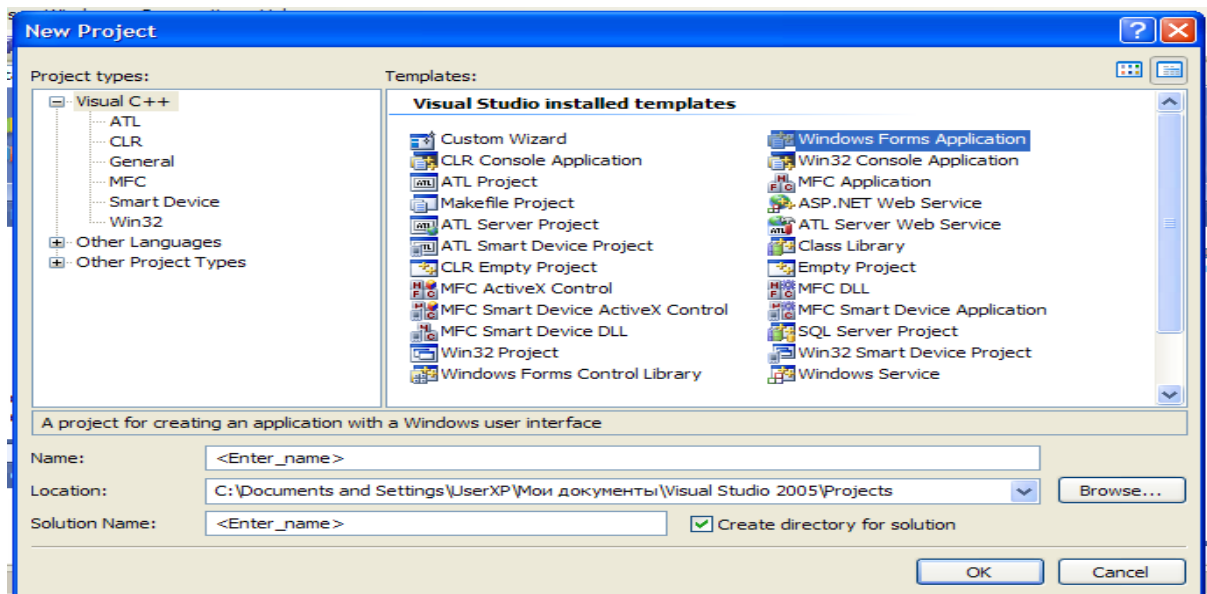
Yangi projektни yaratishni fayl menusi yordamida yoki bosh saxifadan yangi projekt oynasini ochish orqali amalga oshiriladi. Fayl menusi orqali ochish

File – >New – >Project orqali amalga oshiriladi.



2 – rasm

Yangi proyekt oynasida (3 - rasm) dasturlash tilini, proyekt turini tanlash, proyekt nomini va qayerda saqlanishini ko`rsatish kerak.



3 – rasm

Endi dasturlash tilini Visual C++, proyektning turini Win32 Console Application, nomini Topshiriq1, saqlash joyini kursatamiz, Ok tugmasini bosamiz va 4 – rasmdagi oyna ochiladi. Bu oynada **Next >** tugmasini bosamiz oyna ochiladi.

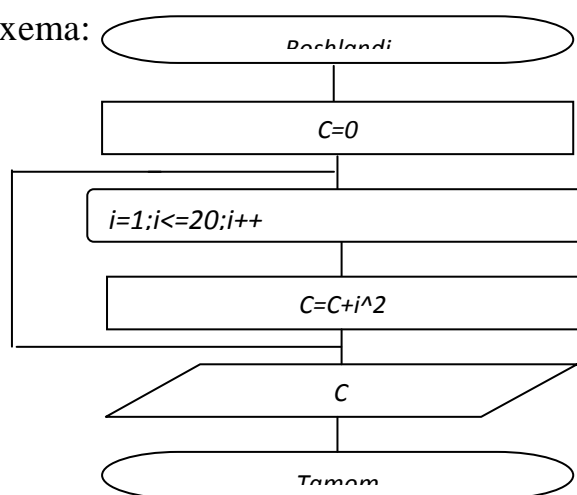
Ochilgan oynada proyektning hususiyatlari ko`rsatiladi bunda hech narsani o`zgar tirmay Finish tugmasini bosamiz.

Bunda VS2005 ning asosiy oynasi va dasturiy qismi (kodi) saqlanadigan fayl ochilgan saxifa ham ochiladi. Bu saxifaga misolning dasturiy qismi yoziladi va shu faylda saqlanadi. Dasturlashga o`tishdan oldin unig algoritmini tuzish kerak.

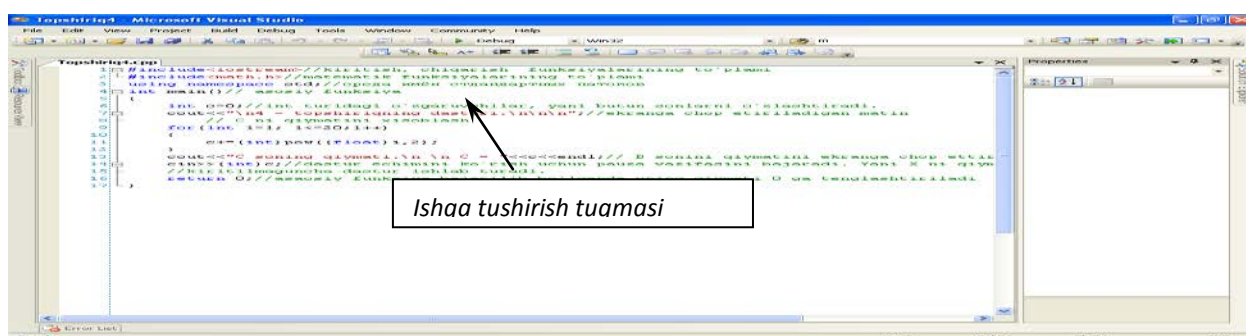
Dasturlash algoritmi yuqorida ko`rsatilgandek bo`ladi. 1 – misolni dasturlash algoritmini ko`rib chiqamiz.

$$1 - \text{misol. } S = 1^2 + 2^2 + 3^2 \dots + 20^2 ; ;$$

Misolni umumiy sodda ko`rinishga keltiriylik $S = \sum i^2$ i 1 dan 20 gacha butun sonlar. Buda siklik jarayonni qursak maqsadga muvofiq boladi i 1 dan 20 gacha o`zgarishi kerak, qadami +1 va S ni qiymati mos ravishda qo`shilib borishi kerak to sikl tugamagunicha. S ni qiymatini ekranga chop ettirish kerak, ya'ni foydalanuvchi topshiriq natijalarini ko`rsin va shu bilan dastur tugaydi. Endi bu ketma - ketliklarni blok sxema ko`rinishida yozamiz. Blok sxema:



Algoritmini yozib bo`lgandan keyin shu algoritim asosida dasturiy qismini yozamiz



Dasturiy qisimni yozib bo`lgandan keyin dasturni kompilatsiya qilish va ishga tushirish kerak. Kompilatsiya va ishga tushirishni asosiy oynadagi ishga tushirish tugmasi orqali amalgam oshiramiz va dastur ishga tushadi. Ekranda qora oyna ochiladi , C sonning natijasi ekranda chiqadi va bu natijani to`g`riligini tekshirib ko`ramiz (matematik usul bilan). Dastur yopilishi (tugashi) uchun ixtiyoriy sonni

klaviatura yordamida kiritamiz va klaviaturadagi Enter tugmasini bosamiz. Shunda dastur o`z ishini tugatadi.

```

f:\SAODAT\Topshiriqlar\dasturi\Topshiriq4\debug\Topshiriq4.exe
4 - topshiriqning dasturi.
C soning qiymati.
C = 2870

```

4-amaliy mashg'ulotga topshiriqlar:

1. $1^2+2^2+3^2+4^2+5^2+\dots$ qo'shiluvchilar soni 5 ta.
2. $1-\frac{1}{3}+\frac{1}{5}-\frac{1}{7}+\frac{1}{9}+\dots$ qo'shiluvchilar soni 4 ta.
3. $1+\frac{1}{3^2}+\frac{1}{5^2}+\frac{1}{7^2}+\dots$ qo'shiluvchilar soni 6 ta.
4. $1^3+2^3+3^3+\dots$ qo'shiluvchilar soni 8 ta.
5. $1-\frac{1}{2^2}+\frac{1}{3^2}+\frac{1}{4^2}+\dots$ qo'shiluvchilar soni 3 ta.
6. $1-\frac{x^2}{2!}+\frac{x^4}{4!}-\frac{x^6}{6!}+\dots$ qo'shiluvchilar soni 6ta.
7. $1-\frac{1}{2}+\frac{1}{3}-\frac{1}{4}+\frac{1}{5}+\dots$ qo'shiluvchilar soni 9 ta.
8. $\frac{x^2}{2}+\frac{x^3}{3}-\frac{x^4}{4}+\dots$ qo'shiluvchilar soni 10ta.
9. $1+2x+3x^2+4x^2+5x^2+\dots$ qo'shiluvchilar soni 7 ta.
10. $\frac{1}{x}+\frac{2}{x^2}+\frac{3}{x^3}+\frac{4}{x^4}+\dots$ qo'shiluvchilar soni 9 ta.

11. $\frac{1}{2} + \frac{3}{2^2} + \frac{5}{2^3} + \frac{7}{2^4} + \dots$ qo'shiluvchilar soni 5 ta.

12. $1 - x^2 + \frac{x^4}{2!} - \frac{x^6}{3!} + \frac{x^8}{4!} - \dots$ qo'shiluvchilar soni 6 ta.

13. $2^2 + 6^2 + 10^2 + 14^2 + \dots$ qo'shiluvchilar soni 8 ta.

14. $1 - \frac{1}{3 \cdot 3} + \frac{1}{5 \cdot 3^2} - \frac{1}{7 \cdot 3^3} + \dots$ qo'shiluvchilar soni 10 ta.

15. $1 + 3x + 5x^2 + 7x^3 + \dots$ qo'shiluvchilar soni 12 ta.

16. $\sin x + \frac{\sin 3x}{x} + \frac{\sin 5x}{5} + \frac{\sin 7x}{7} + \dots$ qo'shiluvchilar soni 11 ta.

$X=0.3$

Javob: 0.7853

17. $X^5 + X^6 + X^7 + X^8 + \dots$ qo'shiluvchilar soni 6 ta.

$X=0.8$

Javob :1.3107

18. $1 \cdot 2 + 2 \cdot 3x + 3 \cdot 4x^2 + 4 \cdot 5x^3 + \dots$ qo'shiluvchilar soni 13 ta.

Javob: 78.125

19. $x - \frac{x^2}{3} + \frac{x^5}{5} + \frac{x^7}{7} + \dots$ qo'shiluvchilar soni 6 ta.

$X=0.2$

Javob:0.1823

20. $1 - \frac{1}{3 \cdot 3} + \frac{1}{5 \cdot 3^2} - \frac{1}{7 \cdot 3^3} + \dots$ qo'shiluvchilar soni 8 ta.

Javob:0.9068

21. $1 - \frac{x}{1!} + \frac{x^2}{2!} - \frac{x^3}{3!} + \dots$ qo'shiluvchilar soni 9 ta.

$X=4$

Javob:0.0183

22. $\frac{1}{4 \cdot 5} + \frac{1}{5 \cdot 6} + \frac{1}{6 \cdot 7} + \frac{1}{7 \cdot 8} + \dots$ qo'shiluvchilar soni 7 ta.

Javob: 0.2187

23. $2^2 + 6^2 + 10^2 + 14^2 + \dots$ qo'shiluvchilar soni 6 ta.

Javob: 3876

24. $1 - 3x^2 + 5x^4 - 7x^6 + \dots$ qo'shiluvchilar soni 9 ta.

$X=0.4$

Javob :0.4458

25. $1 + \frac{1}{3^2} + \frac{1}{5^2} + \frac{1}{7^2} + \dots$ qo'shiluvchilar soni 10 ta.
26. $1 - \frac{1}{2^2} + \frac{1}{3^2} - \frac{1}{4^2} + \dots$ qo'shiluvchilar soni 11 ta.
27. $\frac{1}{x} + \frac{2}{x^2} + \frac{3}{x^3} + \frac{4}{x^4} + \dots$ qo'shiluvchilar soni 12 ta.
28. $2^2 + 6^2 + 10^2 + 14^2 + \dots$ qo'shiluvchilar soni 10 ta.
29. $X^5 + X^6 + X^7 + X^8 + \dots$ qo'shiluvchilar soni 8 ta.
 $X=0.8$ Javob :1.3107
30. $1 - \frac{1}{3*3} + \frac{1}{5*3^2} - \frac{1}{7*3^3} + \dots$ qo'shiluvchilar soni 9 ta.
 Javob:0.9068

4-amaliyot mashg'uloti uchun test

- Texnik topshiriq xujjatini yozishdan oldin dasturiy maxsulotga nechta ekspluatatsion talablar ro'yxatini tuzish zarur?
 A.5ta B.3ta C.9ta
- Tez-tez o'zgartirish imkoniyatlari deb qaysi ekspluatatsion talablarga aytiladi?
 A. Adaptatsiya B. Moslashuvchanlik C. Effektivlik
- Natijalarni tekshirish imkoniyatlari qaysi ekspluatatsion talablarga aytiladi?
 A. Moslashuvchanlik B. Tekshiruvchanlik C. Effektivlik
- “Ergonomika”** qanday ma'noni anglatadi?
 A.Lotinch “mehnat va qoida”
 B. Grekcha ”mexnat va o'qish” C. Grekcha “mehnat va qoida”
- Standart bo'yicha texnik topshiriqning qaysi bo'limida dasturiy maxsulotning asosiy maqsadi yoziladi?
 A. Yo'llanma B. Ishlab chiqarishga asos C. Kirish qismi
- Standart bo'yicha texnik topshiriqning qaysi bo'limida dasturiy maxsulotning nomi kerakligi, aktuvalligi, shunga o'xshash dasturlar bilan solishtirish natijasi va agar

yaratilgan dastur boshqa maxsulotning qismi sifatida ishlab chiqarilgan bo'lsa, uning faqat yangi xususiyatlari keltiriladi.

A. Yo'llanma B. Ishlab chiqarishga asos C. Kirish qismi

7. « Texnik topshiriqqa tegishli hujjat – bu... » jumlani to'ldiring.

A. Sistemali foydalanuvchiga ko'rsatma.

B.Sistemali dasturchiga ko'rsatma.

C.Sistema buyurtmachiga ko'rsatma.

8. Texnik topshiriqqa tegishli xujjatdagi tushuntirish xati necha varroqdan iborat bo'lishi kerak ?

A. 10-15

B.15-20

C.25-30

Javob : 1. C 2.A 3.B 4. C 5.A 6. C 7.B 8. C

3. G. Mayers. Nadejnosty programmogo obespecheniya. - M.: Mir, 1980.

Dastur kodi BMIning ilova qismi 4-ilova bandida keltirilgan.

5.5.DASTURLASH TEXNOLOGIYA ARXITEKTURASI “TASHQI SPETSIFIKATSIYA” HUJJATI.

Reja:

1. Dasturiy vosita arxitekturasi tushunchasi;
2. DV arxitekturasi asosiy bosqichlari;
3. Tashqi spetsifikatsiya hujjati.

Ishdan maqsad: Dasturlash texnologiya arxitekturasi hamda

“Tashqi spetsifikatsiya “ hujjati haqida amaliy ko'nikma hosil qilish.

Nazariy qism

Dasturiy vosita arxitekturasi tushunchasi: DV arxitekturasi – bu uning tuzilishi uning tashqaridan ko'rinishi, ya'ni DV birga ishlaydigan quyi tizimlarning jamidan iborat tizim kabi tasavvur etishdir. Bunday quyi tizimlar sifatida odatda alohida dasturlar ajralib turadi. Arxitekturani ishlab chiqish DT murakkabligining birinchi bosqichi hisoblanadi.

DT arxitekturasi ishlab chiqishning asosiy masalalari:

- Dasturiy quyi tizimlarni ajratish va ularda DVning tashqi funktsiyalarni (tashqi tavsifda berilgan) aks ettirish;
- Ajratilgan dasturiy quyi tizimlar o'rtasidagi o'zaro bog'liqlik usullarini aniqlash;
- Ushbu bosqichda qabul qilinadigan qarorlarni hisobga olgan holda funksional tavsiflarni konkretlashtirish amalga oshiriladi.

Dasturiy vositalar arxitekturalarining asosiy bosqichlari

Dasturiy vositalar arxitekturalarining quyidagi asosiy klasslari mavjud:

1. Maqsadga ega dastur;
2. Avtonom ravishda bajariladigan dasturlar kompleksi;
3. Qatlamga ega dasturiy tizim;
4. Parallel bajariladigan dasturlar kollektivi.

Maqsadga ega dastur DV arxitekturasi chala holatini taqdim etadi. DV tarkibiga faqat bitta dastur kiradi.

Konstruktsiyalashda (design) DV quyidagi jarayonlarni qamrab oladi: DV arxitekturasi ishlab chiqish, DV dasturlarining strukturalari va ularning batafsil tavsifini ishlab chiqish.

Tashqi spetsifikatsiya hujjati.

Agar hujjatlarda biror-bir satxlarni aniqlovchi maqsadlar keltirilgan bo'lsa, ular **spetsifikatsiya** deb ataladi.

“Spetsifikatsiya” o'ziga xoslikdir. Spetsifikatsiyalar yengil o'qiluvchan, qisqa, aniq, ammo tugal axborot beruvchi hujjatdir.

Loyixalashtirish- jarayonida modullarni aniqlash, rejalarni ajratish, ularning tuzilishini aniqlash, talablarni aniqlab ularga moslashib ichki va tashqi

spetsifikatsiyalarni belgilash jarayonlaridan boshlanib kuzatib borish bo'limi ishlab chiqarish bilan tugallanishi lozim.

Dasturiy maxsulotning tashqi spetsifikatsiyasida maxsulotni tashqi ishini tavsiflovchi hujjatlar keltiriladi. Bu hujjatlar dasturning ichki tuzilishiga tegishli emas.

Tashqi spetsifikatsiya hujjatini yozishda quyidagilarni e'tiborga olish lozim.

1. Chaqirilayotgan modul xisoblashdagi axborot shu modulni o'ziga qarashli bo'lgan tashqi spetsifikatsiyaga tegishli bo'lishi lozim.

2. Tashqi spetsifikatsiya hujjati foydalanuvchi va dasturchi uchun yaqin bo'lgan tilda yoziladi.

3. Modullar shunday ko'rinishda yozilishi kerakki, ularga biror-bir o'zgartirish kiritilganda bu o'zgartirishlar modullarga tegishli bo'lmagan boshqa modullarga ta'sir ko'rsatmasligi kerak.

4. Tashqi spetsifikatsiyani baxolovchi shaxs faqat shu spetsifikatsiyada aniqlangan hususiyatlarni e'tiborga olishi lozim. Masalan: agar tashqi spetsifikatsiyada qo'shimcha tushuntirishlarsiz quyidagi tushuncha keltirilgan bo'lsin:

a) o'zgaruvchi parametrlari 1dan 10gacha bo'lgan qiymatlarni o'zlashtirishi mumkin.

Shu tushuncha keltirilgan baxolovchi a parametrning qiymati sifatida xar qanday xaqiqiy sonni xam nazarda tutishi mumkin. Shu bilan birgalikda a parametrning qiymati sifatida 1 va 10 qiymatlar ko'rinmasligi mumkin.

5. Cheklanishlar tugal va aniq ko'rinishga ega bo'lib, ularning kelib chiqish sabablarini ko'rsatish shart emas, odatda bu sabablar dasturchilarga mazkur modullarni kengaytirish va o'zgartirish uchun zarur bo'lsa, ichki spetsifikatsiyalarda keltiriladi.

6. Tashqi spetsifikatsiyalarning tugalligini va aniqligini tekshirishni dasturlash jarayonidan oldin bajarilishi tavsiya qilinadi. Shunga ko'ra dasturlash jarayonida avval ushbu hujjat foydalanuvchi tomonidan tekshirilishi lozim.

“Tashqi spetsifikatsiya hujjati” texnik topshiriq hujjatining davomi sifatida rasmiylashtirilgan bo’lib, uni odatda “texnik loyiha” deb ataladi.

Bu hujjatda dastur modulining tashqi xususiyatlari tavsiflanmog’i lozim.

Tashqi spetsifikatsiya hujjati quyidagilardan iborat:

1. Dastur nomi va tavsifi.
2. Foydalanuvchi bilan muloqotni amalga oshirish.
3. Menyu va funksional klavishlarni ishini tavsiflash.
4. Barcha ekran (monitor) ko’rinishlarini tavsiflash.
5. Dastur moduli ishlash jarayonida beriladigan savollar va ularga javob berish yo’llari.
6. Xatoliklar haqida ma’lumot.
7. Ma’lumotlar bazasini tashkil qilish.
8. Tashqi ko’rinishlarni tavsiflash.

Nazorat savollari:

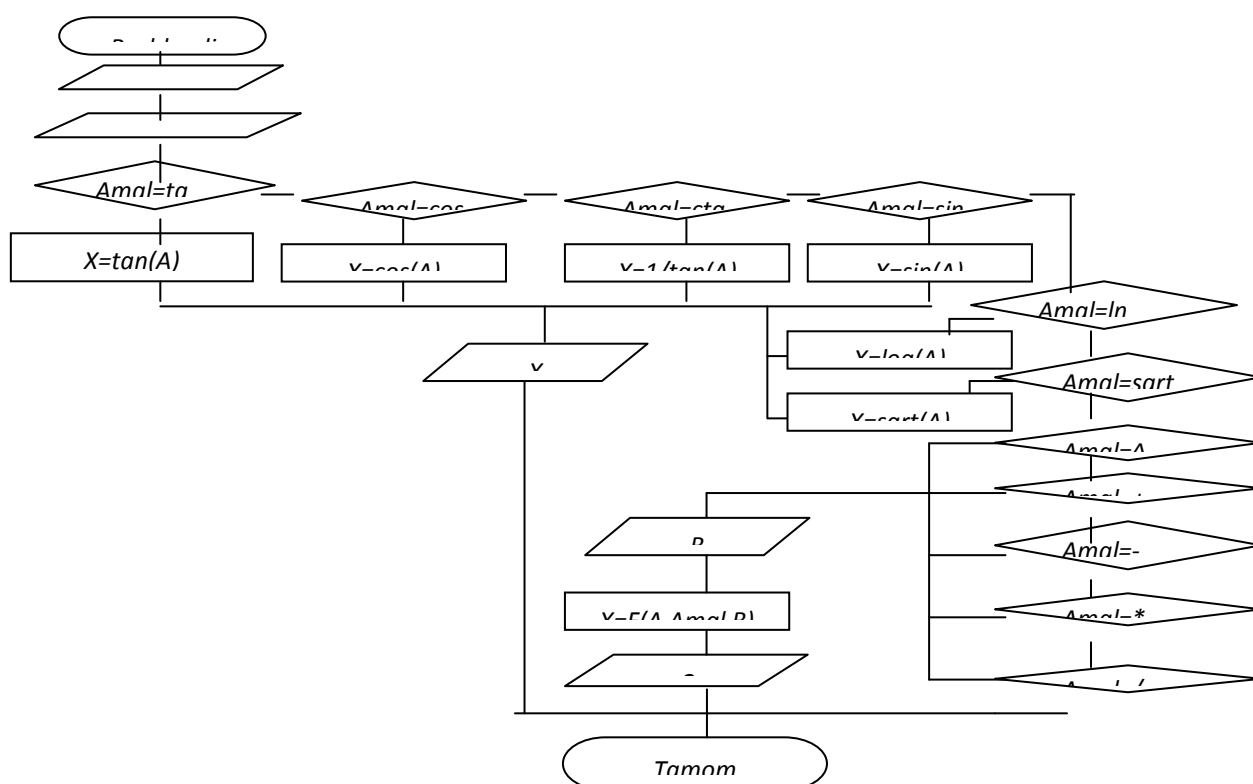
1. Dasturiy ta’minotni tashqi loyixalash nima?
2. Algoritmni ishlab chiqish nima?
3. Tashqi loyixalash nima?
4. “Tashqi spetsifikatsiya” xujjatini ko’rsating
5. Foydalanuvchi bilan muloqat tashkil qilish.
6. DT arxitekturasi asosiy sinflarini sanab o’ting.
7. Tashqi spetsifikatsiya xujjatini yozishda nimalarni e’tiborga olish lozim?
8. Tashqi spetsifikatsiya xujjati nimalardan iborat?

Tayanch iboralar: DV arxitekturasi, konstruksiyalash, spetsifikatsiya, loyihalashtirish.

5-topshiriq tavsifi

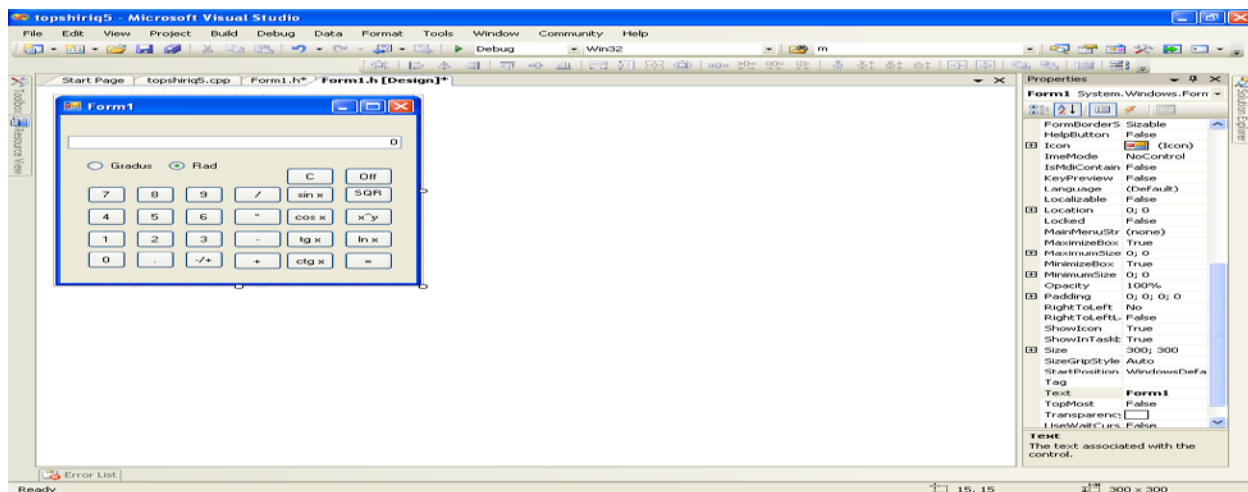
5 – topshiriqda dasturchi dasturlash texnologiyalarini yaxshi o`rgangan bo`lishi va class, metod, svoystva, event siklik jarayonlar, logic funksiyalarni va murakkab, oddiy matematik, trigonametrik amallarni hisoblashni bilishi kerak.

