

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

URGANCH DAVLAT UNIVERSITETI

B. Razzaqov, S. K. Jafarov, S. A. Matkarimov

**C++ DASTURLASH