5 – topshiriqdagi 1 – misolni misol uchun olamiz ya'ni kalkulyatorning dasturini yozamiz . Kalkulyatorning klassik modeli bu a sonni ustida birma-bir amal bajarishdan iborat. A sonini qiymatini foydalanuvchi kiritadi amalni ham foydalanuvchi tanlaydi, hisob- kitob amalini bajargandan keyin natijani ekranga chop ettiradi. Quyidagiidagi blok sxema yordamida dasturni tuzamiz:



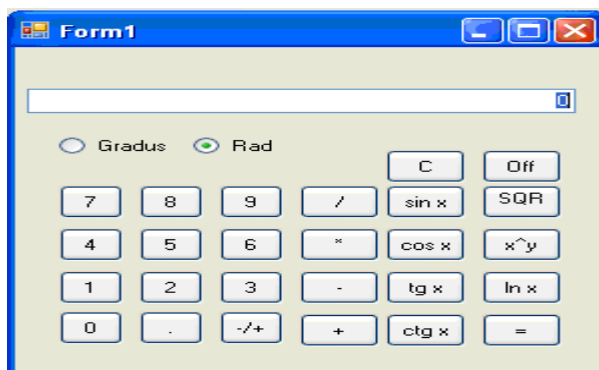
Kalkulyator Windows kalkulatoriga o`xshagan bo`lishi uchun uning ish muhiti oynadan iborat bo`lishi kerak. Shuning uchun projekt yaratganda projekt turini Windows Forms Application ni tanlirimiz va VS2005 ning asosiy ishchi oynasida bosh forma oynaning asosi ochilgan saxifa ochiladi. Istrumentlar qutisidan foydalanib tugma, matn, yozuv oynasini formaga joylashtiramiz o`zimizga yoqqan shaklda va tugmalarning ust qismidagi yozuvga mos ravishda kalkulyator tugmalarini yozuviga moslab yozish mumkin yoki o`zingizning hoxishingizga ko`ra yoki oqituvchining talabiga ko`ra joylashtirishingiz mumkin. Ko`rinishini (1 – rasm)

o`zingizga yoqqan holga keltirgandan so`ng uning dasturiy qismini yozishga o`tishingiz mumkin.



1 – rasm

Dasturiy qismini tugatganimizdan so`ng uni kompilyatsiya qilib ishga tushiramiz va ekranda biz yaratgan kalkulyator oynasi ochiladi nomi Form1(2 – rasm). Endi kalkulyator ishlashini tekshiramiz to`g`ri xisob kitob amallarini bajarvottimi yo`mi tekshirib ko`ramiz(matematik usul bilan qalam, ruchka, qog`oz)



2 – rasm

5-amaliy mashg`ulotga topshiriqlar:

Topshiriq 1.Fakultet talabalarining o`zlashtirishi.

So`ralgan guruhning yoki fakultet tarkibiga kiruvchi barcha guruh talabalar ro`yxati (kurs va guruh bo`yicha ajratilgan holda), har bir talabaning fanlardan olgan reyting ballari ($RB=N1 + N2 + YaN$), ularga mos keluvchi «5» ballik baholash

tizimidagi baxosi, jadval ko'rinishida foydalanuvchining hoxishiga ko'ra ekranga yoki printeriga chiqarilsin. Guruh talabalarining ro'yxatidan keyin "qarzdor" talabalar ro'yxati (Fan + RB) ham ekranga yoki printeriga chiqarilsin.

Topshiriq 2.Fakultet talabalarining davomati.

So'ralgan guruhning yoki fakultet tarkibiga kiruvchi barcha guruh talabalar ro'yxati (kurs va guruh bo'yicha ajratilgan holda), har bir talabaning berilgan vaqt oralig'ida qoldirgan soatlar soni(fanlar bo'yicha ajratilgan holda) foydalanuvchining hoxshiga ko'ra ekranga yoki printeriga chiqarilsin.

Topshiriq 3.Shaxmat o'yinining modeli.

Shaxmat o'yinining modelini dasturiy ta'minotini ishlab chiqish kerak.

O'yinda faqat quyidagi figuralar qatnashsin: figuralar: shox va farzin; qora figuralar: shox oq

Ularning shaxmat doskasidagi boshlangich holatlari foydalanuvchi ta'rafdan kiritilsin. O'yin 10 ta xodgacha davom etsin. O'yinda yurilgan xodlar protokoli va o'yin natijasi foydalanuvchining hoxshiga ko'ra ekranga yoki printeriga chiqarilsin (10 ta xod ichida oqlar qoralarni mot qilolmasa yoki doskada fakat shoxlar qolsa o'yin natijasi - durang hisoblansin).

O'yin qatnashchilari:

- oqlar: foydalanuvchi 1;
- qoralar: foydalanuvchi 2.

Topshiriq 4.Shaxmat o'yinining modeli.

Shaxmat o'yinining modelini dasturiy ta'minotini ishlab chiqish kerak.

O'yinda fakat quyidagi figuralar qatnashsin: oq figuralar: shox; qora figuralar: shox va ikki to'ra.

Ularning shaxmat doskasidagi boshlangich holatlari foydalanuvchi ta'rafdan kiritilsin. O'yin 10 ta xodgacha davom etsin. O'yinda yurilgan xodlar protokoli va o'yin natijasi foydalanuvchining hoxishiga ko'ra ekranga yoki printeriga chiqarilsin (10 ta xod ichida oqlar qoralarni mot qilolmasa yoki doskada faqat shoxlar qolsa o'yin natijasi - durang hisoblansin).

O'yin qatnashchilari:

- oqlar: foydalanuvchi 1;
- qoralar: foydalanuvchi 2.

Topshiriq 5. Chizili tenglamalar sistemasini yechish.

Berilgan 4 ta noma'lumli 4ta tenglamalar sistemasi va ularning yechimlari foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin. Tenglamalar sistemasini yechishda Kramer usuli yoki Teskari matritsa usulidan foydalanilsin. Foydalanuvchining hoxishiga ko'ra bu usullar to'grisida qisqacha nazariy ma'lumotlar ekranga chiqarilishini tashkil etilsin.

Topshiriq 6. Algebraik tenglamalarni yechish.

Uchinchi darajali algebraik tenglamaning $(Ax^3 + Bx^2 + Cx + D = 0)$ yechimlarini berilgan aniqlikda topish kerak. Buning uchun quyidagi usullardan biridan (foydalanuvchining hoxishiga ko'ra) foydalanish kerak: oraliqni ikkiga bo'lish usuli; vatarlar usuli; urinmalar usuli. Foydalanuvchining hoxishiga ko'ra bu usullar to'grisida qisqacha nazariy ma'lumotlar ekranga chiqarilishini tashkil etilsin.

Topshiriq 7. Egri chizikli trapetsiyaning yuzini taqriban hisoblash.

$y = f(x)$ egri chiziq va $y=0$, $x=a$, $x=Y$ to'g'ri chiziqlar bilan chegaralangan shaklning yuzini talab qilingan aniqlikda hisoblash kerak. Buning uchun quyidagi usullardan biridan (foydalanuvchining hoxishiga ko'ra) foydalanish kerak:

- trapetsiyalar usuli;
- Simpson usuli.

Foydalanuvchining hoxishiga ko'ra bu usullar to'grisida qisqacha nazariy ma'lumotlar ekranga chiqarilishini tashkil etilsin.

$$y = \sqrt{1-52} \quad x = -1 \quad x = 1 \quad p = 20;$$

$$u = \dots \quad x = 0 \quad x = 4 \quad p = 40;$$

$$V = 9 + 0.2$$

Topshiriq 8. Funksiyaning qiymatlar jadvaliga asosan grafigini yasash.

Berilgan 3 ta funksiyaning (a, Y) oraliqdagi p ta qiymatlarini topib, jadval ko'rinishida foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin.

Foydalanuvchining hoxishiga ko'ra funktsiyalardan birining grafigi ekranga yoki printerga chiqarilsin.

$$y = \sin(a_1 x + b_1) + S_y$$

$$z = \ln(a_2 x + b_2) + c_2$$

$$t = Ax^4 + Bx^2 + C$$

Topshiriq 9. Tub sonlarga oid masala.

N_1 va N_2 natural sonlar orasida joylashgan tub sonlar ro'yxatini jadval ko'rinishida foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin. Foydalanuvchining hoxishiga ko'ra tub sonlar to'grisidagi qisqacha ma'lumotni ekranga chiqarilsin.

Topshiriq 10. Mukammal sonlar.

Berilgan N natural sondan kichik bo'lgan barcha mukammal sonlar topilsin va foydalanuvchining hoxishiga ko'ra ekranga yoki printerga bo'linuvchilar yig'indisi ko'rinishida chiqarilsin. Foydalanuvchining hoxishiga ko'ra mukammal sonlar to'grisidagi qisqacha ma'lumotni ekranga chiqarilsin.

Masalan:

$$1+2+3=6; \quad 1+2+4+7+14=28.$$

Topshiriq 11. Armstrong sonlari.

Armstrong sonlari ($p=2,3,4$ uchun) topilsin va foydalanuvchining hoxishiga ko'ra ekranga yoki printerga yig'indi ko'rinishida chiqarilsin. Foydalanuvchining hoxishiga ko'ra mukammal sonlar to'grisidagi qisqacha ma'lumotni ekranga chiqarilsin.

Masalan:

$$p=3 \quad 153=13+53+33;$$

$$p=4 \quad 1634=14+64+34+44.$$

Topshiriq 12. Tarjimon o'yini.

Ingliz tilini o'rganish uchun ingliz tilidan o'zbek tiliga tarjima qilish o'yini dasturi tuzilsin. EXM foydalanuvchi ekraniga ingliz tilida yozilgan bir so'zni chiqaradi, foydalanuvchi so'zning tarjimasini ekranga chiqaradi. Tarjima noto'gri

bo'lsa, qaytadan so'raydi (3 ta imkoniyat beriladi). To'g'ri bo'lsa keyingi so'z ekranga chiqariladi.

O'yin dasturi 2ta rejimda:

- tayyorlanish (o'yinni tamom qilish foydalanuvchining hoxishi bo'yicha);
- imtixon (chiqariladigan so'zlar soni foydalanuvchi ta'rifdan

oldindan kiritiladi. Masalan: Yuzta yoki 20 ta yoki 50 ta va x.z.

1-chi urinishda to'g'ri javob bersa - 5 ball, 2-chi urinishda to'g'ri javob bersa - 3 ball, 3-chi urinishda to'g'ri javob bersa - 2 ball berilib, to'plangan ballar maksimal ballning:

86% va undan yuqori - a'lo,

71% dan to 85% gacha - yaxshi,

55% dan to 70%o gacha - o'rta baxo qo'yilsin va ekranga chiqarilsin.

Topshiriq 13.O'zaro tub sonlar.

N ta berilgan sonlar ichidan o'zaro tub sonlar juftliklarini aniqlab, ro'yxatini jadval ko'rinishida foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin. Foydalanuvchining hoxshiga ko'ra o'zaro tub sonlar to'grisidagi qiskacha ma'lumotni ekranga chiqarilsin.

Topshiriq 14. Pifagor sonlari.

Berilgan N dan kichik bo'lgan barcha Pifagor sonlarini (uchlik (a, Y, s)) aniqlab, ro'yxatini jadval ko'rinishida foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin. Foydalanuvchining hoxishiga ko'ra Pifagor sonlari to'grisidagi qiskacha ma'lumotni ekranga chiqarilsin.

Topshiriq 15.Yigindini hisoblash va max ni topishga doir.

N ta xaqiqiy sonlardan iborat to'plamdagi barcha tub sonlarning yigindisini, tub son bo'lmagan sonlar ichidagi max ga ko'paytmasini toping.

Topshiriq 16. Kalendar.

■

So'ralgan ixtiyoriy yilning bir yillik kalendar jadvalini ekranga chiqarilsin. Foydalanuvchining hoxishiga ko'ra kalendar jadvalini chop etishda 2 usuldan biri qo'llanilsin:

- bir ekran saxifasiga bir oy kalendar jadvali chop etilsin;
- bir ekran saxifasiga bir yillik kalendar jadvali chop etilsin.

Topshiriq 17. Shashka o'yining modeli.

Shashka o'yinining modelini dasturiy ta'minotini ishlab chiqish kerak.

O'yinda fakat quyidagi figuralar qatnashsin:

oq figuralar: 3 ta damka; qora figuralar: 1 ta damka.

Ularning shashka doskasidagi boshlangich holatlari foydalanuvchi ta'rafidan kiritilsin. "o'yin 15 ta xodgacha davom etsin. O'yinda yurilgan xodlar protokoli va o'yin natijasi foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin (15 ta xod ichida oqlar qora damkani tutib olmasa o'yin natijasi - durang hisoblansin).

O'yin qatnashchilari:

- oqlar: foydalanuvchi 1;
- qoralar: foydalanuvchi 2.

Topshiriq 18. Kalkulyator modeli.

Kalkulyator modelini ishlab chiqish kerak. Kalkulyator da 4ta arifmetik amallardan tashqari, sonni darajaga oshirish, $\sin x$, $\cos x$, $\tan x$, $\cot x$ va $\ln x$ qiymatlarini hisoblash mumkin bo'lsin.

Topshiriq 19. O'zaro birodar sonlar.

Berilgan N natural sondan kichik bo'lgan o'zaro birodar sonlarning ro'yxati ekranga chiqarilsin. Foydalanuvchining hoxishiga ko'ra o'zaro birodar sonlar to'grisidagi qisqacha ma'lumotni ekranga chiqarilsin. Masalan: 220 va 284 1; 2; 4; 5; 10; 11; 20; 22; 44; 55; PO - 220 sonining bo'linuvchilari. 1; 2; 71; 142 - 284 sonining bo'linuvchilari.

$$1+2+4+5+10+11+20+22+44+55+110=284 \quad 1+2+71+142=220$$

Topshiriq 20. Merser tub sonlari.

Berilgan N natural sondan kichik bo'lgan barcha Merser tub sonlarining ro'yxati ekranga chiqarilsin. Foydalanuvchining hoxishiga ko'ra Merser tub sonlar

tugrisidagi qisqacha ma'lumotni ekranga chiqarilsin (2" -1 ko'rinishidagi tub sonlar,r - ham tub son).

Topshiris 21.Avtomorf sonlari.

Berilgan N natural sondan kichik bo'lgan barcha avtomorf sonlarining ro'yxati ekranga chiqarilsin. Foydalanuvchining hoxishiga ko'ra avtomorf sonlari to'grisidagi qisqacha ma'lumotni ekranga chiqarilsin.

Masalan: $52=25$ yoki $25'' = 625$.

Topshiris 22.Sonlarning EKUK va EKUBlari.

Berilgan N ta natural sonlarning EKUK va EKUB larini toping va ekranga chiqaring. Foydalanuvchining hoxishiga ko'ra sonlarning EKUK va EKUB lari to'grisidagi qiskacha ma'lumotni ekranga chiqarilsin.

Topshiris 23.Talabalar uyi .

Talabalar uyida yashovchi talablar ro'yxati blok, qavat, xonalar bo'yicha va talabalar to'grisidagi ma'lumotlar kurs, guruh, o'kuv turi (byudjet yoki shartnoma), pasport seriyasi, viloyati, xarbiy yo'llanmasi bor yo'qligi va boshqa qo'shimcha ma'lumotlar kiritilsin va foydalanuvchining talabiga ko'ra ma'lumotlar jadval ko'rinishida:

- umumiy ro'yxat 1-chi ko'rinishi (famiyalari buyicha tartiblangan holda);
- umumiy ro'yxat 2-chi ko'rinishi (blok , qavat va xonalar bo'yicha);
- umumiy ro'yxat 3-chi ko'rinishi (kurs va guruhlar bo'yicha tartiblangan holda);
- umumiy ro'yxat 4-chi ko'rinishi (kurs va viloyatlar bo'yicha tartiblangan holda);

ekranga yoki printerga chop etilsin.

Topshiris 24. Mersen tub sonlari.

Berilgan N natural sondan kichik bo'lgan barcha Mersen tub sonlarining ro'yxati ekranga chiqarilsin. Foydalanuvchining hoxishiga ko'ra Mersen tub sonlar tugrisidagi qisqacha ma'lumotni ekranga chiqarilsin (2" -1 ko'rinishidagi tub sonlar,r - ham tub son).

Topshiriq 25. Fakultet talabalarining o'zlashtirishi.

So'ralgan guruhning yoki fakultet tarkibiga kiruvchi barcha guruh talabalar ro'yxati (kurs va guruh bo'yicha ajratilgan holda), har bir talabaning fanlardan olgan reyting ballari ($RB=N1 + N2 + YaN$), ularga mos keluvchi «5» ballik baholash tizimidagi baxosi, jadval ko'rinishida foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin. Guruh talabalarining ro'yxatidan keyin "qarzdor" talabalar ro'yxati ($Fan + RB$) ham ekranga yoki printerga chiqarilsin

Topshiriq 26. Chizikli tenglamalar sistemasini yechish.

Berilgan 4 ta noma'lumli 4ta tenglamalar sistemasi va ularning yechimlari foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin. Tenglamalar sistemasini yechishda Kramer usuli yoki Teskari matritsa usulidan foydalanilsin. Foydalanuvchining hoxishiga ko'ra bu usullar to'grisida qisqacha nazariy ma'lumotlar ekranga chiqarilishini tashkil etilsin.

Topshiriq 27. Fakultet talabalarining davomati.

So'ralgan guruhning yoki fakultet tarkibiga kiruvchi barcha guruh talabalar ro'yxati (kurs va guruh bo'yicha ajratilgan holda), har bir talabaning berilgan vaqt oralig'ida qoldirgan soatlar soni(fanlar bo'yicha ajratilgan holda) foydalanuvchining hoxshiga ko'ra ekranga yoki printerga chiqarilsin

Topshiris 28. Shashka o'yining modeli.

Shashka o'yinining modelini dasturiy ta'minotini ishlab chiqish kerak.

O'yinda fakat quyidagi figuralar qatnashsin:

oq figuralar: 3 ta damka; qora figuralar: 1 ta damka.

Ularning shashka doskasidagi boshlangich holatlari foydalanuvchi ta'rafidan kiritilsin. "o'yin 15 ta xodgacha davom etsin. O'yinda yurilgan xodlar protokoli va o'yin natijasi foydalanuvchining hoxishiga ko'ra ekranga yoki printerga chiqarilsin (15 ta xod ichida oqlar qora damkani tutib olmasa o'yin natijasi - durang hisoblansin).

O'yin qatnashchilari:

- oqlar: foydalanuvchi 1;
- qoralar: foydalanuvchi 2.

Topshiris 29. Avtomorf sonlari.

Berilgan N natural sondan kichik bo'lgan barcha avtomorf sonlarining ro'yxati ekranga chiqarilsin. Foydalanuvchining hoxishiga ko'ra avtomorf sonlari to'grisidagi qisqacha ma'lumotni ekranga chiqarilsin.

Masalan: $52=25$ yoki $25 = 625$.

Topshiris 30. Talabalar uyi .

Talabalar uyida yashovchi talablar ro'yxati blok, qavat, xonalar bo'yicha va talabalar to'grisidagi ma'lumotlar kurs, guruh, o'kuv turi (byudjet yoki shartnoma), pasport seriyasi, viloyati, xarbiy yo'llanmasi bor yo'qligi va boshqa qo'shimcha ma'lumotlar kiritilsin va foydalanuvchining talabiga ko'ra ma'lumotlar jadval ko'rinishida:

- umumiy ro'yxat 1-chi ko'rinishi (famiyalari buyicha tartiblangan holda);
- umumiy ro'yxat 2-chi ko'rinishi (blok , qavat va xonalar bo'yicha);
- umumiy ro'yxat 3-chi ko'rinishi (kurs va guruhlar bo'yicha tartiblangan holda);
- umumiy ro'yxat 4-chi ko'rinishi (kurs va viloyatlar bo'yicha tartiblangan holda);

ekranga yoki printerga chop etilsin.

5 -amaliyot mashg'uloti uchun test:

1. DV arxitekturasi – bu...

*A. uning tuzilishi uning tashqaridan ko'rinishi, ya'ni DV birga ishlaydigan quyi tizimlarning jamidan iborat tizim kabi tasavvur etishdir.

B. Dasturiy quyi tizimlarni ajratish va ularda DVning tashqi funktsiyalarni (tashqi tavsifda berilgan) aks ettirish

C. Ajratilgan dasturiy quyi tizimlar o'rtasidagi o'zaro bog'liqlik usullarini aniqlash.

2. Dasturiy vositalar arxitekturalarining asosiy bosqichlari nechta?

A.2 * B. 4 C.8

3. Maqsadga ega dastur DV arxitekturasi qanday holatini taqdim etadi?

A.tugallangan *B.chala C.tugallanmagan

4. “Spetsifikatsiya” so’zi qanday ma’noni anglatadi?

*A. o’ziga xoslik B. rasmiylashtirish C.rejalashtirish

5. “Dasturiy maxsulotning tashqi spetsifikatsiyasida maxsulotni ishini tavsiflovchi xujjatlar keltiriladi”-jumlaning to’ldiring.

A.... B.ichki * C.tashqi

6. “Tashqi spetsifikatsiya xujjati” texnik topshiriq xujjatining davomi sifatida rasmiylashtirilgan bo’lib, u odatda qanday nomlanadi?

*A. texnik loyixa B.texnik topshiriq C.texnik manba

7. “Tashqi spetsifikatsiya xujjati”da dastur modulining qanday xususiyatlari tavsiflanmog’i lozim?

A.tabiiy B.meyoriy * C.tashqi

8. Xujjatda dastur moduli ishlash jarayonida beriladigan savollar va ularga javob berish yo’llari ko’rsatiladimi?

A. xa B. yo’q C. savollar ko’rsatiladi, ammo ularning javob berish yo’llari ko’rsatilmaydi.

Javoblari: 1-A 2-B 3-B 4-A 5- C 6-A 7- C 8-A

Dastur kodi BMIning ilova qismining 5-ilova bandida berilgan.

6.6. KICHIK DASTURLAR STANDARTI, “ICHKI SPETSIFIKATSIYA” HUJJATI.

Reja:

1. Kichik dasturlar standarti;
2. Kutubxonalar;
3. Dasturlarni ko’chirish;

4. Ichki spetsifikatsiya hujjati.

Ishdan maqsad: Kichik dasturlar standarti o'rganish va "Ichki spetsifikatsiya hujjati haqida amaliy ko'nikma hosil qilish.

Nazariy qism

Kichik dasturlar standarti.

Hujjatlarni va dasturga kerakli bo'lgan talablarni unifikatsiyalash maqsadida ular yagona dasturiy hujjatlar sistemasiga (ESPD) standartlashtirilgan. Uning tarkibiga turi va shaklini aniqlab beruvchi standartlar, hujjatlar tarkibi, uni kuzatib borish, yana qo'shimcha sifatida dasturiy hujjatlarni avtomatlashtirilishini ta'minlovchi standartlar kiradi.

Xar bir dastur, kichik dastur va jarayonlar nima vazifani bajarishligi haqidagi izoxlardan boshlanadi. Kiritish izohlaridagi ma'lumotlar quyidagilardan iborat bo'ladi:

1. Dasturdan maqsad;
2. Dasturni chaqirish va undan foydalanish haqidagi ko'rsatkichlar;
3. Asosiy massiv va o'zgartirgichlarning ro'yxati va vazifasi;
4. Kiritish - chiqarish haqidagi ko'rsatkichlar xamda barcha fayllar ro'yxati;
5. Foydalanilayotgan barcha qism programmalar ro'yxati;
6. Foydalanilgan matematik metodlar va ular haqidagi ma'lumotga ega bo'lgan adabiyotlar ro'yxati;
7. Dasturni bajarilgan vaqti haqida ma'lumot;
8. Kerak bo'lgan hotira xajmi ;
9. Operatorga maxsus ko'rsatmalar ;
10. Muallif haqida ma'lumot;
11. Dastur yozilgan sana.

Kutubxonalar.

Dasturni samaradorligini oshirish, sozlash va tekshirishni yengillashtirish hamda dastur yaratishni qisqartirish uchun dastur kutubxonalari qo'llaniladi. Kutubxona uchun mo'lljallangan kichik dasturlar kategoriyasiga dasturlash tiliga ega bo'lgan funksiyalarni yoki kichik dasturlarni o'z ichiga oladi. Bu kategoriyalarga barcha standart funktsiyalar ($_$, sin, cos va hokozolar), kiradi.

Ikkinchi funktsiya va kichik dasturlar manbasi sizning hisoblash sistemangizdan tashqarida joylashgan. Foydalunuvchiga u yoki bu sistemada bajarilgan dasturlar to'g'ri keladi. Kutubxona dasturlardan foydalanishning sabablaridan biri - ishonchlikni oshirish va dasturlashdagi qiyinchiliklarni kamaytirishdir. Kutubxona dasturlarining uchinchi manbasi bu dasturlar va hisoblash mashinalari to'g'risidagi kitoblardir.

Dasturni kompanovkalash va tashqi aloqalarni taxrirlash.

Tashqi aloqalarni taxrirlash jarayonida tashqi aloqalar taxriri dasturi yoki vazifalarini tartiblovchi shunday sintaktik xatolarni aniqlaydiki, bu xatolarda kichik dasturlarda yozilgan prаметrlar qiymati bir biriga mos kelmaydi, aslida yo'q bo'lgan standart dasturlarni chiqaradi. Masalan SIN o'rniga SIH va shunga o'xshash xatolar.

Dasturlarni ko'chirish.

Ko'chirish samarasi va tez sozlanuvchi dasturni tayyorlashga yordam beradi. Katta dasturlarda ishlashda, ularning asosiy qismini modellashtirish uchun kichik dasturlarni yozish maqsadga muvofiq bo'ladi.

Katta dasturlar uchun keng kiritish-chiqarish va dasturlash bilan testdan o'tkazish uchun ko'p vaqt kerak bo'lishi mumkin. Modelda kiritishning bo'lmasligi va soddalashtirilgan chiqarishlar bo'lishi mumkin. Bu o'rtacha dastur boshqacha ishlovchi real dasturga qaraganda ikki tartibga kamroq bo'lishi kerak bo'ladi. Yakuniy yechim yaqinligi topilganda ham, model asosiy yordamni berishi kerak bo'ladi. Model dasturni har qanday tilda yozishi mumkin va loyixa hamda algoritm tekshirishda foydalanish mumkin.

Bunday yondashishlardan uncha murakkab bo'lmagan dasturlarni loyixalashda foydalanish tavsiya etiladi. Agar kichik dasturlarning o'lchamlari katta bo'lmasa, unda kichik dasturlarni ajratib ularni loyixalashni boshlagan maquloqdir.

Dasturlashda quyidagi sozlash vositalari qo'llaniladi:

1. Xotira tarkibini taxlil qilish;
2. Algoritm bajarilishini nazorat qilish;
3. O'zgaruvchilardagi foydalanishni nazorat qilish;
4. Indeksni tekshirish;
5. Kichik dasturlardan foydalanishni nazorat qilish;
6. O'zgaruvchilarni belgilarini tasvirlash.

Dasturni bajarilishini mantiqiy yo'lini xisobga oladi. Undan dasturchilar tomonidan berilgan jarayonni bajarilish ketma-ketligini va o'zgaruvchilarni hozirgi vaqtdagi belgisini bajarilishini va xisobga olishini tekshirishda foydalaniladi. Bunday kuzatishlarning uchta turi mavjud:

1. O'zgaruvchilarni belgilarini nazorat qilish ;
2. Kichik dasturlarni chiqarishni xisobga olishni nazorat qilish ;
3. Dasturdagi buyruqlarni yetkazishni nazorat qilish.

Ikkinchi turdagi kuzatish asosan kichik dasturlardan foydalanilgan paytda sozlashda qo'llaniladi. Har bir muomaladagi yozuvga kichik dasturni nomi chiqadi, undan chiqilganda-asosiy dasturga qaytilganligi haqida ma'lumot chiqadi.

Bu turdagi tekshirish dasturchiga dasturiy xatolarni aniqlashdagi barcha kerakli ma'lumot bilan ta'minlab turadi. Uning kamchiligi mashina vaqtini ko'p sarf bo'lishidir, hamda berilayotgan ma'lumotlar bir necha ming qatordan iborat bo'lishidadir. Shuning uchun yuqoridagi kamchiliklardan voqif bo'lish uchun dasturni qisimlarga ajratib belgilangan joylarda o'chirib yoqib tekshiriladi.

Ichki spetsifikatsiya hujjati

Spetsifikatsiya - dastur tarkibini va uning hujjatlarni o'z ichiga oladi. Asosiy nusxalar egasining vedimostlarida dasturiy xujjatlar saklovchi korxonalar tarkibi ko'rsatilgan. Kerakli tushuntirishlarga ega bo'lgan dasturiy tekst ham dasturiy hujjatlar hisoblanadi. Bu hujjat bir yoki bir necha qisimli tekstlarni o'z ichiga oladi.

Uning har birida dastur algoritmik tilning belgilari yordamida yozilgan bo'ladi. Dastur tarkibi quyidagilarni o'z ichiga oladi. Umumiy ma'lumotda funktsional vazifasi, mantiqiy tarkibini tartibi tashki, ichki aloqalar, kiruvchi ma'lumotlarni chiqarish va yuklanish bo'yicha instruksiyalar, chiquvchi ma'lumotlar. Texnik topshirik va tashki spetsifikatsiya hujjatlari yuqoridagi ma'lumotlarni mavjud bo'lishini taqozo qiladi. Chunki texnik topshiriq va tashki spetsifikatsiyalarda yaratilgan dasturiy maxsulotlar haqida to'liq axborot keltiriladi. Yuqoridagilardan farqli o'laroq ichki spetsifikatsiya hujjati maxsulotni qay darajada tuzilganligini xamda unga qo'yilgan talab va maqsadlarni akslantiradi. Bu hujjat dasturchi uchun.

I. S hujjati dastur maxsuloti yaratish jarayonini akslantirib xar bir bosqich ishi tugallangandan so'ng yoziladi. Juda ko'p hollarda bu hujjat ichki loyixa nomiga xam ega. Bu hujjatni dasturlash bosqichi boshlanishdan avval rasmiylashtirish tavsiya qilinadi.

Shunga ko'ra bu hujjatda dastur maxsulotini yaratish uchun lozim bo'lgan barcha algoritmlarning tavsifi kengaytiriladi. Bu vaqtga kelib barcha algoritm kattaliklarning to'liq tavsifi mavjud bo'ladi. Blok sistemalar esa shu darajada mukammal ravishda yaratilgan bo'lishi kerakki, ular asosida yozilgan dastur xatolikkga ega bo'lmasin. I.S hujjatning asosiy vazifasi dasturiy maxsuloti hakidagi barcha ma'lumotlarni berishdan tashkil topgan bo'lib unda loyixani ichki mazmuni algortimlari va kelgusida ishlatishga foydali bo'lgan modullar haqidagi tushuncha keltiriladi.

I.S hujjatining standart ko'rinishi:

1. Dastur maxsulotning tavsifi;
2. Maqsadlar;
3. Taktika.

Bu bo'limda ham yuqoridagi hujjatlarda keltirilgan axborot birgalikda yangi terminlar va sintaksik qoidalar haqidagi tushunchalar keltiriladi.

Nazorat savollari :

1. Kichik dasturlar standarti haqida gapirib bering.
2. Dastur kutubxonalar deganda nimani tushunasiz ?
3. Dasturlarni ko'chirish qanday amalga oshiriladi ?
4. Ichki spetsifikatsiya hujjatini so'zlab bering.
5. Ichki spetsifikatsiya va tashqi spetsifikatsiya hujjatlarining farqlarini ayting.
6. Dasturni kompanovkalash nima?
7. Spetsifikatsiya nima ma'noni anglatadi?
8. I.S xujjatning standart ko'rinishi qanday ?

Tayanch iboralar: ESPD, kutubxonalar.dasturni komponovkalash, dasturlarni ko'chirish, spetsifikatsiya.

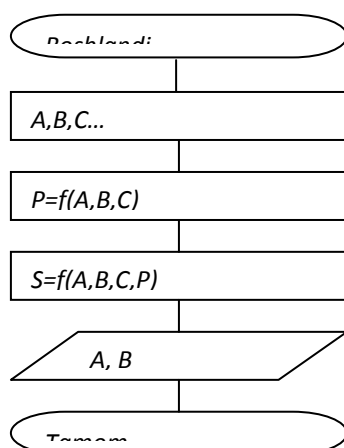
6 – topshiriq tavsifi.

6 – topshiriqda dasturchi ekranga chiqarishning eng odiy, arifmetik amallarni bajarishni murakkab , turli geometrik masalalarni yechib, yechimini ekranda chiqarishni o'rganadi. Bu dasturchining keyinchalik dasturlash mobaynida kerakli geometrik masalalarni yechish va qiymatlarni ekranga chiqarishda kerak bo`ladi.

6 – topshiriqni yechish algoritmini ko`rib chiqili.

Har bir topshiriqda ma'lum bir geometrik shaklni parametrlarni hisoblash kerak. Bu parametrlarni hisoblashda berilgan qiymatlarni mas ravishda berilgan formulalarga qo`yib hisoblash kerak. Hisoblangan qiymatlarni ekranga chop ettirish kerak, ya'ni foydalanuvchi topshiriq natijalarini ko`rsin va shu bilan dastur tugaydi. Endi bu ketma ketliklarni blok sxema ko`rinishida yozamiz:

Blok sxema:



Misol uchun 6 – topshiriqning 1 – misolini yechib ko`ramiz:

Bu misolda yangi projekt yaratib dasturlashni boshlaymiz.

Yangi projektni yaratishni fayl menusi yordamida yoki bosh saxifadan yangi projekt oynasini ochish orqali amalga oshiriladi. Fayl menusi orqali ochish **File – >New – >Project** orqali amalga oshiriladi. Yangi projekt oynasida dasturlash tilini, projekt turini tanlash, projekt nomini va qayerda saqlanishini ko`rsatish kerak. Man dasturlash tilini Visual C++, projektni turini Win32 Console Application, nomini Topshiriq1, saqlash joyini ko`rsataman, Ok tugmasini bosaman va 4 – rasmdagi oyna ochladi. Bu oynada **Next >** tugmasini bosamiz. Ochilgan oynada projektning xususiyatlari ko`rsatiladi men esa xech narsani o`zgar tirmay **Finish** tugmasini bosaman. Bunda VS2005 ning asosiy oynasi va dasturiy qismi (kodi) saqlanadigan fayl ochilgan saxifa ham ochiladi (6 – rasm). Bu saxifaga misolning dasturiy qismi yoziladi va shu faylda saqlanadi. Dasturlashga o`tishdan oldin unig algoritmini tuzish kerak. Dasturlash algoritmi yuqorida ko`rsatilgandek bo`ladi. 1 – misolni dasturlash algoritmini ko`rib chiqamiz.

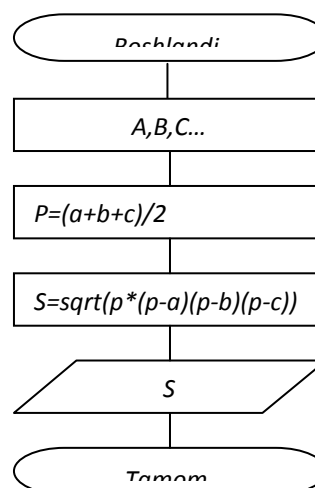
1 – misol.

Uchburchak shaklidagi to`sinning tomonlari uzunliklari mos holda $a=3,6m$, $b=4,8m$, $c=3,4m$ ga teng ekanligi ma'lum bo`lsa to`sinning kesim yuzini toping:

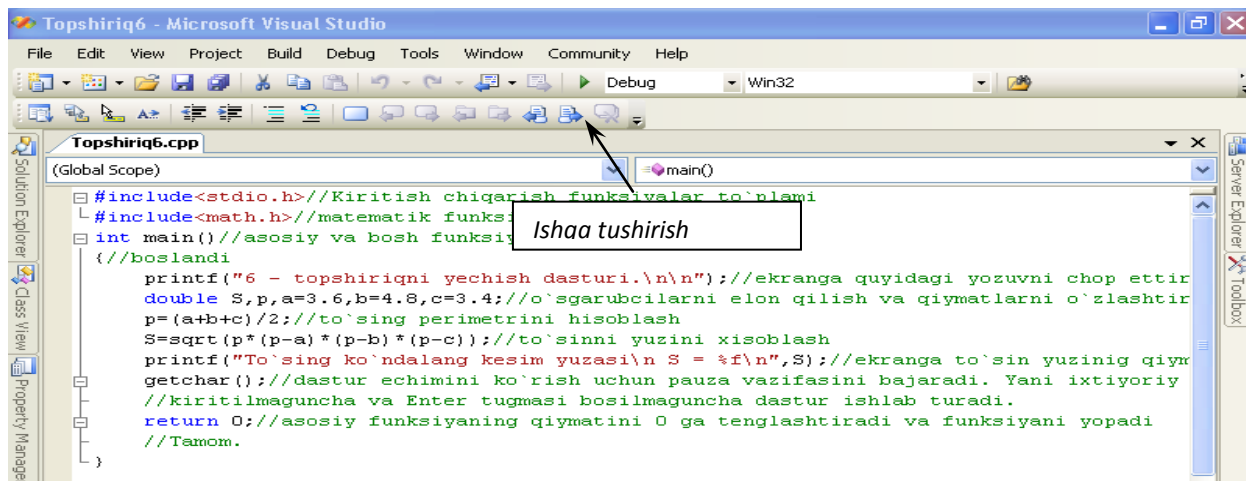
$$S = \sqrt{p(p-a)(p-b)(p-c)}, \quad p=(a+b+c)/2$$

Arifmetik amallarni bajarib P va S sonning qiymatini hisoblash kerak. S sonning qiymati uchburchakning ko`ndalang kesim yuzasiga teng va shu qiymatni ekranga chiqarish kerak, ya'ni foydalanuvchi topshiriq natijalarini ko`rsin va shu bilan dastur tugaydi. Endi bu ketma ketliklarni blok sxema ko`rinishida yozamiz:

Blok sxema:



Algarirmini yozib bo`lgandan keyin shu algaritim asosida dasturiy qismini yozamiz



```
#include<stdio.h>//Kiritish chiqarish funksiyalar to`plami
#include<math.h>//matematik funksiyalar
int main();//asosiy va bosh funksiya
{//boslandi
printf("6 - topshiriqni yechish dasturi.\n\n");//ekranga quyidagi yozuvni chop ettir
double S,p,a=3.6,b=4.8,c=3.4;//o`sgarubcilarni elon qilish va qiymatlarni o`zlashtir
p=(a+b+c)/2;//to`sing perimetrini hisoblash
S=sqrt(p*(p-a)*(p-b)*(p-c));//to`sinni yuzini xisoblash
printf("To`sing ko`ndalang kesim yuzasi\n S = %f\n",S);//ekranga to`sin yuzining qiymatini chop ettir
getchar();//dastur echimini ko`rish uchun pauza vazifasini bajaradi. Yani ixtiyoriy
//Kiritilmaguncha va Enter tugmasi bosilmaguncha dastur ishlab turadi.
return 0;//asosiy funksiyaning qiymatini 0 ga tenglashtiradi va funksiyani yopadi
//Tamom.
}
```

Dasturiy qisimni yozib bo`lgandan keyin dasturni kompilatsiya qilish va ishga tushirish kerak. Kompilatsiya va ishga tushirishni asosiy oynadagi ishga tushirish tugmasi bosiladi va dastur ishga tushadi. Ekranda qora oyna ochiladi va uchburchakning ko`ndalang kesim yuzasining qiymati S ekranda chiqadi. S sonning natijasini to`g`riligini tekshirib ko`ramiz (matematik usul bilan). Dastur yopilishi (tugashi) uchun ixtiyoriy sonni klaviatura yordamida kiritamiz va klaviaturadagi Enter tugmasini bosamiz. Shunda dastur o`z ishini tugatadi.



```
6 - topshiriqni yechish dasturi.
To`sing ko`ndalang kesim yuzasi
S = 6.108805
```

6-amaliy mashg`ulotga topshiriqlar:

Berilgan masalani yechishning hisoblash jarayoniga mos algaritm (blok-sxema tarzida) hamda Beysik yoki C++ algaritmik tilida dasturini tuzing.

1. Qurilish tashkilotidagi suvoqchilar brigadasining birinchi suvoqchisi bir soatda $a=2.4 \text{ m}^2$, ikkinchi suvoqchisi $b=1,76 \text{ m}^2$, uchinchi suvoqchisi $s=1.6 \text{ m}^2$, to`rtinchi suvoqchisi esa $d=2.46 \text{ m}^2$, sirtni suvoq qilinishi ma`lum bo`lsa, u holda butun ish kuni mobaynida (7 soat) qancha metr kvadrat sirtni suvoq qiladi.

$$\{ S=(a+b+c+d) * t \}$$

2. To'rtburchak shaklidagi to'sinning diagonallari mos holda $d_1=2.6$ va $d_2=3.4$ ga teng. Diagonallar orasidagi burchak $d=\pi/4$ ga teng ekanligi ma'lum bo'lsa, shu to'sinning yuzini hisoblang

$$(S = \frac{1}{2} d_1 d_2 \sin \alpha).$$

3. Uchburchak shaklidagi to'sinning tomonlari uzunliklari mos holda $a=3.6$ m, $b=4.8$ m, $c=3.4$ m ga teng ekanligi ma'lum bo'lsa, to'sinning ko'ndalang kesim yuzini toping:

$$(S = \sqrt{p(p-a)(p-b)(p-c)}, p=(a+b+c)/2).$$

4. Uchburchak shaklidagi to'sinning tomonlari uzunliklari mos holda $a=3.6$ m, $b=4.8$ m va $c=3.4$ m ga teng ekanligi ma'lum bo'lsa, to'sinning burchaklarini toping:

$$\alpha = 2 \arcsin \sqrt{(p-b)(p-c)(b-c)}.$$

$$\beta = 2 \arcsin \sqrt{(p-a)(p-c)(a-c)}$$

$$\omega = 2 \arcsin \sqrt{(p-b)(p-a)(b-a)}$$

bunda $p=(a+b+c)/2$.

5. Balandligi $H=22.4$ m bo'lgan va asoslarining radiuslari mos holda $R_1=2.4$ m va $R_2=4.2$ m ga teng bo'lgan kesik konus shaklidagi dudli quvur to'la sirtining yuzini toping:

$$S = \pi(R_1 + R_2)H + \pi R_1^2 + \pi R_2^2$$

6. Parallelepiped shaklidagi to'sinning qirralari uzunliklari mos holda $a=1.4$ m; $b=1.8$ m va $c=8.6$ m ga teng ekanligi ma'lum bo'lsa, to'sinning xajmi va diagonalini hisoblang:

$$V = abc, d = \sqrt{a^2 + b^2 + c^2}.$$

7. Kesimi aylana shaklidagi to'sinning kichik va katta diametrlari mos holda $d=2.4$ m va $D=3.8$ m ga teng bo'lsa, uning kesim yuzini toping:

$$S = \frac{\pi}{4} (D^2 - d^2).$$

8. To'sinning yuqori qismidan tortilgan arqonning uchlariga $m_1=86.7$ kg va $m_2=61.2$ kg massali yuk osilgan bo'lsa, arqonning tarangligi T ni aniqlang:

$$T = 2m_1m_2g/m_1m_2$$

bunda g -erkin tushish tezlanishi, ya'ni $g=9.81$ m/s².

9. Aylanish o'qiga nisbatan m massali qattiq jismning j-inertsiya momenti quyidagi munosabatlar bilan bog'langan:

a) R radiusli bir jinsli to' da silindr uchun:

$$J = \frac{1}{2} mR^2$$

b) ichki va tashqi radiuslari R_1 va R_2 bo'lgan g'ovak silindr uchun inersiya momenti:

$$J = \frac{1}{3} m(R_1^2 + R_2^2)$$

Agar har ikkala bog'lanish uchun inersiya momenti $J=1,4 \cdot 10^{-2}$ kg/m² va massasi $m=5.23$ kg ekanligi ma'lum bo'lsa, qattiq jismdan yasalgan to'la va g'ovak silindrlarning radiuslarini toping, bunda $R_1=2R_2$ deb oling.

10. Mingbuloq neft konida bir-biridan alohida ravishda ikkita quduq qazila boshlandi. Agar asosiy neft manbaiga nisbatan bu quduqlarning chuqurligi eng quyi nuqtasi koordinatalari ma'lum

bo'lsa, quduqning chuqurligi, ya'ni eng quyi nuqtasigacha bo'lgan masofanitoping,

bunda : A(2020 ; 1016 ; 2046) ; B(1806 ; 1209 ; 3024)

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2 + (z_2 - z_1)^2}$$

11. Uchburchak shaklidagi to'sinning tomonlari uzunliklari mos holda $a=2.4$ m; $b=4.8$ m va $c=3.6$ m ga teng ekanligi ma'lum bo'lsa har bir tomonga tushirilgan balandliklarni toping:

$$h_a = (2/a) * S; \quad h_b = (2/b) * S; \quad h_c = (2/c) * S;$$

bunda $S = \sqrt{(p-b)(p-c)(p-a)}$, $p = (a+b+c)/2$.

12. Uchburchak shaklidagi to'sinning tomonlari uzunliklari mos holda $a=2,4$ m; $b=4,8$ m va $c=3,6$ m ga teng ekanligi ma'lum bo'lsa, uning medianasini berilgan tomonlari yordamida toping.

$$m_a = \sqrt{2b^2 + 2c^2 - a^2} / 2 \quad m_b = \sqrt{2a^2 + 2c^2 - b^2} / 2 \quad m_c = \sqrt{2a^2 + 2b^2 - c^2} / 2$$

13. Uchta moddiy nuqtaning massalari ($m_1=0,8$; $m_2=0,4$; $m_3=0,09$) va mos holda koordinatalari [$C_1(1;2)$; $C_2(3;2)$; $C_3(4;3)$] ma'lum bo'lsa, moddiy nuqtaning og'irlik markazi koordinatalarini toping:

$$x_c = (m_1x_1 + m_2x_2 + m_3x_3) / m; \quad y_c = (m_1y_1 + m_2y_2 + m_3y_3) / m; \quad m = m_1 + m_2 + m_3$$

14. Uchburchak shaklidagi to'sinning tomonlari uzunliklari mos holda $a=2,6$ m; $b=3,4$ m; $c=2,4$ m ga teng ekanligi ma'lum bo'lsa, gradus o'lchov birligida uning burchaklarini toping.

$$\cos A = \frac{b^2 + c^2 - a^2}{2bc}; \quad \sin B = \frac{b \sin A}{a}; \quad c = 180 - (A+B)$$

15. Uchburchak shaklidagi to'sinning ikkita tomoni va ular orasidagi burchagi ma'lum bo'lsa (ya'ni $a=2,4$ m, $b=3,6$ m va $\angle c=45^0$), uning uchinchi tomoni va yuzini toping

$$c^2 = a^2 + b^2 - 2ab \cos C$$

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

bunda

$$p = \frac{a+b+c}{2}$$

16. Uchburchak shaklidagi to'sinning ikkita tomoni va ulr orasidagi burchagi ma'lum bo'lsa (ya'ni $a=1.3$ m, $b=2.2$ m va $c=\frac{\pi}{2}$), uning uchinchi tomoni A va B burchaklarini (radian o'lchov birligida) hamda yuzini toping.

$$\sin A = \frac{a \sin C}{c}, \quad \sin B = \frac{b \sin C}{c}, \quad \sin C = \frac{c \sin C}{c}$$

$$c^2 = a^2 + b^2 - 2ab \cos C.$$

17. Uchburchakning bitta tomoni va ikkita burchagi ma'lum bo'lsa, (ya'ni, $a=2,9$ sm, $\angle B=45^\circ$ va $\angle C=60^\circ$) shu uchburchakka tashqi va ichki chizilgan aylana radiusi va A burchagini toping:

$$\angle A = 180^\circ - \angle B - \angle C \quad R = \frac{a}{2 \sin \angle A} \quad r = 4R \sin \frac{\angle A}{2} \sin \frac{\angle B}{2} \sin \frac{\angle C}{2}$$

18. Uchburchakning bitta tomoni va uchala burchagi ma'lum bo'lsa, ya'ni $a=2,7$ sm, $\angle A=60^\circ$, $\angle B=90^\circ$, $\angle C=30^\circ$, b va c tomonlarini xamda shu uchburchakka tashqi chizilgan aylana radiusini toping:

$$b = \frac{a \sin \angle B}{\sin \angle A}; \quad c = \frac{a \sin \angle C}{\sin \angle A}; \quad R = \frac{c}{2 \sin \angle C}$$

19. Uchburchakning bitta tomoni va uchta burchagi ma'lum bo'lsa, (ya'ni $c=3,4$ sm va $\angle A=45^\circ$; $\angle B=45^\circ$; $\angle C=90^\circ$), uning yuzi, ichki chizilgan aylana radiusi va yarim perimetrini toping.

$$S = \frac{c^2 \sin \angle A \sin \angle B}{\sin \angle C} \quad r = c \frac{\sin \frac{\angle A}{2} \sin \frac{\angle B}{2}}{\cos \frac{\angle C}{2}} \quad p = \frac{S}{r}$$

i

20. Uchburchakning uchta tomoni ma'lum bo'lsa ($a=2.6$ sm, $b=2.8$ sm, $c=4.8$ sm) unga icjki chizilgan aylana radiusi va A burchagini toping:

$$r = \sqrt{\frac{(p-a)(p-b)(p-c)}{p}}, \quad \sin \frac{A}{2} = \sqrt{\frac{(p-b)(p-c)}{bc}}$$

6 -amaliyot mashg'uloti uchun test:

1. ESPD nima?
 - A. dasturiy xujjatlarni avtomatlashtirish
 - B. shaklini aniqlab beruvchi standartlar
 - * C. yagona dasturiy xujjatlar sistemasi
2. Operatorga maxsus ko'rsatmalar qaysi ko'rsatkichlarda izohlanadi?
 - *A. kiritish B. chiqarish C. dasturga ko'rsatmalar
3. Dastur kutubxonalari qanday maqsadda qo'llaniladi?
 - A. Dasturni sifatini oshirish, sozlash va tekshiruv murakkabligini oshirish xamda dastur ko'lamini oshirish uchun.
 - *B. Dasturni samaradorligini oshirish, sozlash va tekshirishni yengillashtirish xamda dastur yaratishni qisqartirish uchun.
 - C. Dasturni xaqidagi ma'lumotlarni saqlash, sozlash va tekshirishni amalgam oshirish xamda dastur yaratishni yengillashtirish uchun.
4. Dasturni kompanovkalash va tashqi aloqalarni taxrirlash deganda nimani tushunasiz?
 - A. dasturni sifatini oshirish
 - B. dasturni xajmini kamaytirish

* C. sintaktik xatolarni aniqlash

5. Dasturlarni ko'chirish nima uchun kerak?

*A. Ko'chirish samarasi va tez sozlanuvchi dasturni tayyorlashga yordam beradi.

B. Algoritm tekshirishda foydalanish uchun.

C. Model dasturining tili bir xilda bo'lishi uchun.

6. Dasturlashda nechta sozlash vositalari qo'llaniladi?

A. 9 ta

B. 3 ta

* C. 6 ta

7. Xar bir muomaladagi yozuvga kichik dasturni nomi chiqadi, undan chiqilganda-asosiy dasturga qaytilganligi xaqida ma'lumot chiqadi.

Bu turdagi tekshirish dasturchiga dasturiy xatolarni aniqlashdagi barcha kerakli ma'lumot bilan ta'minlab turadi. Uning kamchiligi nimada ?

A. dasturni qisimlarga ajratib belgilangan joylarda o'chirib yoqib turilishidadir.

*B. mashina vaqtini ko'p sarf bo'lishidir, xamda berilayotgan ma'lumotlar bir necha ming qatordan iborat bo'lishidadir.

C. O'zgaruvchilarni belgilarini tasvirlashdadir.

8. I.S xujjatining standart ko'rinishi ketma-ketligini ko'rsating.

A. 1. Taktika

2. Maqsadlar

3. Dastur maxsulotning tavsifi

B. 1. Maqsadlar

2. Taktika

3. Dastur maxsulotning tavsifi

* C. 1. Dastur maxsulotning tavsifi

2. Maqsadlar

3. Taktika

Javob : 1. C 2.A 3.B 4. C 5.A 6. C 7.B 8. C

3.7. DASTURLASH TEXNOLOGIYASINI TESTLASH. TESTLASH PRINSIPLARI VA USULLARI.

Reja:

1. Testlash;
2. Testlash bosqichlari;
3. Testlash printsiplari.

Ishdan maqsad: Testlash prinsiplari va usullarini o'rganish hamda amaliy ko'nikma hosil qilish.

Nazariy qism

Testlash – bu dasturiy maxsulotning ishlash jarayonida ishlamaydigan qismini topishdir. Testlash dasturiy maxsulotning xatoliklarini tuzatishdagi asosiy usul hisoblanadi. Dasturni qayta tayyorlash dasturiy maxsulotning yetishmovchiligini to'ldirish jarayonidir. Agar dastur anchagina testlar to'plamidan keyin ham to'g'ri natijalarni chiqarib bersa, bu dasturda xato yo'q degan asos bo'lishi kerak emas. Bu paytda dasturni to'g'rilik darajasida gapirish mumkin.

Test – bu oldindan hisoblangan oraliq va oxirgi natijalari ma'lum bo'lgan dasturning to'g'riligini nazorat qiluvchi vositadir. Testni shunday tanlab olish kerakki, dasturchi natijani testni ishlatishdan oldin hisoblab bilsin.

Testlash jarayoni quyidagi bosqichlarga bo'linadi:

1. Normal sharoitda tekshiruv bosqichi;
2. Ekstrimal sharoitda tekshiruv bosqichi;
3. Ayrim sharoitda tekshiruv bosqichi.

Testlash printsiplari quyidagicha ko'rinishga ega:

1. Dasturiy maxsulotni ishlanmasligini testlashni asosiy masalasi deb hisoblab, dasturchi yozgan dasturini testlash maqsadga muvofiq emas;

2. Dasturni to'g'riligini ko'rsatadigan emas, balki xatolarni topish extimoli katta bo'lgan test;

3. Testlarni to'g'ri va to'g'ri bo'lmagan qiymatlar uchun tayyorlang;

4. Testlarni kompyuterda o'tkazishni hujjatlashtiring, har bir testning natijasini maydalab o'rganing;

5. Har bir modulni dasturga faqat bir marta ulang. Testlashni osonlashtirish uchun xech qachon dasturni o'zgartirmang.

6. Dasturiy maxsulotni tekshirish bilan bog'liq bo'lgan testlarni qayta o'tkazish yoki dasturiy maxsulotni ishlatish bilan va agar ko'p o'zgartirishlar kiritilgan bo'lsa, boshqa dasturlar bilan bog'liqligini tekshirish bilan bog'liq bo'lgan testlarni qayta o'tkazing.

Quyidagi masalaga testlar yozing. Parallelipiped diagonalining uzunligi formulasi bo'yicha topiladi. Shu masalaga normal, ekstremal va ayrim hollar uchun test yozing.

Parallelipiped tomonlari			Izoh
1	1	1	Yaxshi testlar $d=1.7320508$
1	2	3	Normal test $d=3.74165773$
0	0	0	Natija 0 ga teng bo'lishi kerak
0	1	2	Bu parallilipiped emas. Nima bo'ladi?
1	0	3	Noto'g'ri qiymatlar
2	1	0	

Normal xolat

Ekstrim xolat

Ayrim xolatlar

1	-6	3	
A	V	S	

Nazorat savollari :

1. Testlash qanday ma'noni anglatadi?
2. Testlash bosqichlarini so'zlab bering.
3. Testlash printsiplari haqida so'zlab bering.
4. Normal sharoitda tekshiruv nima?
5. Ekstrimal sharoitda tekshiruvni tushuntirib bering.
6. Ayrim sharoitda tekshiruv deganda nimani tushunasiz?
7. Testlash prinsipining 2-,5- xamda 6-larini aytib bering.
8. Testlash prinsipining 1-,4- xamda 5-larini aytib bering.

Tayanch iboralar: testlash, test, testlash printsiplari.

7-topshiriq tavsifi

7 – topshiriqda dasturchi dasturlash texnologiyalarini yaxshi o`rgangan bo`lishi va class, metod, svoystva, event siklik jarayonlar, logik funksiyalarni va murakkab matematik, trigonometrik amallarni hisoblashni bilishi kerak. Bundan tashqari Graphics metodlarini ishlatishni va to`g`ri chiziqni chizishni bilishi kerak.

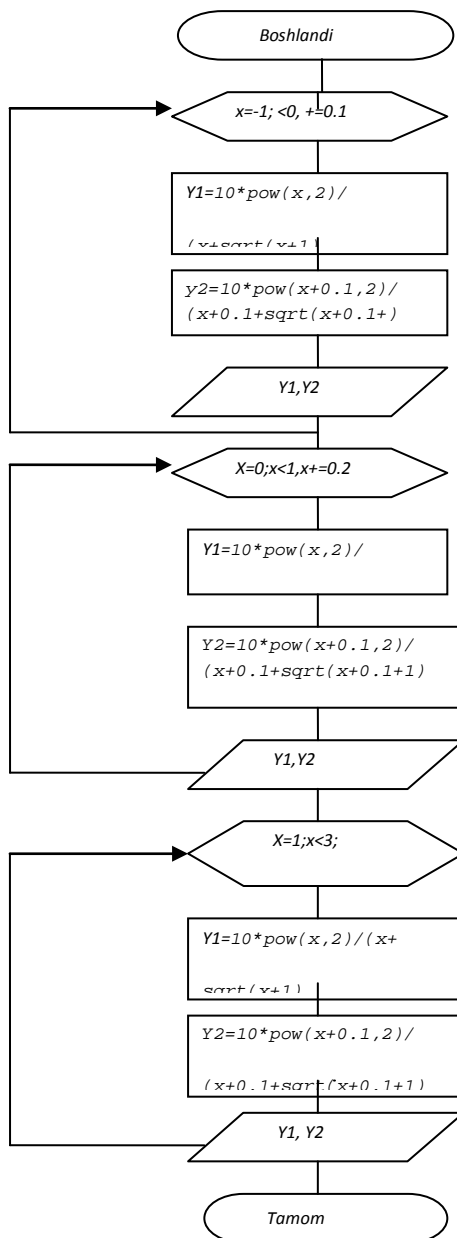
7 – topshiriqdagi 1 – misolni misol uchun olamiz:

$$Y=x^2 \cos x \quad [0;\pi/2] \text{da } \Delta x = \pi/12; \quad [\pi/2; \pi] \text{da } \Delta x = \pi/10; \quad [\pi; 2\pi] \text{da } \Delta x = \pi/4$$

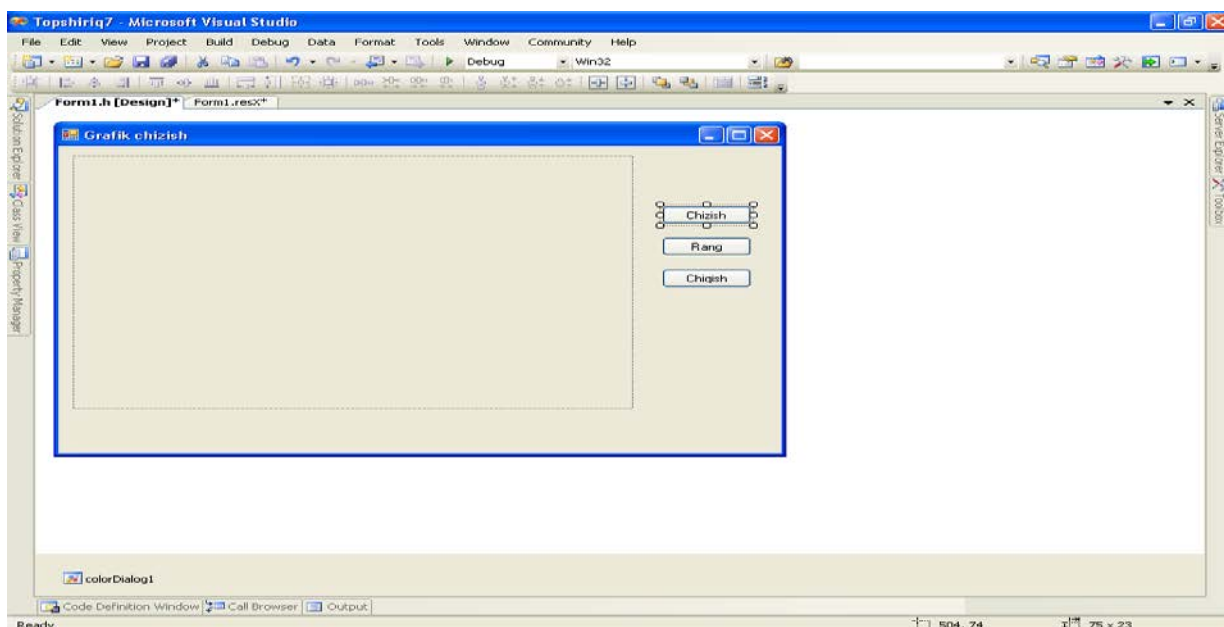
Berilgan funksiyani- x ni berilgan oroliqida hisoblab uning grafigini chizish kerak. Buni **Graphics class**idan foydalanib chiziladi, chizishda to`g`ri chiziq chizish metodidan foydalaniladi. Y ning qiymatini x sonining qiymat oralig`idagi

qiyamatlarni qo`yib hisoblaymiz va chiqqan natijani to`gri chiziq chizish metodining parametrlarinig o`rniga qo`yamiz va chiqqan chizmani ekranga chiqaramiz. X ning qiymatini masalani berilishidagi shartga ko`ra o`zgartiramiz. Endi yuqoridigilar asosida dasturning blok sxemasini chizaylik.

Blok
sxema:



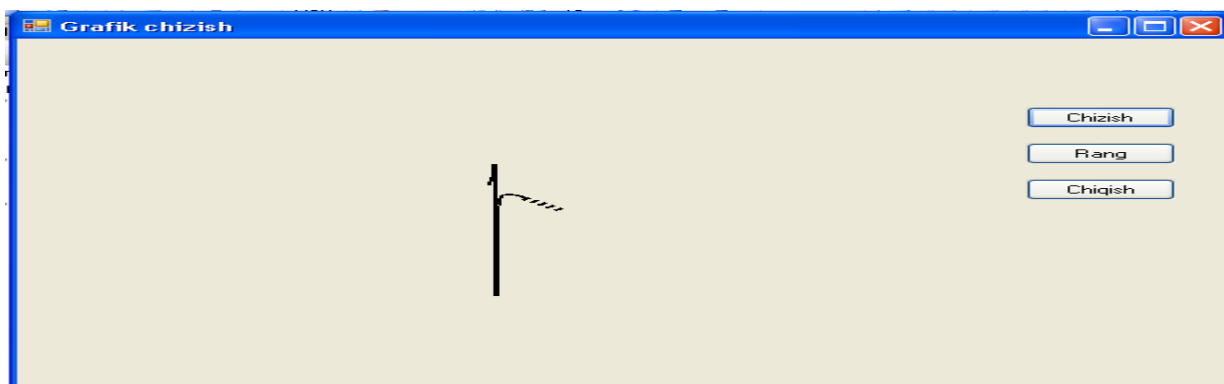
Dasturiy qismini yozishdan avval uning ko`rinishini bizga ma'qul kelgan holda yoki o`qituvchining talabiga ko`ra tuzamiz (1 – rasm)



1 – rasm

Ko`rinishini tuzib bo`lgandan keyin dasturiy qismini yozamiz.

Dasturiy qisimni yozib bo`lgandan keyin dasturni kompilyatsiya qilib ishga tushiramiz va 2 – rasmdagi oyna ochiladi. Ochilgan oynada rangni tanlash va funktsiyaning grafigini chizish tugmasi bor. Funktsiyaning chizish tugmasini bossak ekranda grafik hosil bo`ladi.(2 – rasm)



2 – rasm

7-amaliy mashg'ulotga topshiriqlar:

Variants tartibi	Funksiya	Oraliqlar va qadamlar
1	2	3
1	$y = 2x^3 + \sqrt[3]{x+1}$	$[-2;0]$ da $\Delta x = 0,2$; $[0;1]$ da $\Delta x = 0,1$; $[2;5]$ da $\Delta x = 0,5$;
2	$y = \ln \sqrt[3]{x^2 + 4x + 2}$	$[0;1]$ da $\Delta x = 0,1$; $[2;4]$ da $\Delta x = 0,5$; $[5;8]$ da $\Delta x = 1$
3	$y = x^2 \cos x$	$[0;\pi/2]$ da $\Delta x = \pi/12$; $[\pi/2; \pi]$ da $\Delta x = \pi/10$; $[\pi; 2\pi]$ da $\Delta x = \pi/4$
4	$y = \arctg \sqrt{x^2 + 1}$	$[-1;0]$ da $\Delta x = 0,2$; $[0;0,5]$ da $\Delta x = 0,1$; $[0,5;1]$ da $\Delta x = 0,05$;
5	$y = 2,3^{x-\sqrt{x-1}} - \sqrt{x}$	$[-1;1]$ da $\Delta x = 0,2$; $[1;2]$ da $\Delta x = 0,1$; $[2;3]$ da $\Delta x = 0,4$;
6	$y = 2x^2 + \sin^3 x$	$[-\pi;\pi/2]$ da $\Delta x = \pi/16$; $[\pi/2; 0]$ da $\Delta x = \pi/4$; $[\pi; \pi/2]$ da $\Delta x = \pi/8$
7	$y = \arcsin^2 x + e^x$	$[-1;0]$ da $\Delta x = 0,4$; $[0;0,5]$ da $\Delta x = 0,1$; $[0,5;1]$ da $\Delta x = 0,05$
8	$y = e^{\sqrt[3]{x-1}}$	$[-5;0]$ da $\Delta x = 0,5$; $[0;2]$ da $\Delta x = 0,5$; $[3,6]$ da $\Delta x = 0,3$
9	$y = e^x \sin^2 x$	$[-1;0]$ da $\Delta x = 0,2$; $[0;2]$ da $\Delta x = 0,4$; $[4;7]$ da $\Delta x = 0,5$;
10	$y = 3x^3 + \sqrt{x+1}$	$[-2;0]$ da $\Delta x = 0,4$; $[0;2]$ da $\Delta x = 0,5$; $[3;7]$ da $\Delta x = 0,4$;
11	$y = x^3 + \sin x$	$[0;1]$ da $\Delta x = 0,2$; $[1;2]$ da $\Delta x = 0,4$; $[2;5]$ da $\Delta x = 0,5$;
12	$y = \sqrt{x^2 + 4x + \ln x}$	$[2;4]$ da $\Delta x = 0,4$; $[4;6]$ da $\Delta x = 0,5$; $[6;7]$ da $\Delta x = 0,2$;
13	$y = \frac{\sin x}{x}$	$[1;2]$ da $\Delta x = 0,2$; $[0;1]$ da $\Delta x = 0,4$; $[4;6]$ da $\Delta x = 0,5$;
14	$y = x^2 - 1,5x $	$[-1;1]$ da $\Delta x = 0,4$; $[1;2]$ da $\Delta x = 0,2$; $[2;5]$ da $\Delta x = 0,5$;

15	$y = tge^{x+1}$	[-1;0]da $\Delta x = 0,4$; [0;0,5]da $\Delta x = 0,1$; [0,5;1]da $\Delta x = 0,05$;
16	$y = \sin^2 x + 1,5^x$	[-1;0]da $\Delta x = 0,2$; [0;1]da $\Delta x = 0,1$; [1;4]da $\Delta x = 0,5$;
17	$y = \sqrt{x^2 + 3x^3} + \ln x$	[2;3]da $\Delta x = 0,5$; [3;5]da $\Delta x = 0,4$; [5;7]da $\Delta x = 0,5$;
18	$y = \cos^2 x^3 + 1 $	[2;0]da $\Delta x = 0,4$; [2;3]da $\Delta x = 0,2$; [3;6]da $\Delta x = 0,5$;
19	$y = e^{x^{2+4,5x}}$	[-1;0]da $\Delta x = 0,2$; [0;0,5]da $\Delta x = 0,05$; [0,5;1]da $\Delta x = 0,1$;
20	$y = \sqrt{\sin^2 4x} e^x$	[-2;0]da $\Delta x = 0,4$; [0;1]da $\Delta x = 0,2$; [1;2]da $\Delta x = 0,1$;

7 -amaliyot mashg'uloti uchun Test

1. Testlash qanday ma'noni anglatadi?

A. dasturni barcha qismini birma –bir taxlil qilish jarayoni.

*B. dasturiy maxsulotning ishlash jarayonida ishlamaydigan qismini topishdir.

C. dasturiy maxsulotni ishlash jarayonini kuzatish.

2. Test nima ma'noni anglatadi?

A. bu oldindan xisoblangan oraliq va oxirgi natijalari noma'lum bo'lgan dasturning to'g'riligini nazorat qiluvchi vositadir.

*B. bu oldindan xisoblangan oraliq va oxirgi natijalari ma'lum bo'lgan dasturning to'g'riligini nazorat qiluvchi vositadir.

C. bu yakuniy xisoblanuvchi oraliq va oxirgi natijalari noma'lum bo'lgan dasturning to'g'riligini nazorat qiluvchi vositadir.

3. Testlash jarayoni bosqichlari ketma-ketligini to'g'ri ko'rsating.

A. 1. Ekstrimal sharoitda tekshiruv

2. Normal sharoitda tekshiruv

3. Ayrim sharoitda tekshiruv

*B. 1. Normal sharoitda tekshiruv

2. Ekstrimal sharoitda tekshiruv

3. Ayrim sharoitda tekshiruv

C. 1. Ayrim sharoitda tekshiruv

2. Ekstrimal sharoitda tekshiruv

3. Normal sharoitda tekshiruv

4. Testlash prinsiplari nechta?

*A. 6 ta

B. 9 ta

C. 10 ta

5. Testlash prinsipining 6-sini ko'rsating.

A. Testlarni kompyuterda o'tkazishni xujjatlashtiring, xar bir testning natijasini maydalab o'rganing.

B. Dasturni to'g'riligini ko'rsatadigan emas, balki xatolarni topish extimoli katta bo'lgan test.

* C. Dasturiy maxsulotni tekshirish bilan bog'liq bo'lgan testlarni qayta o'tkazish yoki dasturiy maxsulotni ishlatish bilan va agar ko'p o'zgartirish kiritilgan bo'lsa, boshqa dasturlar bilan bog'liqligini tekshirish bilan bog'liq bo'lgan testlarni qayta o'tkazib yuboring.

6. Testlash prinsipining 3-sini ko'rsating.

*A. Testlarni to'g'ri va to'g'ri bo'lmagan qiymatlar uchun tayyorlang.

B. Dasturiy maxsulotni ishlanmasligini testlashni asosiy masala deb xisoblab, dasturchi yozgan dasturni testlash maqsadga muvofiq emas.

C. Dasturni to'g'riligini ko'rsatadigan emas, balki xatolarni topish extimoli katta bo'lgan test.

7. Testlash prinsipining 1-sini ko'rsating.

A. Testlarni to'g'ri va to'g'ri bo'lmagan qiymatlar uchun tayyorlang.

B. Dasturni to'g'riligini ko'rsatadigan emas, balki xatolarni topish extimoli katta bo'lgan test.

* C. Dasturiy maxsulotni ishlanmasligini testlashni asosiy masala deb xisoblab, dasturchi yozgan dasturni testlash maqsadga muvofiq emas.

8. Testlash prinsipining 4-sini ko'rsating.

A. Dasturiy maxsulotni ishlanmasligini testlashni asosiy masala deb xisoblab, dasturchi yozgan dasturni testlash maqsadga muvofiq emas.

B. Testlarni to'g'ri va to'g'ri bo'lmagan qiymatlar uchun tayyorlang.

* C. Testlarni kompyuterda o'tkazishni xujjatlashtiring, xar bir testning natijasini maydalab o'rganing.

Javoblari: 1-B 2-B 3-B 4-A 5- C 6-A 7- C 8- C

Dastur kodi BMIning ilova qismi 7-ilova bandida keltirilgan.

8.1. DASTURIY TA'MINOTNI ISHLATISH. ISHLATISHGA KO'RSATMALAR.

Reja:

1. DTning ekspluatatsiya (ishlatish) jarayoni;
2. Ishlatishga ko'rsatmalar;
3. Dasturiy maxsulotga qo'yiladigan talablar;
4. Dastur hujjatlariga talablar.

Ishdan maqsad: Dasturiy ta'minotni ishlatish va amaliy ko'nikma xosil qilish.

Nazariy qism

Dasturiy taminotni tatbiq qilishda kuzatish, xodimlarni o'qitish va zarur bo'lgan paytda maxsulotni kengaytirish jarayoni o'tkaziladi. Agarda oldingi bosqichlar to'g'ri va sifatli o'tkazilgan bo'lsa, yakuniy jarayon xatosiz o'tadi.

Dasturiy taminot ekspluatatsiya (ishlatish) jarayoni quyidagilardan iborat:

1. Dasturni kompyuterga kiritib, kirish malumotlari ustada amallar bajarish;
2. Kutiladigan natijalarni olish;
3. Dasturiy ta'minot ishonchliligini ta'minlash.

Dasturiy ta'minotni kuzatib borish jarayoni quyidagilardan iborat:

1. Ekspluatatsion xizmat qilish;
2. Dasturiy ta'minot zarur bo'lgan xarakteristikalarini ta'minlash;
3. Dasturiy ta'minotni hr xil hsoblash tizimlariga o'tkazish va moslashtirish.

Dasturiy ta'minotni kuzatib borish jarayoniga quyidagi o'zgarishlarni kiritish lozim:

1. Xatolarni tuzatish;
2. Dasturiy ta'minotda ishlatiladigan apparatura konfiguratsiyasiga moslashtirish;
3. **Modernizatsiya** – dasturni funktsional xossalarini yaxshilash.

<i>Xatolarni tuzatish 20%</i>
<i>Modernizatsiya 60%(yangilash)</i>
<i>Adaptatsiya 20% (Moslashuvchanlik)</i>

Dasturiy ta'minotni ishlatish bo'yicha ko'rsatmalar.

Maktab o'quvchilari uchun mo'ljallangan bir argumentli grafikdan funktsiyaga dasturiy maxsulotga tayangan holda texnik topshiriq tayyorlansin, yozilgan dasturda hisoblash jadvali bo'lishi va berilgan soxadan berilgan formulalardan argumentni qadamini va soxa chegaralarini o'zgartirgan xolda funktsiyaning grafigi chizilishi lozim undan tashqari dastur yangi kiritgan formulalarni eslab qolishi kerak.

1.Kirish qismi:

Bu texnik topshiriq 1 ta o'zgaruvchi f-ya qiymatlari jadvalini va grafigini chizish dasturiga tegishli bo'lib yuqori sinfdagi maktab o'quvchilariga mo'ljallangan.

Maktabning elementar algebra kursidan f-yani olamiz, yechish qiyin bo'lgan masalalardan biri, quyidagi masalani o'rganish jaryonida o'quvchilar bir o'zgaruvchi argumentni grafigini chizishi kerak . Bu jarayonda ular funktsiyaning berilgan xarakteristik nuqtalarini ildizlarini va 1-2 uzilish nuqtalarini va boshqa ma'lumotlarni ishlatishlari kerak. Bunday dasturlarni ishlatishda eng universal dasturiy ta'minotlar bor albatta. Lekin uning interfeysi foydalanuvchi uchun juda qiyinlik qiladi.

Ularni tushunish uchun kamida oliygox talabalari o'rganadigan oliy matematikadan habardor bo'lishlari ham kamlik qiladi. Bu esa maktab o'quvchilari uchun emas albatta. Ishlab chiqariladigan maxsulot maktab o'quvchilariga yuqoridagi mavzu bo'yicha o'z bilimlarini almashishga yordam beradi.

2. Ishlab chiqarish uchun sabab. DT kafedra bilan maktab o'rtasidagi shartnoma asosida ishlab chiqariladi.

3. Dasturiy maxsulotga qo'yiladigan talablar.

Funksional xarakteristikalariga qo'yiladigan talablar:

1. Dastur quyidagi funktsiyalarni bajara olishi kerak:

A) bir o'zgaruvchi funktsiyaning analitik ko'rinishi kiritish va uzoq vaqt davomida uni sistemada saqlab qolish;

B) funktsiyani aniqlash intervalini kiritish va o'zgartira olish;

V) argument qoidalarini kirita olish va uni o'zgartira olish;

G) ko'rsatilgan oraliqda «funktsiya uzilish nuqtalariga ega emas» shartiga asosan shu intervalda funktsiya qiymatlar jadvalini yoki grafigini chizish.

4. Boshlangich ma'lumotlar:

A) funktsiyaning analitik qiymatlari;

B) funktsiyani olinish nuqtalariga son orasidagi argumentni o'zgartirish qadami

5. Ishonchlilik talablari :

1. Kiritilayotgan ma'lumotlarni tekshirish;

2. Tizim va dastur bilan ishlayotganda foydalanish.

6. Shaxsiy kompyuterlarda ishlash imkoniyati :

Minimal konfiguratsiya:

1. Protsessor -pentium va undan yuqori;

2. Operativ xotira 32 mb va undan yuqori.

7 Axborot va dasturiy moslashuvchanlik :

Tizim quyidagi operatsion sistemali boshqarishda ishlashi shart.

-WINDOWS 98.2000.

8. Dastur hujjatlariga talablar:

1. "Professional dasturchi har bir dasturga kerakli modullarni yozadi"- VAN KASSEL .Dastur tekstni va hamma kerakli narsalarni o'z ichiga olishi kerak;

2. Ishlab chiqarilayotgan dastur asosiy matematik terminlar va o'quvchilar uchun va o'quvchilar uchun ko'rsatmalar va yordamchi ma'lumotlarga ega bo'lishi kerak;

3. Yordamchi hujjatlarga quyidagilar kiradi:

- 1) Tushuntirish xati 30-40 varaqda bo'lishi;
- 2) Sistemali dasturchiga ko'rsatma yozilishi;
- 3) Foydalanuvchiga ko'rsatma berilishi;
- 4) Dasturiy tizim strukturaviy sxemasi chizilishi;
- 5) Ma'lumotlar diagrammasi ishlab chiqilishi ;
- 6) Foydalanuvchi interfeysi ko'rinishlari mavjud bo'lishi kerak.

Ichki spetsifikatsiya hujjati:

Ichki spetsifikatsiya hujjati texnik topshiriq va tashqi spetsifikatsiya hujjatlari mavjud bo'lishini taqozo qiladi. Chunki texnik topshiriq va tashqi spetsifikatsiyalarda yaratilgan dasturiy maxsulotlar haqida to'liq axborot keltiriladi. Yuqoridagilardan farqli o'laroq ichki spetsifikatsiya hujjati maxsulotni qay darajada tuzilganligiyu unga qo'yilgan talab va maqsadlarni akslantiradi. Bu hujjatdan foydalanganda dasturning u tomonidan to'liq qabul qilishi shart emas ya'ni bu hujjat dasturchi uchun.

I. S xujjati dastur maxsuloti yaratish jarayonini akslantirib xar bir bosqich ishi tugallangandan so'ng yoziladi. Juda ko'p hollarda bu hujjat **ichki loyixa** nomiga ham ega. Bu hujjatni dasturlash bosqichini boshlashdan avval rasmiylashtirish tavsiya qilinadi. Shunga ko'ra bu hujjatda dastur maxsulotini yaratish uchun lozim bo'lgan barcha algoritmlarning tavsifi kengaytiriladi. Dasturlash bosqichi ichki spetsifikatsiyalar hujjati rasmiylashtirilib bo'lgandan song boshlanadi.

Bu vaqtga kelib barcha algoritm kattaliklarning to'liq tavsifi mavjud bo'ladi. Blok sistemalar esa shu darajada mukammal ravishda yaratilgan bo'ladiki ular asosida yozilgan dastur xatolikka ega bo'lamaydi. I.S Xujjatning asosiy vazifasi dasturiy maxsuloti haqidagi barcha ma'lumotlarni berishdan tashkil topgan bo'lib unda loyixani ichki mazmuni algortimlari va kelgusida ishlatishga foydali bo'lgan modullar haqidagi tushuncha keltiriladi. I.S xujjatining standart ko'rinishi quyidagicha:

1. Dastur maxsulotning tavsifi;
2. Maqsadlar;

3. Taktika.

Bu bo'limda ham yuqoridagi hujjatlarda keltirilgan axborot birgalikda yangi terminlar va sintaksik qoidalar haqida tushuncha keltiriladi.

Ichki kattaliklar . Ayrim modullarda kirish va chiqish parametrlari hisoblangan kattaliklarga ishlov berish ko'zda tutiladi. Bunday kattaliklar ichki kattaliklar deb hisoblanib ularning tavsifi aynan shu bo'limda keltiriladi. Bu bo'lim qanchalik to'liq bo'lsa, yaratilayotgan dastur shunchalik mukammal bo'ladi.

Tadbiq qilish strategiyasi .Bu bo'limda tadbiq qilish rejasi keltiriladi. Agar modullarning vazifasi odatdagi standart vazifalar bilan mos tushmasa u holda modullargagina hos bo'lgan belgilashlar keltiriladi. Bu bo'limda DV ni rostlashni tashkil qilishda umumiy ko'rsatmalar beriladi. Lekin oldin quyidagi ishlanmaning boshlangich bosqichlarida xatolarni topish zarurligini tasdiqlovchi favquloddagi hodisani ta'kidlab o'tishimiz kerak: DV da aniqlangan va tuzatilgan xatolarning soni ortgan sari, undagi topilmagan xatolar borligining nisbiy extimoli ham oshadi. Bu hodisani quyidagicha tushunish kerak. DV da topilgan xatolarning soni ortgan sari, bizda yo'l qo'ygan xatolarimiz haqida xali topilmagan xatolar soni to'g'risida ham tasavvur paydo bo'ladi.

Quyida rostlashni tashkil qilishda ko'rsatmalar va qoidalar ko'rinishida keltiriladi.

Qoida 1. DV ni ishlanmasini testlashni asosiy masala deb hisoblang, uni yuqori malakali va iqtidorli dasturchilarga topshiring, o'zingizni dasturingizni testlash maqsadga muvofik emas.

Qoida 2. Dasturni to'g'riligini ko'rsatadigan emas, xatolarini topish extimoli katta bo'lgan test yaxshidir.

1-qadam Mumkun bo'lgan va mumkun bo'lmagan qiymatlarning har bir soxasi uchun bajariladi.

2-qadam. Ixtiyoriy bo'linishining har yili xech bo'lmaganda bir yilda o'tmishga ishonch hosil qilish uchun modul matnini tekshiring.

3-qadam. Har bir sikl uchun quyidagi uchta vaziyatni ta'minlaydigan testlarning mavjudligiga ishonch hosil qilish uchun, modul matnini tekshiring,

sikl jismi bir marta bajariladi va sikl jismi maksimal son marta bajariladi.

Foydalanishning yengilligi, ma'lum darajada, foydalanilayotgan hujjatlarning tarkibi va sifati bilan, shu bilan birga dasturlash yo'li bilan amalga oshiriladigan ba'zi xossalari orqali aniqlanadi. Foydalanilayotgan hujjatlashtirish, DV larning, P-hujjatlanganlik va axborotlanganlik deb ataluvchi, sifat primitivlari bilan bog'liqdir. Uning sifatini ta'minlash bilan texnik yozuvchilar shug'ullanadilar. Shu bilan bog'lik, hozirgi vaqtda keng qo'llaniladigan foydalanuvchini interaktiv rejimda axborotlashtirish yo'liga ahamiyat berishimiz kerak (DV dasturlarini qo'llash jarayonida). Bunday axborotlashtirish ko'p hollarda, foydalanuvchi uchun avtonom hujjatlashtirishga qaraganda qulay bo'ladi, chunki foydalanuvchiga hech qanday qidiruvsiz, kontekstini chaqirib zarur axborotni chaqirishga imkon beradi. Foydalanuvchini bunday axborotlashtirish, juda perspektiv hisoblanadi. DV sining kommunakabellilik, mustaxkamlilik va ximoyalanganlik sifat primitivlari dasturlash yo'li bilan amalga oshiriladi. Maxsus vaziyatlarni qayta ishlashni mos amalga oshirish va mos foydalanadigan interfeys yaratish yo'llari bilan kommunakabellilik ta'minlanadi. Ko'p hollarda maxsus vaziyatlarni qayta tiklashning ma'nosi, foydalanuvchini, dastur bajarilayotganligi to'g'risida, axborotlashtirish zaruriyati paydo bo'lganligini bildiradi. Bunda foydalanuvchiga berilayotgan axborotni tushunish uchun ishlatilishi sodda bo'lishi kerak. Ammo, maxsus vaziyatlar, dasturning modul tarkibini yetarli past darajasida namoyon bo'ladi, foydalanuvchiga tushunarli bo'lgan xabarni shu maxsus holat paydo bo'lishiga olib kelgan, harakatlar aktivlashtirilgan, kontekst ma'lum bo'lgan, tarkibning yuqori darajalarida tuzish mumkin. Boshqa modulda paydo bo'ladigan maxsus holatlarni qayta ishlashi uchun sodda bo'lmagan yechimlarni qabul qilish kerak. Paydo bo'lgan maxsus vaziyat haqidagi axborotni dastur modullariga uzatishning murojat tizimlari bo'yicha ko'p qo'llanadigan usuli juda dag'aldir: u moduldan qaytishda qo'shimcha tekshirishlarni talab qiladi va shu modulga murojatni, qo'shimcha parametrlar xisobiga qiyinlashtiradi. Dastur bajarilish operatsiyasi muxitiga shu axborotni, maxsus hollarni qayta ishlovchilariga, shunday qayta ishlovchilardan dinamik tuzilgan navbat bo'yicha to'g'ri uzatish esa

xotiraning umumiy sarfiga jiddiy ta‘sir ko‘rsatadi, bunda ular DV sining ish vaqtiga ko‘zga ko‘rinarli ta‘sir qilmaydi. Bundan tashqari, bu ta‘sirni(eng avval, vaqt effektivlikka nisbatan) oldindan (DV sini amalga oshirish tugaguncha) doim to‘g‘ri baxolay olmaymiz. Aytilganlarni xisobga olib DVsi effektivligini ta‘minlash uchun quyidagi prinsiplarga amal qilish tavsiya qilinadi.

- Oldin ishonchli DVsini ishlab chiqish kerak, keyin esa uning sifat spetsifikatsiyasiga mos talab darajasiga, uning effektivligini oshirish bilan shugullanish kerak ;

- DV si effektivligini oshirish uchun eng avval, optimallashtiruvchi kompilyatordan foydalanish kerak, bu talab qilinayotgan effektivlikni ta‘minlashi mumkin ;

- Agar DVni effektivligi uning spetsifikatsiyasini qanoatlantirmasa, talab qilinayotgan DVsi effektivligi nuqtai nazardan eng kritik modullarni toping, shu modullarni birinchi navbatda qo‘lda qayta ishlash yo‘li bilan optimallashtirishga harakat qiling ;

- Agar talab qilinayotgan DVsining effektivligiga erishishda talab qilinmasa, modulni optimallashtirish bilan shugullanmaslik kerak ;

- DVsining vaqt bo‘yicha effektivligi nuqtai-nazaridan, kritik modullarni qidirishi uchun DV bajarilishi vaqtida mos o‘lchamlar yo‘li bilan, DV si ishlash vaqtini modullar bo‘yicha taqsimotini olish talab qilinadi. Buni DV sini qo‘llash protsessida har bir modulga murojaat chastotasini aniqlay oladigan, dinamik analizator (maxsus dasturlash instrumenti) yordamida bajarsa bo‘ladi.

Nazorat savollari:

1. DTni ekspluatatsiya jarayoni nimalardan iborat?
2. DTni kuzatib borish jarayoni nimalardan iborat?
3. Modernizatsiya nima?
4. DTni ishlatish bo‘yicha qanday ko‘rsatmalarni bilasiz?
5. DTga qanday talablar qo‘yiladi?
6. Ichki kattaliklar haqida gapirib bering.

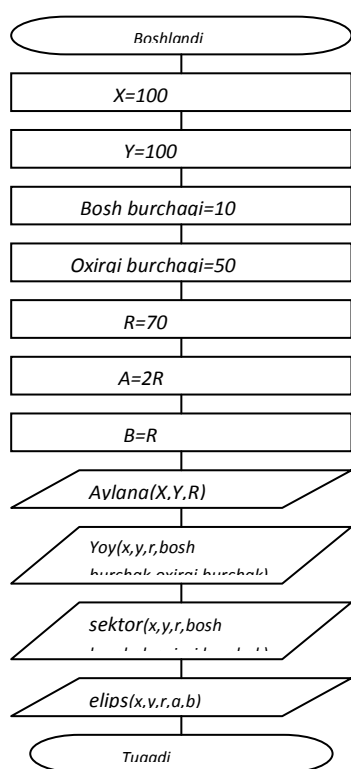
7. Tadbiriq qilish strategiyasi qanday?

8. Dastur hujjatlariga qanday talablar qo'yiladi?

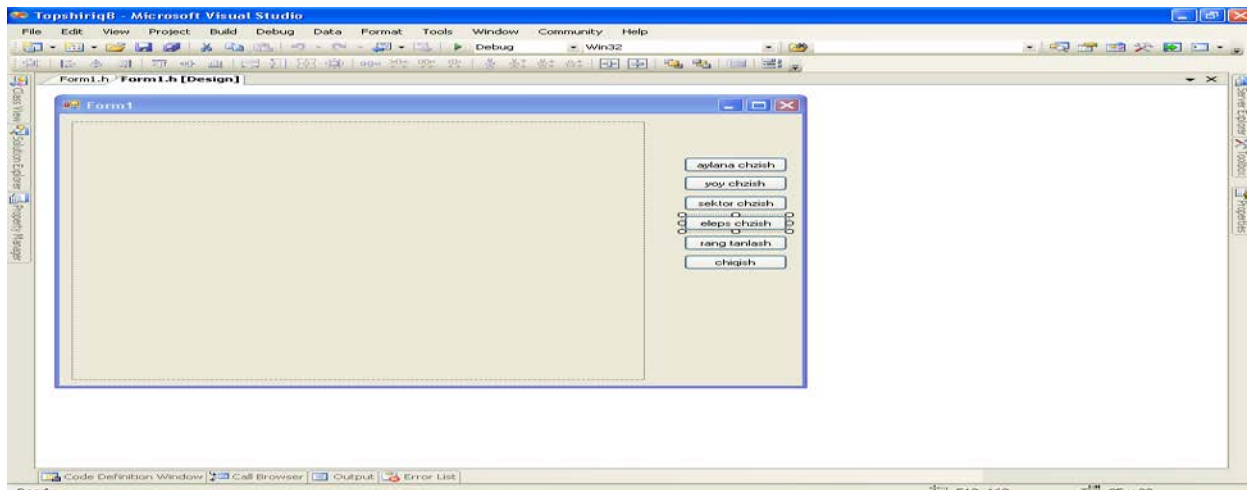
Tayanch iboralar: modernizatsiya, ishonchlilik talablari, ichki kattaliklar, tadbiriq qilish strategiyasi.

8-topshiriq taxlili

8 – topshiriqda dasturchi dasturlash texnologiyalarini yaxshi o`rgangan bo`lishi va class, metod, svoystva, event siklik jarayonlar, logik funksiyalarni va murakkab matematik, trigonometrik amallarni hisoblashni bilishi kerak. Bundan tashqari **Graphics** metodlarini ishlatishni va murakkab shakllarni chizishni bilishi kerak. 8 – topshiriqdagi 1 – misolni misol uchun olamiz: Aylana markazi (100,100) nuqtada bo`lgan radiusi 50 ga teng va boshlangich burchagi 10 gradus oxirgi burchagi 50 gradusli aylana, sektor va yoini chizing. Berilgan qiymatlari hamda Graphics classining metodlari yordamida chizamiz har bir chizmani alohida- alohida tugmalarga joylashtiramiz. Tugmalar bosilganda mos ravishda aylana, sektor, yoy, elips chiziladi. Endi esa blok sxemasini chizamiz. Blok sxema

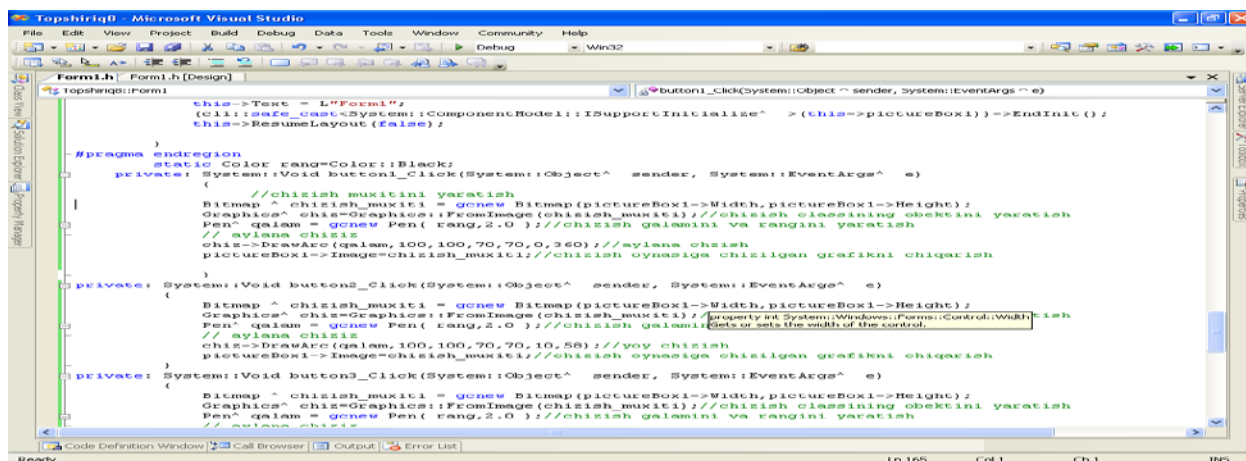


Blok sxemasini chizgandan keyin dasturning ko`rinishini tuzamiz (1 – rasm)



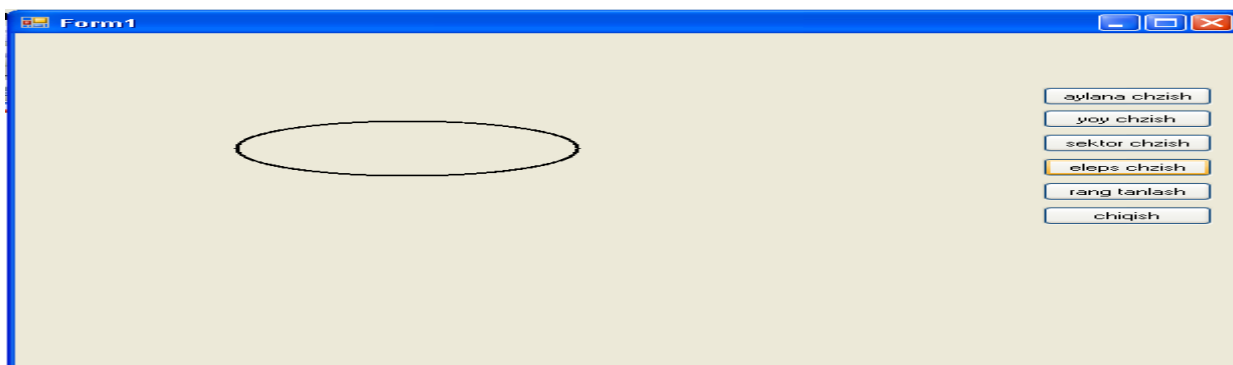
1- rasm

Ko`rinishini tuzib olgandan keyin dasturiy qismini yozamiz.(2 – rasm)



2 – rasm

Dasturiy qisimni yazib bo`lgandan keyin dasturni ishga tushiramiz va mos ravishda tugmalarni bosib chizmalarni chiqaramiz.(3 - rasm)



3 – rasm

8-amaliy mashg'ulotga topshiriqlar:

Markaziy koordinatalari (X.Y) bo'lgan radiusi (bosh burchagi, oxirgi burchagi va ovali) berilgan. Aylana (yoki yoy), sektor, ellipslar chizing.

Topsh. Tar.	Markaz koordin.	rad.	rang.	Bosh bur.	Oxir. bur.	Oval	Ayl.	Yoy	Sektor	El lips
1	(120,30)	70	Kŷk	90	50	2	xa	xa	yo'q	xa
2	(140,40)	90	1	1	-1	1/4	xa	Yo'q	xa	xa
3	(100,100)	50	1	-1,2	-3	-	xa	Yo'q	xa	yo'q
4	(50,50)	60	1	1,2	3	-	xa	xa	yo'q	yo'q
5	(140,100,)	60	1	-	-	5/16	xa	Yo'q	yo'q	xa
6	(190,40)	90	1	-1,1	-1,2	5/6	xa	Yo'q	xa	xa
7	(140,50)	80	1	-1,2	-2,6	2	xa	xa	xa	xa
8	(80,20)	60	1	-1,2	-1,6	4	xa	yo'q	xa	xa
9	(110,140)	50	1	1	1,8	2	xa	xa	yo'q	xa
10	(145,60)	90	1	-	-	3/4	xa	xa	yo'q	xa
11	(110,20)	80	1	1,4	2,2	-	xa	xa	yo'q	yo'q

12	(120,30)	80	1	1,8	2,8	2/7	xa	xa	yo'q	xa
13	(190, 50)	70	1	-	-	2/5	xa	yo'q	yo'q	xa
14	(140,60)	80	1	1,4	2,2	-	xa	xa	yo'q	yo'q
15	(100,80)	70	1	1,2	2,4	3/7	xa	xa	yo'q	xa
16	(160,60)	80	1	1	-3	2/9	xa	yo'q	yo'q	xa
17	(100,140)	70	1	1	1,8	2	xa	xa	yo'q	xa
18	(120,40)	90	1	1,8	-2,9	-	xa	xa	xa	yo'q
19	(140,60)	100	1	-1	-1,5	2/7	xa	yo'q	xa	xa
20	(120,40)	110	1	0,9	-1,2	2	xa	yo'q	xa	xa
21	(110,20)	70	1	1	1,8	3	xa	xa	yo'q	xa
22	(140,50)	80	1	1	-2	2/4	xa	yo'q	xa	xa
23	(190,40)	110	1	-1,2	-1,3	3/4	xa	yo'q	xa	xa
24	(140,30)	80	1	-	-	2/3	xa	yo'q	yo'q	xa
25	(80,20)	60	1	-1,2	-1,8	4	xa	yo'q	xa	xa
26	(100,80)	70	1	2.1	-2	4/7	xa	yo'q	xa	xa
27	(110,120)	80	1	1	1,8	2	xa	xa	yo'q	xa
28	(110,30)	60	1	1.2	2,4	3/9	xa	xa	yo'q	xa
29	(160,110)	90	1	-1.4	-2,8	3	xa	yo'q	xa	xa
30	(120,40)	90	1	1,8	-2,9	-	xa	xa	xa	yo'q

8 -amaliyot mashg'uloti uchun test:

1. Dasturiy taminot ekspluatatsiya (ishlatish) jarayoni nimalardan iborat?
 - A. 1. Dasturni kompyuterga kiritish xamda chiqarish.
 2. Buyurtmachi talabini qondirish.
 3. Dasturiy ta'minot ishonchliligini ta'minlash.
 - B. 1. Dasturni ustada amallar bajarish.
 2. Kutiladigan natijalarni olish.
 3. Dasturini ishlatish.
 - * C. 1. Dasturni kompyuterga kiritib, kirish ma'lumotlari ustada amallar bajarish.
 2. Kutiladigan natijalarni olish.
 3. Dasturiy ta'minot ishonchliligini ta'minlash.
2. Dasturiy ta'minotni kuzatib borish jarayonida qanday o'zgarishlarni kiritish lozim?
 - *A. 1. Xatolarni tuzatish
 2. Dasturiy ta'minotda ishlatiladigan apparatura konfiguratsiyasiga moslashtirish.
 3. Modernizatsiya – dasturni funktsional xossalarini yaxshilash.
 - B. 1. Xatolarni kuzatish.
 2. Dasturiy ta'minotni moslashtirish.
 3. Dasturni xossalarini yaxshilash.
 - C. 1. Xatolarni topish.
 2. Dasturiy ta'minotni ishlatilatis.
 3. Modernizatsiyalash.
3. Xatolarni tuzatish, modernizatsiya (yangilash) hamda adaptatsiya -moslashuvchanlik uchun necha foiz(%)dan ajratilgan?
 - A. 20%, 20%, 60% *B. 20%, 60%, 20% C. 60%, 20%, 20%
4. Qaysi bo'limda agar modullarning vazifasi odatdagi standart vazifalar bilan mos tushmasa u xolda modullargagina xos bo'lgan belgilashlar keltiriladi?
 - A. Ichki spetsifikatsiya xujjati bo'limida

B. Ichki kattaliklar bo'limida

* C. Tadbiq qilish strategiyasi bo'limida

5. rostlashni tashkil qilishda ko'rsatmalar xaqidagi 1-qoidani ko'rsating.

*A. DV ni ishlanmasini testlashni asosiy masala deb xisoblang, uni yuqori malakali va iqtidorli dasturchilarga topshiring, o'zingiz dasturingizni testlash maqsadga muvofik emas.

B. Dasturni to'g'riligini ko'rsatadigan emas, xatolarni topish extimoli katta bo'lgan test yaxshidir.

C. DV ni ishlanmasini xatolarni topish extimolini asosiy masala deb xisoblang, dastur to'g'riligini ko'rsatishdir.

6. Tadbiq qilish strategiyasi bo'limidagi 1-qadamni to'g'ri belgilang.

A. Xar bir sikl uchun quyidagi uchta vaziyatni ta'minlaydigan testlarning mavjudligiga ishonch xosil qilish uchun, modul matnini tekshiring, sikl jismi bir marta bajariladi va sikl jismi maksimal son marta bajariladi.

B. Ixtiyoriy bo'linishining xar yili xech bo'lmaganda bitta yilda o'tmishga ishonch hosil qilish uchun modul matnini tekshiring.

* C. Mumkun bo'lgan va mumkun bo'lmagan qiymatlarining har bir soxasi uchun bajariladi.

7. Tadbiq qilish strategiyasi bo'limidagi 2-qadamni to'g'ri belgilang.

A. Mumkin bo'lgan va mumkin bo'lmagan qiymatlarining xar bir soxasi uchun bajariladi

*B. Ixtiyoriy bo'linishining xar yili xech bo'lmaganda bitta yilda o'tmishga ishonch xosil qilish uchun modul matnini tekshirin

C. Xar bir sikl uchun quyidagi uchta vaziyatni ta'minlaydigan testlarning mavjudligiga ishonch xosil qilish uchun, modul matnini tekshiring, sikl jismi bir marta bajariladi va sikl jismi maksimal son marta bajariladi.

8. DVsi effektivligini ta'minlash uchun qanday prinsiplarga amal qilish tavsiya qilinadi?

A. Avval dastur tavsifini ishlab chiqish kerak.

B.Avval optimallashtiruvchi kompilyatordan foydalanish kerak, so'ngra DVsini ishlab chiqish kerak.

* C. Oldin ishonchli DVsini ishlab chiqish kerak, so'ngra uning effektivligini oshirish bilan shug'ullanish kerak. DV si effektivligini oshirish uchun optimallashtiruvchi kompilyatordan foydalanish kerak.

Javob : 1. C 2.A 3.B 4. C 5.A 6. C 7.B 8. C

Dastur kodi BMIning ilova qismi 8-ilova bandida keltirilgan.

III bob bo'yicha xulosa

Bu bobda quyidagi sakkizta amaliyot ishlari birma-bir batafsil ko'rib chiqildi:

- 1.Maqsadlarni aniqlash. "Masala qo'yilishi" hujjati.
2. Yashash sikl modellari.
3. Foydalanuvchi bilan muloqot.
4. "Texnik topshiriq" hujjati.
5. DT arxitekturasi. "Tashqi spetsifikatsiya" hujjati.
6. Kichik dasturlar standarti. "Ichki spetsifikatsiya" hujjati.
7. DTni testlash. Testlash prinsiplari va ularning usullari.
8. DTni ishlatish.Ishlatishga ko'rsatmalar.

Shu bilan birga har bir amaliyot ishlari o'z ichiga nazariy bilimlardan tashqari ishning maqsadini, talabaning mustaqil ishiga topshiriqni, amaliyot ishini bajarish tartibini, ish bo'yicha hisobotnomani, talaba bilim darajasini aniqlash uchun test hamda nazorat savollarini, ishlatiladigan adabiyotlar ro'yxatini, nazariy materialni o'rganish bo'yicha uslubiy qo'llanmani o'z ichiga oladi.

Auditoriyadagi amaliyot ishidan oldin talaba keyingi darsga tayyorlab kelishi kerak bo'lgan mustaqil ish uchun topshiriqni oladi. Mustaqil ish

bo'yicha hisobotnoma nazorat savollariga qisqa javoblarni hamda topshiriqni dasturi va yechimini o'z ichiga olishi kerak.

Auditoriyada o'qituvchi tomonidan talabanning bilim darajasi tekshirilgandan keyin o'z varianti bo'yicha talaba topshiriqni bajaradi. Dars yakunida o'qituvchi talaba bajargan ishini tahlil qiladi va baholaydi.

Auditoriyadagi darsda material qanchalik chuqur o'rganilishiga e'tibor beriladi. Mavzuni mustaqil o'rganishda talabaga tushunarsiz bo'lgan savollar aniqlanadi va tushuntiriladi.

IV BOB. XAYOT FAOLIYATI XAVFSIZLIGI

4.1. Ishlab chiqarish muhitida mikroiklimni inson organizmiga ta'siri.

Ishlab chiqarish zonalari ob-havo sharoitini havoning quyidagi ko'rsatgichlari belgilaydi:

1. Havoning xarorati graduslarda o'lchanadi
2. Havoning nisbiy namligi foizlarda aniqlaniladi
3. Havo bosim R mm simob ustuni yoki Paskal bilan o'lchanadi.
4. Ish joylardagi havo harorat tezligi m/sek o'lchanadi

Ulardan tashqari ob havo sharoitiga ta'sir qiluvchi ishlab chiqarish omillari ham mavjud, bular har xil mashina mexanizmlari va ishlov berilayotgan material yuzalaridan tarqaladigan issiqlik nurlari bo'lib havo haroratini oshirishga olib keladi.

Bu omillar ta'siridan hosil bo'ladigan ishlab chiqarish zonasidagi havo muhitini sanoat mikroiklimi deb ataladi.

Ishlab chiqarishda xonalarga qo'yiladigan mikroiklimning gigiyenik normalari.

Ishlab chiqarish mikroiklim normalari mehnat xavfsizligi standartlar tizimi «Ish zonasi mikroiklim» me'yorlariga asosan belgilangan. Ular gigiyenik, texnik va iqtisodiy negizlarga asoslangan.

Sanoat korxonalaridagi xonalar, yil fasllari va ish toifasiga qarab, ulardagi harorat, nisbiy namlik va havo harakatining ish joylari uchun ruxsat etilgan normalarida belgilangan.

Ish toifalari quyidagicha belgilaniladi: 1 toifa yengil jismoniy ishlar- o'tirib, tik turib yoki yurib bajariladi, biroq muntazam jismoniy zo'riqish yoki yuk ko'tarishni

talab qilmaydi, energiya sarfi soatiga 150 kkal (172J)ni tashkil qiladi. Bunga tikuvchilik, aniq asbobsozlik va shu kabi ishlar kiradi.

II toifa o'rtacha og'irlikdagi jismoniy ishlarga- soatiga 150-250 kkal (172-293J) energiya sarflanadigan faoliyat turlari kiradi. Bunga doimiy yurish va og'ir bo'lmagan (10 kg gacha) yuklarni tashish bilan bog'liq bo'lgan ishlar kiradi. Masalan yigiruv- to'qish ishlari, mexanik –yigiruv, payvandlash sexlaridagi ishlar kiradi.

III toifadagi og'ir jismoniy ishlar- muntazam jismoniy zoriqish, xususan ogir yuklarni muttasil bir joydan ikkinchi joyga ko'chirish va ko'tarib yurish bilan bog'lik ishlar kiradi. Bunga energiya sarfi soatiga 250 kkal dan yuqori bo'ladi. Bunday ishlarga temirchilik va boshqa qator sexlarda bajariladi.

Harorat, nisbiy namlik va havo xarakatining tezligi me'yordagi va yo'l qo'yilishi mumkin bo'lgan miqdori ko'rinishida normalanadi. Me'yoriy miqdor deganda odamga uzoq muddat va muntazam ta'sir etganda tashqi muhitga moslashuv reaksiyalarini kuchaytirmasdan organizmning normal faoliyati va issiqlik olatini saqlashini ta'minlaydigan mikroiklim ko'rastgichlarining yig'indisi tushunilib, ular issiqlik sezish mutadilligini vujudga keltiradi va ish qobiliyatini oshirish uchun shart –sharoit hisoblanadi. Yo'l qo'yilishi mumkin bo'lgan mikroiklim sharoiti organizm faoliyatini va issiqlik holatidagi o'zgarishdarni, fiziologik moslanisht imkoniyatlaridan chetga chiqmaydigan tashqi muhitga moslanish reaksiyalarini kuchayishini bartaraf etadigan va tez normaga soladigan mikroiklim ko'rsatishlarining yig'indisidir. Bunda sog'lik uchun xatarli holatlar vujudga kelmaydi, biroq kayfiyatning yomonlashuvi va ish qobiliyatining pasayishi kuzatilishi mumkin.

Ishlab chiqarish xonalarida harorat, nisbiy namlik va havo harakati tezliklarining me'yorlari ish toifasiga ko'ra Sovuqda :

	Havo harorati.	Nisbiy namlik	harakat tezligi m/sek
1 toifa	20-23	60-30%,	0.2

2 toifa	18-20	60-40	0.2
3 toifa	16-18	60-40	0.3

issiq:

1toifa	20-30	60-30	0.3
2 toifa	20-30	60-30	0.5-0.7
3 toifa	20-30	60-30	0.5-1.0

Yilning sovuq kunlarida yo'l qo'yiladigan normalar

1 toifa	19-25	75	0.2
2 toifa	17-25	75	0.3
3 toifa	13-25	75	0.5

issikda 1 va 2 toifalada	28	55-75	0.2-0.7
3 toifada	26	75	0.5-1.0

Inson hayoti havoning ahamiyati juda kattaligi ma'lum. Uning kimyoviy tarkibi, fizikaviy hususiyatlari va tarkibidagi har xil moddalarning bo'lishi, havodan nafas olish, mehnat qilayotgan kishilar uchun juda muxim. Chunki havoning tozaligi inson salomatligini saqlovchi muhim omil hisoblaniladi. Yer atmosferasi quruq havo bilan ma'lum miqdordagi suv bug'laring aralashmasidan tashkil topgan. Ma'lumki havodagi gazlardan eng muhimi kisloroddir. Havo holati uning bosim, zichligi, harorati, absolyut namlik, namlik sig'imi, nisbiy namlik, issiqlik sig'imi va boshqalar bilan belilaniladi. Shamollatish usullari. Korxonalarni loyhalashda iqlim sharoitini, kuyosh nurlarini tushush holatini va xonalrdagi jihozlarni to'g'ri joylashtirish masalalari to'g'ri hal qilingan bo'lsa, shamollatish vositalrini o'rnatish shunchalik oson bo'ladi. Shamollatish vositalarini o'rnatishda shamollatish sxemasining iqtisodiy kam xarajat bulishi bilan birga, iloji boricha kam metall sarf qiliadiganini tanlash zarur.

Ishlab chikarish binoalarida mikroiqlimni saqlab turish uchun ya'ni toza havo uchun ventelatsiya bilan ta'minlash kerak bo'ladi. Ventilatsiyaning keng qo'llaniladigan turi tabiiy umumiy ventilatsiya (aeratsiya) xilidir. Yana mexanik

ventilatsiya xillari mavjud(okimli, so'rib oluvchi, oqimli-surib oluvchi, konditsionerli).Korxonalarda ajralib chikayotgan zararli moddalar faqat issiqlik bo'lsa, unda almashtirilayotgan havoning hisoblanadigan miqdori $G_1=Q/0.24(t_1-t)$ bu yerda

G_1 -chikarilib tashlanishi kerak bo'lgan havo miqdori, kgg's:

Q - ortiqcha issiqlik miqdori:

t_1 - chikarilib yuborilayotgan havoning harorati

t -havo oqimining harorati.

Havodagi ortiqcha namlikni yo'qatishda quyidagi formuladan foydalaniladi

$L=G/(d_1-d)p$ bu erda

L -havo oqimining xajmi m^3/g :

G -binodagi suv bug'laring massasi ,g/s:

d_1 - chikarilib yuborilayotgan havodagi namlik g/kg

d - tashqi havodagi namlik miqdori g/kg.

Havodagi gaz holdagi zararli moddalarni yo'qotishda quyidagi formuladan foydalaniladi $L=K/K_{ch}-K_{ok}$ bu yerda:

K - bino ga tarkaluvchi zararli moddalarni massasi mgg's:

K_{ch} – chikarilib yuborilayotgan zararli moddalarning konsentratsiyasi mgg's: K_{ok} - havo okimidagi zararli moddalar konsentratsiyasi mgg's:

K_{ch} ning kattaligi REK teng yoki kamroq bo'lishi kerak. K_{ok} ning kons. esa sanitariya normalaridan kamroq bo'lib ruxsat etilgan kons. 0.3 dir.

Tabiiy shamollatish tashqaridan bino ichiga kirgan sovuq havo bino ichidagi issiklik xisobiga issiklik kabul kilib, isigandan keyin xajmi kengayadi va engillashib binoning yukorisiga karab xarakatlanadi va truba orqali tashqariga chiqib ketadi. Bu xodisa aeratsiya deb ataladi.

Mexanizmlar ishlagan vaqtda uzidan ko'p issiqlik ajratib chikaradi va bu issiklikni tabiiy shamollatish yo'li bilan chikarib yuborish iqtisodiy samara beradi. Bunda

asososiy e'tiborni havoni kirish va chiqish joylarini ta'minlashga qaratilgan. Ma'lumki issiq havo yuqoriga qarab sovuq havo pastga qarab yo'naladi. Shuning uchun ko'p issiqlik ajralib chiqadigan xonalarda sovuq havoni poldan 4 m balandlikdan berish maqsadga muvofikdir. Sovuq havo pastga yo'nalishida issiq havo bilan aralashadi, isiydi va vujudga kelgan tabiiy oqimlar xarakatiga qo'shilib uzluksiz harakat hosil qiladi. Bu uzluksiz harakat davomida oqimlarga yangidan-yangi miqdorlar qushilishi natijasida yuqori to'siqlar tomon yunaladi va bir qismi tabiiy shamollatish tirqishidan tashqariga bir qismi sovib yana pastga tushadi va bu bilan xonadagi havoni aylanma xarqatinikuchaytirishga uz xissasini qushadi. Shunday qilib binolarning ichida havo harakatlarining tutash oqimlari vujudga keladi.

Tabiiy shamollatishni hisoblashda, asosan ma'lum isish hisobiga engillashib, binoning yuqori qismida yig'ilgan ortiqcha bosimni biron-bir havo chiqarib yuborish joyidan tashqariga yo'naltirish mo'ljallangan.

Ortiqcha bosim balandlik hisobiga hosil bo'lganligidan uni quyidagicha ifodalash mumkin: $R=N(\gamma-\gamma)$

Bundan tashqari tabiiy havo almashinish shamol ta'sirida bulishi mumkin. Agar bino shamol uriladigan tomondagi bosim shamol hisobiga birmuncha ijobiy bo'lsa, shamol urmaydigan tomonda bosim salbiy yo'nalishda bo'lsa unda quyidagi formula orqali ifodalash mumkin: $\Delta R=R-R_2$ bu erda R – shamol urilayotgan tomondagi bosim: R_2 –shamol urilmaydigan tomondagi bosim.

Agar binoga xar ikkala bosim kuchi tabiiy shamollatish vazifasini bajaradi deb hisoblasak:

$$\Delta R=(\gamma-\gamma) N+(R-R)$$

Ortiqcha bosim miqdorini aniqlangandan keyin chiqarilib yuboriladigan havo miqdorini ham aniqlash mumkin. $Q=\mu f\sqrt{2q\Delta R}$ Bu erda

μ - havo miqdori koeffitsenti:

f - havo chiqarilib yuborilayotgan teshik kesim yuzasi:

Agar chiqarilib yuborilayotgan havo miqdori kirib kelayotgan havo miqdoriga teng desak unda biz kirib kelayotgan havo harakati tezligini topishimiz mumkin :

$$V=Q/G' \quad \text{bu erda:}$$

G' -havo chiqib ketayotgan tiriqish kesim yuzasi

Havodagi changdan tozalashda turili vositalardan foydalanish mumkin, chang tozalagichlarning turlari ko'p. Chang tozalovchi apparatlar ishlatish jihatidan qo'layligi, chang tozalash darajasiva arzon-qimmatligiga qarab tanlaniladi.

Havo namligini mo''tadilashda ham muhim ahamiyatga ega. Havoni mo''tadillashtirish deganda sanoat korxonasi xonalaridagi havo holatini ichki omillar: issiqlik ajralib chiqish, namlik va tashqi omillar havoning issiq sovuqligidan qat'iy nazar, avtomatik ravishda bir xil me'yorda saqlab turish tushuniladi.

Havoni sanitariya-gigiena sharoitini yaxshilashdan tashqari texnologik talablar asosida ham mu'tadillashtirish mumkin.

Havoni mu'tadillashtirish uchun maxsus kondensionelardan foydalaniladi Kondensionerlar havoni qabul qiladi filytirib beradi, issiqyoki sovuq bilan ta'minlaydi, harakatga keltiradi, namlaydi va boshqa jarayonlarni bajaradi

4.1.Yong'in xavfsizligi.

Yong'in iqtisodiyotga katta moddiy zarar yetkazadi. Buning ustiga, ko'p hollarda yong'in vaqtida baxtsizlik hodisalari ham ro'y beradi. Bu esa o'z navbatida yong'indan saqlanish hamda mehnat muhofazasi qoidalarini yaxshi o'rganib-bilib olish hamda ularni o'zaro uzviy bog'lay bilish vazifasini yuklaydi.

Yong'inga qarshi kurashish texnikasining vazifalari yong'in chiqish sabablarini o'rganish, yong'inning oldini olishning eng maqbul usullarini, shuningdek yong'inni qisqa vaqt ichida o'chirish vositalari va usullarini topish hamda belgilab olishdan iboratdir.Yong'in maxsus manbadan tashqarida bo'ladigan, nazorat qilib bo'lmaydigan yonish bo'lib, juda katta moddiy zarar yetkazadi.

Yong'in chiqishga asosan olovdan noto'g'ri foydalanish; elektr ustanovkalarni, pechlarni, tutun trubalarini montaj qilish va ishlatish qoidalarining buzilishi; xalq xo'jaligi ob'ektlarini loyihalash va qurishda yong'in xavfsizligi normalari talabalarining buzilishi; yong'in jihatdan xavfli jihozlarni ishlatishda va oson alanganlanadigan materiallarda foydalanishda yong'in xavfsizligi qoidalariga rioya qilmaslik; bolalarning olov bilan o'ynashi; momaqaldiroq razryadlari sabab bo'ladi.

Yong'inning oldini olish va yong'indan saqlash tartiblari yong'inning ochiq alanga va uchqunlar, havo, buyumlarning yuqori temperaturasi, zaharli yonish mahsullari, tutun, kislorodning kamayib ketishi, bino hamda inshootlarning qulashi va shikastlanishi, portlash kabi faktorlarning odamlarga ta'sirining oldini olish kerak. Bu vazifalarni hal qilish uchun yong'in-portlash jihatdan xavfli modda va materiallar o'rniga yonmaydigan hamda qiyin yonadigan material va moddalardan iloji boricha ko'p foydalanish, yonuvchi muhitni izolyatsiyalash (texnologik jarayonlarni avtomatlashtirish, germetiklash va h.k.), yong'inning yonish manbaidan atrofga tarqalishiga yo'l qo'ymaslik, o't o'chirish vositalari, guruhli hamda yakka tartibda himoya vositalari, signalizatsiya va yong'in haqida xabar berish vositalaridan foydalanish, yong'in chiqqanda odamlarni evakuatsiya qilish tartibini to'g'ri tashkil etish, ob'ektlarni yong'indan qo'riqlash zarur. Yonish jarayoni havo kislorodi bilan yonuvchan moddaning ximiyaviy birikish reaksiyasidan iborat. yonish jarayonida juda ko'p miqdorda issiqlik va yorug'lik chiqadi. Moddalarning yonishi brom, oltingugurt, xlor va boshqa moddalar bilan biriktirishda ham yoki ularning bug'lari mavjud bo'lganda ham sodir bo'lishi mumkin. Portlash - yonishning hususiy holi bo'lib, u bir onda kechadi va bunda qisqa vaqt ichida ko'p miqdorda issiqlik va yorug'lik chiqadi. Deyarli hamma yonuvchi moddalar, ularning qanday agregat birikma holida bo'lishidan qat'iy nazar (suyuq, qattiq, gazsimon va hokazo), uglerod S., vodorod N., kislorod O dan tashkil topgan organik birikmalardir. Moddaning yonuvchanligi asosan undagi S va N ning miqdoriga bog'liq. Yonuvchan aralashma alanga olguncha o'z-o'zidan qizib, oxiri yona boshlaydigan temperatura yonuvchan aralashmaning o'z-o'zidan alanganlanish temperaturasi deb ataladi. Bunday oksidlanish reaksiyasining tezligi shunday bo'ladiki, ajralib chiqayotgan issiqlik

atrof-muhitga yutilishga ulgurmayi. O'z-o'zidan yonish o'z-o'zidan alanganishdan shu bilan farq qiladiki, o'zqo'zidan yonish tashqi issiqlik manbai hisobiga emas, balki moddaning o'zida ro'y beradigan ximiyaviy, biologik yoki fizik jarayonlar hisobiga sodir bo'ladi.

Qiziganda bug' va gazlar hosil qila oladigan hamma moddalarning yonish prtsessi alanga bilan kechadi. Koks,grafit, pistako'mir, qurum alangasiz yonadi. O't olish harorati yonuvchan moddalar (suyuqlik)ning eng past harorati bo'lib, shu haroratda ochiq alov yaqinlashtirilganda gaz yoki bug'larning havoga qo'shilganda alanganadigan aralashmasi hosil bo'ladi. Oson alanganadigan suyuqliklar 45S dan past haroratda, yonuvchi suyuqliklar 45S dan yuqori haroratda o't oladi. Yonuvchi qattiq moddalarning alanganish tezligi va yonish tezligi ularning solishtirma sirtiga bog'liq. Yonuvchi gazlar, bug'lar, changlar (atsetilen, vodorod, SO₂, benzin, skipidar, toshko'mir changi va hokazo) havo kislorodi bilan aralashib portlovchi aralashmalar hosil qilishi mumkin.Portlash uchun quyidagi ikki shart:

1) bug'-havo yoki gaz-havo aralashmasining muayyan konsentratsiyasi bo'lishi:

2) moddani o'z-o'zidan alanganish haroratiga qizdira oladigan impulys (alanga, zarb, siqilish va hokazo) bo'lishi kerak.

Quyida ba'zi moddalarning pastki va yuqorigi portlash chegaralari keltiriladi:

	P_{pastki}	P_{yuqori}
Benzin.....	1,1%	5,4%
Vodorod.....	4,1%	75%
Atsetilen.....	1,5%	82%
Uglerod (IV) oksid	12,8%	75%

Havo oqimiga ko'ra, yonish to'liq yoki chala bo'ladi. Chala yonish sharoitida ajraladigan yonish mahsullari inson hayoti va sog'lig'i uchun xavf tug'diradi. Masalan, 3-4,5% So₂ inson uchun xavfli bo'lsa, 0,4% SO uni haloq etadi. 60-70 S haroratli havodan bir necha minut nafas olinganda organizmda qaytmas jarayonlar yuz berib, inson haloq bo'ladi. Gaz-havo yoki bug'-havo aralashmalari trubalarda

yonganda alanganing tarqalish tezligi 0,3-2,7m/s ni, kichik o'lchamli idishlarda yonganda 6,5-10m/s ni tashkil etadi, trubalarda portlash yuz berganda alanga 1000-4000m/s tezlik bilan tarqaladi (detonatsion yonish sodir bo'ladi).

O'lchami 10 sm va undan katta bo'lgan chang (zarrachalar) aerazol deb ataladi va eng xavfli hisoblanadi. Havodan o'tirgan chang aerogel deb ataladi. Yog'och uni aerazolning alanganish harorati 775S ga, aerogeliniki esa 275S ga teng.

IV bob bo'yicha xulosa

Ushbu bobda hayot faoliyati xavfsizligining ,ishlab chiqarish muhitida mikroiqlimni inson organizmiga ta'siri hamda yong'in xavfsizligi boyicha batafsil ko'rsatmalar berildi. Bu bobda hayot faoliyati xavfsizligi uchun xavf soluvchi hodisalar ro'y berganda uni oldini olish hamda unga qarshi ko'rilishi mumkin bo'lgan chora- tadbirlar haqida birma-bir ko'rsatmalar berildi. Misol uchun foydalanuvchi faqatgina dasturiy ta'minotdangina foydalanganda emas, balki hayotda sodir bo'ladigan xavfli baxtsiz voqealarga tayyor turishi uchun ko'rsatmalar berildi. Barcha hayot faoliyatiga tegishli bo'lgan xisob-kitoblar to'g'ri yechilgan.

Ulardan tashqari ob havo sharoitiga ta'sir qiluvchi ishlab chiqarish omillari ham mavjudligi , har xil mashina mexanizmlari va ishlov berilayotgan material yuzalaridan tarqaladigan issiqlik nurlari havo haroratini oshirishga olib kelishi haqidagi malumotlar yoritib o'tilgan. Inson hayoti havoning ahamiyati juda kattaligi ma'lum. Uning kimyoviy tarkibi, fizikaviy hususiyatlari va tarkibidagi har xil moddalarning bo'lishi, havodan nafas olish, mehnat qilayotgan kishilar uchun juda muhim hisoblanadi. Havoning tozaligi inson salomatligini saqlovchi muhim omil hisoblanganligi sababli barcha kerakli chora –tadbirlar to'liq yoritilgan.

Turli xil asbob uskunalarning qisqa to'qnashuvi sababli yong'in iqtisodiyotga katta moddiy zarar yetkazadi. Buning ustiga, ko'p hollarda yong'in vaqtida baxtsizlik hodisalari ham ro'y beradi. Yong'indan saqlanish hamda mehnat muhofazasi qoidalarini yaxshi o'rganib chiqildi.

Xulosa

O'quv jarayonidagi asosiy muammolardan biri talabalarning o'zlashtirish darajasini oshirish, berilayotgan bilimlarni tushunish, xotirada saqlash va qo'llash yo'llarini egallashni yaxshilashdan iborat bo'lmoqda.

Ma'lumki, inson ma'lumotlarning 80 foizini ko'rish, 15 foizga yaqinini eshitish va qolgan 5 foizini maza-ta'm bilish- sezgi organlari orqali oladi. Biroq xayotda ma'lumotlarni faqat qabul qilibgina qolmay uni yodda saqlash xam kerak. Bunda xotira yetakchi o'rin egallaydi. Inson ma'lumotlarni ko'rganiga eshitganiga nisbatan o'zi nimadir yaratib, ijod etib amaliyotda qo'llaganida yodida ko'proq, yaxshi saqlab qoladi. Bunday dars jarayonlarini amalga oshirishda yangi texnologiyalar bebaho yordamchi bo'lib qolmoqda.

Axborot texnologiyalarini ta'lim jarayoniga joriy etish bo'yicha jamiyatimizning tez suratlar bilan o'sib boruvchi ehtiyojlari oliy ta'lim muassasalarida auditoriya va auditoriyadan tashqari mashg'ulotlarda elektron qo'llanmalar, virtual stendlar, Internet tarmog'i imkoniyatlaridan foydalangan holda o'rganish, shuningdek, masofaviy ta'limni joriy etish bo'yicha izchil nazariy hamda amaliy tadbirlar bajarilishini taqazo etadi.

Bugungi kunda mamlakatimizning ta'lim sohasiga, shu jumladan oliy ta'limga katta e'tibor qaratilmoqda. Masofaviy ta'lim g'arbda an'anaviy bo'lib qoldi.

Har bir yuqori mavqega ega bolgan universitet hozirgi kunda albatta masofaviy ta'lim dasturiga ega. Tahminlarga qaraganda aynan masofali ta'lim keng tarqalib boradi. Ta'limning bu shaklini «Kelajak ta'limi» deb atamoqdalar.

Multimediya muhitning qo'llanishi, mavzuni tushunish va ma'lumotlarni faol o'zlashtirishni ta'minlabgina qolmay, talabalarning o'zlashtirish darajasini oshiruvchi motivlardan (matn, tovush, animatsiya grafiklaridan birgalikda foydalanish imkoniyatlari) hisoblanadi.

Ammo barcha ham uzoq joylarga borib, universitetlarga ўqishga kirib oliy ta'lim ololmaydi. Shuning uchun masofaviy ta'lim olish taklif etiladi..

Dastavval masofali ta'lim oliy ma'lumotga ega bo'lgan insonlarning malakalarini oshirish uchun mo'ljallangan edi. Hozirgi kunda texnikaning rivojlanishi bilan masofali ta'lim imkoniyatlari kengaymoqda.

Masofali ta'lim shubhasiz ta'lim tizimi shakllaridan biri hisoblanadi va insonning bilim olish va axborotlardan foydalanish huquqini amalga oshirishga yo'naltirilgan. Masofaviy ta'lim har bir inson uchun kasbiy faoliyati bilan bir qatorda asosiy yoki qo'shimcha ta'lim olish imkoniyatini beradi.

Global kompyuter tarmog'i – Internetning rivojlanishi natijasida, jahon ta'lim tizimi takomillashuvining yangi istiqbollari paydo bo'ldi. Hozirda ta'limning an'anaviy usullari Internet, kompyuter va kommunikatsiya tizimlarini qo'llashga asoslangan o'qitishning yangi uslublari bilan boyib bormoqda.

Masofaviy talim –bu turli geografik mintaqalarda joylashgan o'qituvchi va talabalarni bir-biri bilan bog'lovchi jarayondir.

Adabiyotlar ro'yxati.

1. D.A.Chernev. Texnologiya programirovaniya, 2003, O'qituvchi
2. D.A.Chernev. Texnologiya razrabotki programmnoy obespecheniya 2004
Toshkent: Mexnat
3. G.S.Ivanova. Texnologiya programirovaniya, 2002, M.:MGTU im. Baumana
4. G.S.Ivanova. Osnovi programirovanie, 2001, M.:MGTU im. Baumana
5. Knut D. Iskusstvo programirovaniya dlya EVM: Poluchislennye algoritmi, 1977.
6. Knut D. Iskusstvo programirovaniya dlya EVM: T.2. Poluchislennye algoritmi,
1977, Spravochnoe izdanie - M. : Mir
7. Knut D. Iskusstvo programirovaniya dlya EVM: T. 3 : Sortirovka i poisk, 1978
Virt N. Algoritmi + strukturi dannix= programmi, 1985, M. Mir
8. Virt N. Algoritmi i strukturi dannix: s primerami na Paskale, 2005 SPb. : "Nevskiy
dialekt"
- 10.G.Ivanova i dr. Ob'ektno-orintirovannoe programirovanie, 2001, M.:MGTU im.
11. Lipaev V.V. Nadejnosty programmnykh sredstv. - M.: «Sinteg», 1998.
12. Kosimova Sh.T., Chernov D.A. Programmalashch texnologiyasiga kirish.
Ukuv kullanma. Toshkent: ToshDTU, 1997.
- 13.N.X.Latipova. D.A.Chernev «Dasturlash texnologiyasi» fanidan tajriba ishlarini
o'tkazish bo'yicha metodik kursatmalar. T: ToshDTU, 2000.
14. www.C++kitob.tayyor.uz
15. [www.C++\(Uzb\).uz](http://www.C++(Uzb).uz)
16. [www.C++ Для чайников.ru](http://www.C++Для чайников.ru)
17. www.Как программировать на C++.ru
18. Shcherbina Ya.Ya. Osnovi protivopojarnoy texniki, Kiev.: Budyvelynik, 1985.
- 19.GOST 12.1.005-88 Sistema standartov bezopasnosti truda. Vozdux rabochey zoni.
Obshchie sanitarno-gigienicheskie trebovaniya.
- 20.GOST 12.2.003-91 Sistema standartov bezopasnosti truda. Oborudovanie
proizvodstvennoe. Obshchie trebovaniya bezopasnosti.

ILOVA

1- ILOVA

```
// topshiriqq1.cpp : Defines the entry point for the console application.
//

#include "stdafx.h"
#include<iostream>//kiritish, chiqarish funksiyalarining to`plami

using namespace std;//standart oqimlarning nomlari muhiti
int _tmain(int argc, _TCHAR* argv[])
{
    float x,y,a,b;//float turdagi o`zgaruvchilar, ya'ni kasr qismli sonlarni kasr
qismini ham//o`zlashtiradi.
    cout<<"\n1 - topshiriqni yechish dasturi.\n A va B sonining qiymatini
hisoblashdan oldin "<<
        "X va Y sonlarini qiymatlarini kiritish kerak.\n\n";//ekranda chop
ettiriladigan matn
    cout<<"\n X sonining qiymatini kiriting.\n X = "; //ekranda chop ettiriladigan
matn
    cin>>x;//x ga ekranda kiritilgan(ya'ni klaviatura yordamida yozilgan son, 1 -
probelgacha. )//son qiymatini o`zlashtirish funksiyasi, bu son kasr qismli yoki butun
son bo`lishi mumkin.
    cout <<"\n Y sonining qiymatini kiriting.\n Y =";//ekranga chiqariladigan
matn
    cin>>y;//y ga ekranda kiritilgan(ya'ni klaviatura yordamida yozilgan son, 1 -
probelgacha. )//son qiymatini o`zlashtirish funksiyasi, bu son kasr qismli yoki butun
son bo`lishi mumkin.
    //X va Y sonlarini kiritgandan keyin A va B soning qiymatini hisoblaymiz
    a=x*y/14+11;// A sonini qiymatini hisoblash
    b=y/a;// B sonini qiymatini hisoblash
    cout<<"A sonning qiymati.\n A = "<<a<<endl;// B sonini qiymatini ekranga
chop ettirish.
    cout<<"B sonning qiymati.\n B = "<<b<< endl; // A sonini qiymatini
ekranga chop ettirish.
    cin>>(float)x;//dastur yechimini ko`rish uchun pauza vazifasini bajaradi. Ya'ni
X ni qiymati
    //kiritilmaguncha dastur ishlab turadi.
    return 0;
}
```

2- ILOVA


```

#include<iostream>//kiritish, chiqarish funksiyalarining to`plami
#include<math.h>//matematik amallar funksiyalarining to`plami
using namespace std;//standart oqimlar nomlari muhiti
int main();// asosiy funksiya
{
    float a,b,c;//float turidagi o`zgaruvchilar, ya'ni kasr qisimli sonlarni kasr
qismini ham//o`zlashtiradi.
    cout<<"n2 - topshiriqni yechish dasturi.\n\n ";//ekranda chop ettiriladigan
matn
    a=sqrt(2.0)+0.8;// A sonini qiymatini hisoblash
    b=sqrt(2.0)-0.2;// B sonini qiymatini hisoblash
    c=((2-b)/(b-1)+2*(a-1)/(a-2))/(b*(a-1)/(b-1)+a*(2-b)/(a-2));// C sonini
qiymatini hisoblash
    cout<<"C sonining qiymati.\n c = "<<c<<endl;// C sonini qiymatini ekranga
chop ettirish.
    cin>>(float)a;//dastur yechimini ko`rish uchun pauza vazifasini bajaradi. Ya'ni
X ni qiymati
    //kiritilmaguncha dastur ishlab turadi.
    return 0;//asosiy funksiya bajarilib bo`lganda uning qiymati 0 ga
tenglashtiriladi
}

```

3- ILOVA

```

#include<iostream>//kiritish, chiqarish funksiyalarining to`plami
using namespace std;// standart oqimlar nomlari muhiti
int main();// asosiy funksiya
{
    float x,a;//float turidagi o`zgaruvchilar, ya'ni kasr qisimli sonlarni kasr qismini
ham
    //o`zlashtiradi.
    cout<<"n3 - topshiriqni yechish dasturi.\n X soning qiymatini hisoblashdan
oldin "<<
        "A sonini qiymatini kiritish kerak.\n\n ";//ekranda chop ettiriladigan
matn
    cout<<"\n A soning qiymatini kiriting.\n A = "; //ekranda chop ettiriladigan
matn
    cin>>a;//x ga ekranda kiritilgan(ya'ni klaviatura yordamida yozilgan son, 1 -
probelgacha. )
    //son qiymatini o`zlashtirish funksiyasi, bu son kasr qisimli yoki butun son
bo`lishi mumkin.
    //A sonini kiritgandan keyin X sonining qiymatini hisoblaymiz
    if(a<-3)//solishtirish funksiyasi
        //X soning qiymati -3 dan kichik bo`lganda X ni hisoblash

```

```

        x=2*a-3;// X sonini qiymatini hisoblash
else //aks xolda
    if(a==3)//solishtirish funksiyasi
        //X soning qiymati -3 ga teng bo`lganda X ni hisoblash
        x=2+a;// X sonini qiymatini hisoblash
    else//aks holda
        x=1/a;// X sonini qiymatini hisoblash
    cout<<"X sonining qiymati.\n X = "<<x<<endl;// X sonini qiymatini ekranga
chop ettirish.
    cin>>(float)x;//dastur yechimini ko`rish uchun pauza vazifasini bajaradi ya`ni
X ni qiymati
    //kiritilmaguncha dastur ishlab turadi.
    return 0;//asosiy funksiya bajarilib bo`lganda uning qiymati 0 ga
tenglashtiriladi
}

```

4-ILOVA

```

#include<iostream>//kiritish, chiqarish funksiyalarining to`plami
#include<math.h>//matematik funksiyalar to`plami
using namespace std;// standart oqimlar nomlari muhiti
int main()// asosiy funksiya
{
    int c=0;//int turidagi o`zgaruvchilar, ya`ni butun sonlarni o`zlashtiriladi.
    cout<<"n4 - topshiriqning dasturi.\n\n";//ekranga chop etiriladigan matn
        // C ni qiymatini hisoblash sikli
    for(int i=1; i<=20;i++)
    {
        c+=(int)pow((float)i,2);
    }
    //sikl tugagandan keyin C ni qiymatini ekranga chiqarish
    cout<<"C soning qiymati.\n \n C = "<<c<<endl;// B sonini qiymatini ekranga
chop ettirish.
    cin>>(int)c;//dastur yechimini ko`rish uchun pauza vazifasini bajaradi ya`ni X
ni qiymati
    //kiritilmaguncha dastur ishlab turadi.
    return 0;//asosiy funksiya bajarilib bo`lganda uning qiymati 0 ga
tenglashtiriladi
}

```

5- ILOVA

```
/// 5 – topshiriqning dasturiy qismi
#pragma once
#include<Math.h>

namespace topshiriq5 {

using namespace System;
using namespace System::ComponentModel;
using namespace System::Collections;
using namespace System::Windows::Forms;
using namespace System::Data;
using namespace System::Drawing;

/// <summary>
/// Summary for Form1
///
/// WARNING: If you change the name of this class, you will need to change
the
///     'Resource File Name' property for the managed resource compiler tool
///     associated with all .resx files this class depends on. Otherwise,
///     the designers will not be able to interact properly with localized
///     resources associated with this form.
/// </summary>
public ref class Form1 : public System::Windows::Forms::Form
{
public:
    Form1(void)
    {
        InitializeComponent();
        //
        //TODO: Add the constructor code here
        //
    }

protected:
    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    ~Form1()
    {
        if (components)
        {

```

```

        delete components;
    }
}

private: System::Windows::Forms::TextBox^ textBox1;
private: System::Windows::Forms::Button^ button1;
private: System::Windows::Forms::Button^ button2;
private: System::Windows::Forms::Button^ button3;
private: System::Windows::Forms::Button^ button4;
private: System::Windows::Forms::Button^ button5;
private: System::Windows::Forms::Button^ button6;
private: System::Windows::Forms::Button^ button7;
private: System::Windows::Forms::Button^ button8;
private: System::Windows::Forms::Button^ button9;
private: System::Windows::Forms::Button^ button10;
private: System::Windows::Forms::Button^ button11;
private: System::Windows::Forms::Button^ button12;
private: System::Windows::Forms::Button^ button13;
private: System::Windows::Forms::Button^ button14;
private: System::Windows::Forms::Button^ button15;
private: System::Windows::Forms::Button^ button16;
private: System::Windows::Forms::Button^ button17;
private: System::Windows::Forms::Button^ button18;
private: System::Windows::Forms::Button^ button19;

private: System::Windows::Forms::RadioButton^ radioButton1;
private: System::Windows::Forms::RadioButton^ radioButton2;
private: System::Windows::Forms::Button^ button20;
private: System::Windows::Forms::Button^ button21;
private: System::Windows::Forms::Button^ button22;
private: System::Windows::Forms::Button^ button23;
private: System::Windows::Forms::Button^ button24;
private: System::Windows::Forms::Button^ button25;
private: System::Windows::Forms::Button^ button26;
protected:

private:
    /// <summary>
    /// Required designer variable.
    /// </summary>
    System::ComponentModel::Container ^components;

#pragma region Windows Form Designer generated code
    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.

```

```

/// </summary>
void InitializeComponent(void)
{
    this->textBox1 = (gcnew System::Windows::Forms::TextBox());
    this->button1 = (gcnew System::Windows::Forms::Button());
    this->button2 = (gcnew System::Windows::Forms::Button());
    this->button3 = (gcnew System::Windows::Forms::Button());
    this->button4 = (gcnew System::Windows::Forms::Button());
    this->button5 = (gcnew System::Windows::Forms::Button());
    this->button6 = (gcnew System::Windows::Forms::Button());
    this->button7 = (gcnew System::Windows::Forms::Button());
    this->button8 = (gcnew System::Windows::Forms::Button());
    this->button9 = (gcnew System::Windows::Forms::Button());
    this->button10 = (gcnew System::Windows::Forms::Button());
    this->button11 = (gcnew System::Windows::Forms::Button());
    this->button12 = (gcnew System::Windows::Forms::Button());
    this->button13 = (gcnew System::Windows::Forms::Button());
    this->button14 = (gcnew System::Windows::Forms::Button());
    this->button15 = (gcnew System::Windows::Forms::Button());
    this->button16 = (gcnew System::Windows::Forms::Button());
    this->button17 = (gcnew System::Windows::Forms::Button());
    this->button18 = (gcnew System::Windows::Forms::Button());
    this->button19 = (gcnew System::Windows::Forms::Button());
    this->radioButton1 = (gcnew
System::Windows::Forms::RadioButton());
    this->radioButton2 = (gcnew
System::Windows::Forms::RadioButton());
    this->button20 = (gcnew System::Windows::Forms::Button());
    this->button21 = (gcnew System::Windows::Forms::Button());
    this->button22 = (gcnew System::Windows::Forms::Button());
    this->button23 = (gcnew System::Windows::Forms::Button());
    this->button24 = (gcnew System::Windows::Forms::Button());
    this->button25 = (gcnew System::Windows::Forms::Button());
    this->button26 = (gcnew System::Windows::Forms::Button());
    this->SuspendLayout();
    //
    // textBox1
    //
    this->textBox1->Location = System::Drawing::Point(6, 34);
    this->textBox1->Name = L"textBox1";
    this->textBox1->Size = System::Drawing::Size(274, 20);
    this->textBox1->TabIndex = 0;
    this->textBox1->Text = L"0";
    this->textBox1->TextAlign =
System::Windows::Forms::HorizontalAlignment::Right;

```

```

//
// button1
//
this->button1->Location = System::Drawing::Point(22, 113);
this->button1->Name = L"button1";
this->button1->Size = System::Drawing::Size(32, 27);
this->button1->TabIndex = 1;
this->button1->Tag = L"7";
this->button1->Text = L"7";
this->button1->UseVisualStyleBackColor = true;
this->button1->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
//
// button2
//
this->button2->Location = System::Drawing::Point(62, 113);
this->button2->Name = L"button2";
this->button2->Size = System::Drawing::Size(32, 27);
this->button2->TabIndex = 2;
this->button2->Tag = L"8";
this->button2->Text = L"8";
this->button2->UseVisualStyleBackColor = true;
this->button2->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
//
// button3
//
this->button3->Location = System::Drawing::Point(22, 148);
this->button3->Name = L"button3";
this->button3->Size = System::Drawing::Size(32, 27);
this->button3->TabIndex = 3;
this->button3->Tag = L"4";
this->button3->Text = L"4";
this->button3->UseVisualStyleBackColor = true;
this->button3->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
//
// button4
//
this->button4->Location = System::Drawing::Point(102, 183);
this->button4->Name = L"button4";
this->button4->Size = System::Drawing::Size(32, 27);
this->button4->TabIndex = 4;
this->button4->Tag = L"3";
this->button4->Text = L"3";

```

```
        this->button4->UseVisualStyleBackColor = true;
        this->button4->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
        //
        // button5
        //
        this->button5->Location = System::Drawing::Point(22, 183);
        this->button5->Name = L"button5";
        this->button5->Size = System::Drawing::Size(32, 27);
        this->button5->TabIndex = 5;
        this->button5->Tag = L"1";
        this->button5->Text = L"1";
        this->button5->UseVisualStyleBackColor = true;
        this->button5->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
        //
        // button6
        //
        this->button6->Location = System::Drawing::Point(22, 216);
        this->button6->Name = L"button6";
        this->button6->Size = System::Drawing::Size(32, 27);
        this->button6->TabIndex = 6;
        this->button6->Tag = L"0";
        this->button6->Text = L"0";
        this->button6->UseVisualStyleBackColor = true;
        this->button6->Click += gcnew System::EventHandler(this,
&Form1::button6_Click);
        //
        // button7
        //
        this->button7->Location = System::Drawing::Point(102, 148);
        this->button7->Name = L"button7";
        this->button7->Size = System::Drawing::Size(32, 27);
        this->button7->TabIndex = 7;
        this->button7->Tag = L"6";
        this->button7->Text = L"6";
        this->button7->UseVisualStyleBackColor = true;
        this->button7->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
        //
        // button8
        //
        this->button8->Location = System::Drawing::Point(62, 148);
        this->button8->Name = L"button8";
        this->button8->Size = System::Drawing::Size(32, 27);
```



```

        this->button8->TabIndex = 8;
        this->button8->Tag = L"5";
        this->button8->Text = L"5";
        this->button8->UseVisualStyleBackColor = true;
        this->button8->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
        //
        // button9
        //
        this->button9->Location = System::Drawing::Point(102, 113);
        this->button9->Name = L"button9";
        this->button9->Size = System::Drawing::Size(32, 27);
        this->button9->TabIndex = 9;
        this->button9->Tag = L"9";
        this->button9->Text = L"9";
        this->button9->UseVisualStyleBackColor = true;
        this->button9->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
        //
        // button10
        //
        this->button10->Location = System::Drawing::Point(62, 183);
        this->button10->Name = L"button10";
        this->button10->Size = System::Drawing::Size(32, 27);
        this->button10->TabIndex = 10;
        this->button10->Tag = L"2";
        this->button10->Text = L"2";
        this->button10->UseVisualStyleBackColor = true;
        this->button10->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
        //
        // button11
        //
        this->button11->Location = System::Drawing::Point(102, 216);
        this->button11->Name = L"button11";
        this->button11->Size = System::Drawing::Size(32, 27);
        this->button11->TabIndex = 11;
        this->button11->Text = L"-/+";
        this->button11->UseVisualStyleBackColor = true;
        this->button11->Click += gcnew System::EventHandler(this,
&Form1::button11_Click);
        //
        // button12
        //
        this->button12->Location = System::Drawing::Point(62, 216);

```



```
this->button12->Name = L"button12";
this->button12->Size = System::Drawing::Size(32, 27);
this->button12->TabIndex = 12;
this->button12->Text = L".";
this->button12->UseVisualStyleBackColor = true;
this->button12->Click += gcnew System::EventHandler(this,
&Form1::button12_Click);
//
// button13
//
this->button13->Location = System::Drawing::Point(185, 113);
this->button13->Name = L"button13";
this->button13->Size = System::Drawing::Size(40, 27);
this->button13->TabIndex = 13;
this->button13->Tag = L"s";
this->button13->Text = L"sin x";
this->button13->UseVisualStyleBackColor = true;
this->button13->Click += gcnew System::EventHandler(this,
&Form1::button13_Click);
//
// button14
//
this->button14->Location = System::Drawing::Point(185, 148);
this->button14->Name = L"button14";
this->button14->Size = System::Drawing::Size(40, 27);
this->button14->TabIndex = 14;
this->button14->Tag = L"c";
this->button14->Text = L"cos x";
this->button14->UseVisualStyleBackColor = true;
this->button14->Click += gcnew System::EventHandler(this,
&Form1::button14_Click);
//
// button15
//
this->button15->Location = System::Drawing::Point(185, 183);
this->button15->Name = L"button15";
this->button15->Size = System::Drawing::Size(40, 27);
this->button15->TabIndex = 15;
this->button15->Tag = L"t";
this->button15->Text = L"tg x";
this->button15->UseVisualStyleBackColor = true;
this->button15->Click += gcnew System::EventHandler(this,
&Form1::button15_Click);
//
// button16
```

```

//
this->button16->Location = System::Drawing::Point(185, 218);
this->button16->Name = L"button16";
this->button16->Size = System::Drawing::Size(40, 27);
this->button16->TabIndex = 16;
this->button16->Tag = L"k";
this->button16->Text = L"ctg x";
this->button16->UseVisualStyleBackColor = true;
this->button16->Click += gcnew System::EventHandler(this,
&Form1::button16_Click);
//
// button17
//
this->button17->Location = System::Drawing::Point(233, 113);
this->button17->Name = L"button17";
this->button17->Size = System::Drawing::Size(40, 27);
this->button17->TabIndex = 17;
this->button17->Tag = L"q";
this->button17->Text = L"SQRT";
this->button17->UseVisualStyleBackColor = true;
this->button17->Click += gcnew System::EventHandler(this,
&Form1::button17_Click);
//
// button18
//
this->button18->Location = System::Drawing::Point(233, 148);
this->button18->Name = L"button18";
this->button18->Size = System::Drawing::Size(40, 27);
this->button18->TabIndex = 18;
this->button18->Tag = L"^";
this->button18->Text = L"x^y";
this->button18->UseVisualStyleBackColor = true;
this->button18->Click += gcnew System::EventHandler(this,
&Form1::button18_Click);
//
// button19
//
this->button19->Location = System::Drawing::Point(233, 183);
this->button19->Name = L"button19";
this->button19->Size = System::Drawing::Size(40, 27);
this->button19->TabIndex = 19;
this->button19->Tag = L"l";
this->button19->Text = L"ln x";
this->button19->UseVisualStyleBackColor = true;

```

```

        this->button19->Click += gcnew System::EventHandler(this,
&Form1::button19_Click);
//
// radioButton1
//
this->radioButton1->AutoSize = true;
this->radioButton1->Checked = true;
this->radioButton1->Location = System::Drawing::Point(89, 72);
this->radioButton1->Name = L"radioButton1";
this->radioButton1->Size = System::Drawing::Size(45, 17);
this->radioButton1->TabIndex = 21;
this->radioButton1->TabStop = true;
this->radioButton1->Text = L"Rad";
this->radioButton1->UseVisualStyleBackColor = true;

//
// radioButton2
//
this->radioButton2->AutoSize = true;
this->radioButton2->Location = System::Drawing::Point(22, 72);
this->radioButton2->Name = L"radioButton2";
this->radioButton2->Size = System::Drawing::Size(59, 17);
this->radioButton2->TabIndex = 22;
this->radioButton2->Text = L"Gradus";
this->radioButton2->UseVisualStyleBackColor = true;

//
// button20
//
this->button20->Location = System::Drawing::Point(142, 113);
this->button20->Name = L"button20";
this->button20->Size = System::Drawing::Size(40, 27);
this->button20->TabIndex = 23;
this->button20->Tag = L"/";
this->button20->Text = L"/";
this->button20->UseVisualStyleBackColor = true;
this->button20->Click += gcnew System::EventHandler(this,
&Form1::button20_Click);
//
// button21
//
this->button21->Location = System::Drawing::Point(142, 148);
this->button21->Name = L"button21";
this->button21->Size = System::Drawing::Size(40, 27);
this->button21->TabIndex = 24;

```

```

        this->button21->Tag = L"*";
        this->button21->Text = L"*";
        this->button21->UseVisualStyleBackColor = true;
        this->button21->Click += gcnew System::EventHandler(this,
&Form1::button21_Click);
        //
        // button22
        //
        this->button22->Location = System::Drawing::Point(142, 183);
        this->button22->Name = L"button22";
        this->button22->Size = System::Drawing::Size(40, 27);
        this->button22->TabIndex = 25;
        this->button22->Tag = L"-";
        this->button22->Text = L"-";
        this->button22->UseVisualStyleBackColor = true;
        this->button22->Click += gcnew System::EventHandler(this,
&Form1::button22_Click);
        //
        // button23
        //
        this->button23->Location = System::Drawing::Point(142, 218);
        this->button23->Name = L"button23";
        this->button23->Size = System::Drawing::Size(40, 27);
        this->button23->TabIndex = 26;
        this->button23->Tag = L"+";
        this->button23->Text = L"+";
        this->button23->UseVisualStyleBackColor = true;
        this->button23->Click += gcnew System::EventHandler(this,
&Form1::button23_Click);
        //
        // button24
        //
        this->button24->Location = System::Drawing::Point(233, 218);
        this->button24->Name = L"button24";
        this->button24->Size = System::Drawing::Size(40, 27);
        this->button24->TabIndex = 27;
        this->button24->Text = L="";
        this->button24->UseVisualStyleBackColor = true;
        this->button24->Click += gcnew System::EventHandler(this,
&Form1::button24_Click);
        //
        // button25
        //
        this->button25->Location = System::Drawing::Point(185, 84);
        this->button25->Name = L"button25";

```

```

        this->button25->Size = System::Drawing::Size(40, 27);
        this->button25->TabIndex = 28;
        this->button25->Text = L"C";
        this->button25->UseVisualStyleBackColor = true;
        this->button25->Click += gcnew System::EventHandler(this,
&Form1::button25_Click);
        //
        // button26
        //
        this->button26->Location = System::Drawing::Point(233, 84);
        this->button26->Name = L"button26";
        this->button26->Size = System::Drawing::Size(40, 27);
        this->button26->TabIndex = 29;
        this->button26->Text = L"Off";
        this->button26->UseVisualStyleBackColor = true;
        this->button26->Click += gcnew System::EventHandler(this,
&Form1::button26_Click);
        //
        // Form1
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
        this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(292, 266);
        this->Controls->Add(this->button26);
        this->Controls->Add(this->button25);
        this->Controls->Add(this->button24);
        this->Controls->Add(this->button23);
        this->Controls->Add(this->button22);
        this->Controls->Add(this->button21);
        this->Controls->Add(this->button20);
        this->Controls->Add(this->radioButton2);
        this->Controls->Add(this->radioButton1);
        this->Controls->Add(this->button19);
        this->Controls->Add(this->button18);
        this->Controls->Add(this->button17);
        this->Controls->Add(this->button16);
        this->Controls->Add(this->button15);
        this->Controls->Add(this->button14);
        this->Controls->Add(this->button13);
        this->Controls->Add(this->button12);
        this->Controls->Add(this->button11);
        this->Controls->Add(this->button10);
        this->Controls->Add(this->button9);
        this->Controls->Add(this->button8);

```

```
this->Controls->Add(this->button7);
this->Controls->Add(this->button6);
this->Controls->Add(this->button5);
this->Controls->Add(this->button4);
this->Controls->Add(this->button3);
this->Controls->Add(this->button2);
this->Controls->Add(this->button1);
this->Controls->Add(this->textBox1);
this->Name = L"Form1";
this->Text = L"Form1";
this->ResumeLayout(false);
this->PerformLayout();
```

```
}
```

#pragma endregion

#pragma region kalkulyatorning asosiy fuksiyalari

```
//yordamchi o`zgaruvchilar
```

```
static bool test=true;// tugma birinchi marta bosilishini tekshirish uchun
```

kerak

```
static bool testnuqta=true;//nuqta bir marta yozilishi uchun kerak
```

```
static bool testIshora=true;// ishora bir marta yozilishi uchun kerak
```

```
static double temp=0;//birinchi sonli qiymatni saqlab turish uchun kerak
```

```
static char amal;// matematik amallarni farqlash uchun kerak
```

```
private: System::Void button5_Click(System::Object^ sender, System::EventArgs^ e)
```

```
{
```

```
    if(test)
```

```
    {
```

```
        textBox1->Text=this->ActiveControl->Text;//bosilgan ixtiyoriy
```

tugmaning

```
        //ustidagi yozuvini yozuv oynasiga chiqaradi(yozadi)
```

```
        test=false;//Birinchi sonli qiymat yozilganini bildiradi 0dan
```

tashqari

```
        // bu 2 ta 0 soni yoki yozuv oynasidagi sonni ketidan yozib ketish
```

uchun

```
        //kerak
```

```
    }
```

```
    else
```

```
        textBox1->Text+=this->ActiveControl->Text;//bosilgan
```

ixtiyoriy tugmaning

```
        //ustidagi yozuvini yozuv oynasidagi yozuvga qo'shib yozadi
```

```
    }
```

```

private: System::Void button6_Click(System::Object^ sender, System::EventArgs^
e)
    {
        if(test)
            textBox1->Text=this->ActiveControl->Text;//bosilgan
ixtiyoriy tugmaning
            //ustidagi yozuvini yozuv oynasiga chiqaradi(yozadi)
        else
            textBox1->Text+=this->ActiveControl->Text;//bosilgan
ixtiyoriy tugmaning
            //ustidagi yozuvini yozuv oynasidagi yozuviga qo`shib yozadi
    }
private: System::Void button12_Click(System::Object^ sender, System::EventArgs^
e)
    {
        if(testnuqta)
        {
            textBox1->Text+=",";//yozuv oynasida vergul belgisini
qo`yadi
            test=false;//ikkinchi marta qo`yilishini oldini oladi
        }
        testnuqta=false;//ikkinchi marta qo`yilishini oldini oladi
    }
private: System::Void button11_Click(System::Object^ sender, System::EventArgs^
e)
    {
        if(testIshora)
        {
            textBox1->Text="-"+textBox1->Text;//yozuv oynasida
minus belgisini qo`yadi
            testIshora=false;//ikkinchi marta qo`yilishini oldini oladi
        }
        else
        {
            textBox1->Text=textBox1->Text->Replace("-", "");//yozuv
oynasidagi minus belgisini o`chiradi
            testIshora=true;//minus belgisini yozishga ruxsat beradi
        }
    }
private: System::Void button24_Click(System::Object^ sender, System::EventArgs^
e)
    {

```

`double a=Convert::ToDouble(textBox1->Text);`//yozuv oynasidagi sonni a o'zgaruvchiga //o'zlashtiradi uni matnli qiymatini o'zgaruvchining turiga mos ravishda o'zgartirib o'zlashtiradi.

`switch(amal)`//amalnig qiymatiga mos ravishdagi arifmetik amallarni bajarish

`//siklinig boshlanishi`

`{`

`case '+':` textBox1->Text=(temp+a).ToString();//amal qiymati qo'shish bo'lganda qo'shish amali bajariladi
`break;`//sikl to'xtatilib undan chiqish

`case '-':` textBox1->Text=(temp-a).ToString();//amal qiymati ayirish bo'lganda ayirish amali bajariladi
`break;`//sikl to'xtatilib undan chiqish

`case '*':` textBox1->Text=(temp*a).ToString();//amal qiymati ko'paytirish bo'lganda ko'paytirish amali bajariladi
`break;`//sikl to'xtatilib undan chiqish

`case '/':` textBox1->Text=(temp/a).ToString();//amal qiymati bo'lish bo'lganda bo'lish amali bajariladi
`break;`//sikl to'xtatilib undan chiqish

`case '^':` textBox1->Text=pow(temp,a).ToString();//amal qiymati darajaga oshirish bo'lganda darajaga oshirish amali bajariladi
`break;`//sikl to'xtatilib undan chiqish

`default:` `break;`//amalnig boshqa barcha hollarida ham sikl to'xtatiladi va undan chiqiladi.

`}`

`temp=Convert::ToDouble(textBox1->Text);`//yozuv oynasidagi sonni temp o'zgaruvchisiga //o'zlashtiradi uni matnli qiymatini o'zgaruvchining turiga mos ravishda o'zgartirib o'zlashtiradi.

`}`

`private:` System::Void button20_Click(System::Object^ sender, System::EventArgs^ e)

`{`

`//bo'lish tugmasi bosilganda quyidagi funksiyalar bajariladi`
`button24_Click(sender,e);`//teng tugamasidagi funksiyalar to'plami bajariladi

`test=true;`//ikkinchi sonli qiymatni yozuv oynasiga kiritishni belgilaydi

`amal='/';`//amal o'zgaruvchisiga bo'lish amali o'zlashtiriladi

`}`

`private:` System::Void button13_Click(System::Object^ sender, System::EventArgs^ e)


```

    {
        test=true;//ikkinchi sonli qiymatni yozuv oynasiga kiritishni
belgilaydi
        button24_Click(sender,e);//teng tugamsidagi funksiyalar to`plami
bajariladi
        double a=Convert::ToDouble(textBox1->Text);//yozuv
oynasidagi sonni a uzgaruvchisiga
        //o`zlashtiradi uni matnli qiymatini o`zgaruvchining turiga mos
ravishda o`zgartirib o`zlashtiradi.
        /*Trigonometrik funksiyalarnig qiymati radianlarda berilishi
kerak. Chunki, dasturlash tilida ularni radianlar qiymatida hisoblaydi va shu sababli
foydalanuvchi sanoq tizimini ajratish uchun 2 ta belgilanadigan tugmalar kalkulyator
oynasiga o`rnatilgan
        Bu tugmalar foydalanuvchi qaysi sanoq tizimida iwlayapganini
ajratadi va mos ravishda gradusdan radianga o`tish amali bajariladi.*/
        if(radioButton1->Checked)// Rad tugmasi tanlangan bo`lsa
kiritilgan qiymatlar //o`zgartirilmay tregonametrik funksiya hisoblanadi.
        textBox1->Text=(sin(a)).ToString();//sinusni hisoblash funksiyasi
        else// aks holda ya`ni Gradus tanlangan bo`lsa kiritilgan qiymatlar
mos ravishda radianga //o`zgartirish amali amalga oshiriladi.
        textBox1->Text=(sin(a*Math::PI/180)).ToString();//gradus
qiymatidan radian qiymatiga o`tkazib //sinusni hisoblash fuksiyasi
        temp=Convert::ToDouble(textBox1->Text);//yozuv oynasidagi
sonni temp uzgaruvchisiga //o`zlashtiradi uni matnli qiymatini o`zgaruvchining
turiga mos ravishda o`zgartirib o`zlashtiradi.
    }
private: System::Void button25_Click(System::Object^ sender, System::EventArgs^
e)
    {
        //bu tugma hamma yordamchi o`zgaruvchilar va yozuv oynasidagi
matnni dastlabki //holatga keltiradi ya`ni boshlang`ich qiymatlarni o`zlashtiradi.
        temp=0.0;//temp o`zgaruvchisinig qiymatini 0 ga tenglash ya`ni
dastlabki holga keltirish
        textBox1->Text="0";//yozuv oynasiga boshlang`ich (0)yozuvni
yozadi
        amal=' ';//amalda amallarni o`chirib bo`sh holga keltiradi
        test=true;//yangi qiymat kiritish holiga keltiradi.
        testIshora=true;//ishora qo`yish mumkin bo`ladi
        testnuqta=true;//nuqta qo`yish ya`ni butun va kasr qismini
ajratish.
    }
private: System::Void button26_Click(System::Object^ sender, System::EventArgs^
e)
    {

```

```

        this->Close();//kalkulyator dasturini tugatadi va uni ishini
to`xtatib yopadi
    }
#pragma endregion
    //qolgan tugmalarning fuksiyalari ham yu`qoridagi tugmalar funksiyalari
kabi yoziladi faqat //tugmaning bajaradigan amaliga mos ravishda yozilishi kerak.
private: System::Void button21_Click(System::Object^ sender, System::EventArgs^
e)
    {
        button24_Click(sender,e);
        test=true;
        amal='*';
    }
private: System::Void button22_Click(System::Object^ sender, System::EventArgs^
e)
    {
        button24_Click(sender,e);
        test=true;
        amal='-';
    }
private: System::Void button23_Click(System::Object^ sender, System::EventArgs^
e)
    {
        button24_Click(sender,e);
        test=true;
        amal='+';
    }
private: System::Void button18_Click(System::Object^ sender, System::EventArgs^
e)
    {
        button24_Click(sender,e);
        test=true;
        amal='^';
    }
private: System::Void button14_Click(System::Object^ sender, System::EventArgs^
e)
    {
        //cosinusni hisoblash
        test=true;
        button24_Click(sender,e);
        double a=Convert::ToDouble(textBox1->Text);
        if(radioButton1->Checked)
            textBox1->Text=(cos(a)).ToString();
        else

```

```

        textBox1->Text=(cos(a*Math::PI/180)).ToString();

        temp=Convert::ToDouble(textBox1->Text);
    }
private: System::Void button15_Click(System::Object^ sender, System::EventArgs^
e)
    {
        //tangensni hisoblash
        test=true;
        button24_Click(sender,e);
        double a=Convert::ToDouble(textBox1->Text);
        if(radioButton1->Checked)
            textBox1->Text=(tan(a)).ToString();
        else
            textBox1->Text=(1/tan(a*Math::PI/180)).ToString();

        temp=Convert::ToDouble(textBox1->Text);
    }
private: System::Void button16_Click(System::Object^ sender, System::EventArgs^
e)
    {
        //catangensni hisoblash
        test=true;
        button24_Click(sender,e);
        double a=Convert::ToDouble(textBox1->Text);
        if(radioButton1->Checked)
            textBox1->Text=(1/tan(a)).ToString();
        else
            textBox1->Text=(1/tan(a*Math::PI/180)).ToString();
        temp=Convert::ToDouble(textBox1->Text);
    }
private: System::Void button17_Click(System::Object^ sender, System::EventArgs^
e)
    {
        //ildizni hisoblash
        test=true;
        button24_Click(sender,e);
        double a=Convert::ToDouble(textBox1->Text);
        textBox1->Text=sqrt(a).ToString();
        temp=Convert::ToDouble(textBox1->Text);
    }
private: System::Void button19_Click(System::Object^ sender, System::EventArgs^
e)
    {
        //ln ni hisoblash

```

```

        test=true;
        button24_Click(sender,e);
        double a=Convert::ToDouble(textBox1->Text);
        textBox1->Text=log(a).ToString();
        temp=Convert::ToDouble(textBox1->Text);
    }
};
}

```

6- ILOVA

```

#include<stdio.h>//Kiritish chiqarish funksiyalar to`plami
#include<math.h>//matematik funksiyalar to`plami
int main();//asosiy va bosh funksiya
{//boshlandi
    printf("6 - topshiriqni yechish dasturi.\n\n");//ekranga quyidagi yozuvni
    chiqaradi
    double S,p,a=3.6,b=4.8,c=3.4;//o`sgaruvchilarni e`lon qilish va qiymatlarni
    o`zlashtirish
    p=(a+b+c)/2;//to`sinning perimetrini hisoblash
    S=sqrt(p*(p-a)*(p-b)*(p-c));//to`sinning yuzini hisoblash
    printf("To`sing ko`ndalang kesim yuzasi\n S = %f\n",S);//ekranga to`sin
    yuzining qiymatini chop ettirish
    getch();//dastur yechimini ko`rish uchun pauza vazifasini bajaradi. Ya`ni
    ixtiyoriy qiymat //kiritilmaguncha va Enter tugmasi bosilmaguncha dastur ishlab
    turadi.
    return 0;//asosiy funksiyaning qiymatini 0 ga tenglashtiradi va funksiyaning
    yopadi
    //Tamom.
}

```

7- ILOVA

```

#pragma once
#include <math.h>

namespace Topshiriq7 {

    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;

```

```

using namespace System::Drawing;

/// <summary>
/// Summary for Form1
///
/// WARNING: If you change the name of this class, you will need to change
the
///     'Resource File Name' property for the managed resource compiler tool
///     associated with all .resx files this class depends on. Otherwise,
///     the designers will not be able to interact properly with localized
///     resources associated with this form.
/// </summary>
public ref class Form1 : public System::Windows::Forms::Form
{
public:
    Form1(void)
    {
        InitializeComponent();
        //
        //TODO: Add the constructor code here
        //
    }

protected:
    /// <summary>
    /// Clean up any resources being used.
    /// </summary>
    ~Form1()
    {
        if (components)
        {
            delete components;
        }
    }

private: System::Windows::Forms::PictureBox^ pictureBox1;
private: System::Windows::Forms::Button^ button1;
private: System::Windows::Forms::Button^ button2;
private: System::Windows::Forms::Button^ button3;
private: System::Windows::Forms::ColorDialog^ colorDialog1;
protected:

private:
    /// <summary>
    /// Required designer variable.
    /// </summary>

```

```
System::ComponentModel::Container ^components;
```

```
#pragma region Windows Form Designer generated code
```

```
/// <summary>
```

```
/// Required method for Designer support - do not modify
```

```
/// the contents of this method with the code editor.
```

```
/// </summary>
```

```
void InitializeComponent(void)
```

```
{
```

```
    this->pictureBox1 = (gcnew
```

```
System::Windows::Forms::PictureBox());
```

```
    this->button1 = (gcnew System::Windows::Forms::Button());
```

```
    this->button2 = (gcnew System::Windows::Forms::Button());
```

```
    this->button3 = (gcnew System::Windows::Forms::Button());
```

```
    this->colorDialog1 = (gcnew
```

```
System::Windows::Forms::ColorDialog());
```

```
    (cli::safe_cast<System::ComponentModel::ISupportInitialize^
```

```
>(this->pictureBox1))->BeginInit();
```

```
    this->SuspendLayout();
```

```
    //
```

```
    // pictureBox1
```

```
    //
```

```
    this->pictureBox1->Location = System::Drawing::Point(12, 12);
```

```
    this->pictureBox1->Name = L"pictureBox1";
```

```
    this->pictureBox1->Size = System::Drawing::Size(468, 313);
```

```
    this->pictureBox1->TabIndex = 0;
```

```
    this->pictureBox1->TabStop = false;
```

```
    //
```

```
    // button1
```

```
    //
```

```
    this->button1->Location = System::Drawing::Point(504, 74);
```

```
    this->button1->Name = L"button1";
```

```
    this->button1->Size = System::Drawing::Size(75, 23);
```

```
    this->button1->TabIndex = 1;
```

```
    this->button1->Text = L"Chizish";
```

```
    this->button1->UseVisualStyleBackColor = true;
```

```
    this->button1->Click += gcnew System::EventHandler(this,
```

```
&Form1::button1_Click);
```

```
    //
```

```
    // button2
```

```
    //
```

```
    this->button2->Location = System::Drawing::Point(504, 113);
```

```
    this->button2->Name = L"button2";
```

```
    this->button2->Size = System::Drawing::Size(75, 23);
```

```
    this->button2->TabIndex = 2;
```

```

        this->button2->Text = L"Rang";
        this->button2->UseVisualStyleBackColor = true;
        this->button2->Click += gnew System::EventHandler(this,
&Form1::button2_Click);
        //
        // button3
        //
        this->button3->Location = System::Drawing::Point(504, 152);
        this->button3->Name = L"button3";
        this->button3->Size = System::Drawing::Size(75, 23);
        this->button3->TabIndex = 3;
        this->button3->Text = L"Chiqish";
        this->button3->UseVisualStyleBackColor = true;
        this->button3->Click += gnew System::EventHandler(this,
&Form1::button3_Click);
        //
        // Form1
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
        this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(604, 380);
        this->Controls->Add(this->button3);
        this->Controls->Add(this->button2);
        this->Controls->Add(this->button1);
        this->Controls->Add(this->pictureBox1);
        this->Name = L"Form1";
        this->Text = L"Grafik chizish";
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^
>(this->pictureBox1))->EndInit();
        this->ResumeLayout(false);

    }
#pragma endregion
    static Color rang=Color::Black;
    private: System::Void button3_Click(System::Object^ sender,
System::EventArgs^ e)
    {
        this->Close();
    }
    private: System::Void button2_Click(System::Object^ sender,
System::EventArgs^ e)
    {
        if(colorDialog1->ShowDialog() ==
System::Windows::Forms::DialogResult::OK)

```

```

        {
            rang=colorDialog1->Color;
        }
    }
private: System::Void button1_Click(System::Object^ sender, System::EventArgs^
e)
    {
        Bitmap ^ chizish_muxiti = gcnew Bitmap(pictureBox1-
>Width,pictureBox1->Height);
        Graphics^ chiz=Graphics::FromImage(chizish_muxiti);
        Pen^ qalam = gcnew Pen( rang,2.0 );
        float y1=0,y2=0;
        for(float x=-1.0;x<0;x+=0.1)
        {
            y1= 10*pow(x,2)/(x+sqrt(x+1));
            y2= 10*pow(x+0.1,2)/(x+0.1+sqrt(x+0.1+1));

            chiz->DrawLine(qalam,pictureBox1-
>Width/2+10*x,pictureBox1->Height/2+y1,pictureBox1->Width/2+10*(x-
0.1),pictureBox1->Height/2+y2);
        }
        for(float x=0;x<1;x+=0.2)
        {
            y1= 10*pow(x,2)/(x+sqrt(x+1));
            y2= 10*pow(x+0.2,2)/(x+0.2+sqrt(x+0.2+1));

            chiz->DrawLine(qalam,pictureBox1-
>Width/2+10*x,pictureBox1->Height/2+y1,pictureBox1->Width/2+10*(x-
0.2),pictureBox1->Height/2+y2);
        }
        for(float x=1;x<3;x+=0.4)
        {
            y1= 10*pow(x,2)/(x+sqrt(x+1));
            y2= 10*pow(x+0.4,2)/(x+0.4+sqrt(x+0.4+1));

            chiz->DrawLine(qalam,pictureBox1-
>Width/2+10*x,pictureBox1->Height/2+y1,pictureBox1->Width/2+10*(x-
0.4),pictureBox1->Height/2+y2);
        }
        pictureBox1->Image=chizish_muxiti;
    }
};
}

```


8- ILOVA

```
#pragma once namespace Topshiriq8 {
    using namespace System;
    using namespace System::ComponentModel;
    using namespace System::Collections;
    using namespace System::Windows::Forms;
    using namespace System::Data;
    using namespace System::Drawing;

    /// <summary>
    /// Summary for Form1
    ///
    /// WARNING: If you change the name of this class, you will need to change
the
    /// 'Resource File Name' property for the managed resource compiler tool
    /// associated with all .resx files this class depends on. Otherwise,
    /// the designers will not be able to interact properly with localized
    /// resources associated with this form.
    /// </summary>
    public ref class Form1 : public System::Windows::Forms::Form
    {
    public:
        Form1(void)
        {
            InitializeComponent();
            //
            //TODO: Add the constructor code here
            //
        }

    protected:
        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        ~Form1()
        {
            if (components)
            {
                delete components;
            }
        }

    private: System::Windows::Forms::PictureBox^ pictureBox1;
    protected:
    private: System::Windows::Forms::Button^ button1;
```

```
private: System::Windows::Forms::Button^ button2;
private: System::Windows::Forms::Button^ button3;
private: System::Windows::Forms::Button^ button4;
private: System::Windows::Forms::Button^ button5;
private: System::Windows::Forms::Button^ button6;
private: System::Windows::Forms::ColorDialog^ colorDialog1;
```

```
private:
    /// <summary>
    /// Required designer variable.
    /// </summary>
    System::ComponentModel::Container ^components;
```

#pragma region Windows Form Designer generated code

```
    /// <summary>
    /// Required method for Designer support - do not modify
    /// the contents of this method with the code editor.
    /// </summary>
    void InitializeComponent(void)
    {
        this->pictureBox1 = (gcnew
System::Windows::Forms::PictureBox());
        this->button1 = (gcnew System::Windows::Forms::Button());
        this->button2 = (gcnew System::Windows::Forms::Button());
        this->button3 = (gcnew System::Windows::Forms::Button());
        this->button4 = (gcnew System::Windows::Forms::Button());
        this->button5 = (gcnew System::Windows::Forms::Button());
        this->button6 = (gcnew System::Windows::Forms::Button());
        this->colorDialog1 = (gcnew
System::Windows::Forms::ColorDialog());
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^
>(this->pictureBox1))->BeginInit();
        this->SuspendLayout();
        //
        // pictureBox1
        //
        this->pictureBox1->Location = System::Drawing::Point(10, 12);
        this->pictureBox1->Name = L"pictureBox1";
        this->pictureBox1->Size = System::Drawing::Size(470, 406);
        this->pictureBox1->TabIndex = 0;
        this->pictureBox1->TabStop = false;
        //
        // button1
        //
        this->button1->Location = System::Drawing::Point(512, 69);
```

```
this->button1->Name = L"button1";
this->button1->Size = System::Drawing::Size(85, 23);
this->button1->TabIndex = 1;
this->button1->Text = L"aylana chzish";
this->button1->UseVisualStyleBackColor = true;
this->button1->Click += gcnew System::EventHandler(this,
&Form1::button1_Click);
//
// button2
//
this->button2->Location = System::Drawing::Point(512, 98);
this->button2->Name = L"button2";
this->button2->Size = System::Drawing::Size(85, 23);
this->button2->TabIndex = 2;
this->button2->Text = L"yoy chzish";
this->button2->UseVisualStyleBackColor = true;
this->button2->Click += gcnew System::EventHandler(this,
&Form1::button2_Click);
//
// button3
//
this->button3->Location = System::Drawing::Point(512, 129);
this->button3->Name = L"button3";
this->button3->Size = System::Drawing::Size(85, 23);
this->button3->TabIndex = 3;
this->button3->Text = L"sektor chzish";
this->button3->UseVisualStyleBackColor = true;
this->button3->Click += gcnew System::EventHandler(this,
&Form1::button3_Click);
//
// button4
//
this->button4->Location = System::Drawing::Point(512, 160);
this->button4->Name = L"button4";
this->button4->Size = System::Drawing::Size(85, 23);
this->button4->TabIndex = 4;
this->button4->Text = L"eleps chzish";
this->button4->UseVisualStyleBackColor = true;
this->button4->Click += gcnew System::EventHandler(this,
&Form1::button4_Click);
//
// button5
//
this->button5->Location = System::Drawing::Point(512, 191);
this->button5->Name = L"button5";
```

```

        this->button5->Size = System::Drawing::Size(85, 23);
        this->button5->TabIndex = 5;
        this->button5->Text = L"rang tanlash";
        this->button5->UseVisualStyleBackColor = true;
        this->button5->Click += gcnew System::EventHandler(this,
&Form1::button5_Click);
        //
        // button6
        //
        this->button6->Location = System::Drawing::Point(512, 222);
        this->button6->Name = L"button6";
        this->button6->Size = System::Drawing::Size(85, 23);
        this->button6->TabIndex = 6;
        this->button6->Text = L"chiqish";
        this->button6->UseVisualStyleBackColor = true;
        this->button6->Click += gcnew System::EventHandler(this,
&Form1::button6_Click);
        //
        // Form1
        //
        this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);
        this->AutoScaleMode =
System::Windows::Forms::AutoScaleMode::Font;
        this->ClientSize = System::Drawing::Size(609, 427);
        this->Controls->Add(this->button6);
        this->Controls->Add(this->button5);
        this->Controls->Add(this->button4);
        this->Controls->Add(this->button3);
        this->Controls->Add(this->button2);
        this->Controls->Add(this->button1);
        this->Controls->Add(this->pictureBox1);
        this->Name = L"Form1";
        this->Text = L"Form1";
        (cli::safe_cast<System::ComponentModel::ISupportInitialize^
>(this->pictureBox1))->EndInit();
        this->ResumeLayout(false);

    }
#pragma endregion
    static Color rang=Color::Black;
    private: System::Void button1_Click(System::Object^ sender,
System::EventArgs^ e)
    {
        //chizish muxitini yaratish

```

```

        Bitmap ^ chizish_muxiti = gcnw Bitmap(pictureBox1-
>Width,pictureBox1->Height);
        Graphics^ chiz=Graphics::FromImage(chizish_muxiti);//chizish
classining obektini yaratish
        Pen^ qalam = gcnw Pen( rang,2.0 );//chizish qalamini va rangini
yaratish

        // aylana chizish
        chiz->DrawArc(qalam,100,100,70,70,0,360);//aylana chizish

        pictureBox1->Image=chizish_muxiti;//chizish oynasiga chizilgan
grafikni chiqarish
    }
private: System::Void button2_Click(System::Object^ sender, System::EventArgs^
e)
    {
        Bitmap ^ chizish_muxiti = gcnw Bitmap(pictureBox1-
>Width,pictureBox1->Height);
        Graphics^ chiz=Graphics::FromImage(chizish_muxiti);//chizish
classining obektini yaratish
        Pen^ qalam = gcnw Pen( rang,2.0 );//chizish qalamini va rangini
yaratish

        // aylana chizish
        chiz->DrawArc(qalam,100,100,70,70,10,58);//yoy chizish

        pictureBox1->Image=chizish_muxiti;//chizish oynasiga chizilgan
grafikni chiqarish
    }
private: System::Void button3_Click(System::Object^ sender, System::EventArgs^
e)
    {
        Bitmap ^ chizish_muxiti = gcnw Bitmap(pictureBox1-
>Width,pictureBox1->Height);
        Graphics^ chiz=Graphics::FromImage(chizish_muxiti);//chizish
classining obektini yaratish
        Pen^ qalam = gcnw Pen( rang,2.0 );//chizish qalamini va rangini
yaratish

        // aylana chizish
        chiz->DrawPie(qalam,100,100,70,70,10,58);//sektor chizish

        pictureBox1->Image=chizish_muxiti;//chizish oynasiga chizilgan
grafikni chiqarish
    }
}

```

```

private: System::Void button4_Click(System::Object^ sender, System::EventArgs^
e)
    {
        Bitmap ^ chizish_muxiti = gcnew Bitmap(pictureBox1-
>Width,pictureBox1->Height);
        Graphics^ chiz=Graphics::FromImage(chizish_muxiti);//chizish
classining obektini yaratish
        Pen^ qalam = gcnew Pen( rang,2.0 );//chizish qalamini va rangini
yaratish

        // aylana chizish
        chiz->DrawEllipse(qalam,100,100,170,70);//elips chizish

        pictureBox1->Image=chizish_muxiti;//chizish oynasiga chizilgan
grafikni chiqarish
    }
private: System::Void button5_Click(System::Object^ sender, System::EventArgs^
e)
    {
        //rang tanlash dialogini ochish
        if(colorDialog1->ShowDialog() ==
System::Windows::Forms::DialogResult::OK)
            {
                rang=colorDialog1->Color;//tanlangan rangning
qiymatini o`zlashtirish
            }
    }
private: System::Void button6_Click(System::Object^ sender, System::EventArgs^
e)
    {
        this->Close();//dasturni yopish
    }
};
}

```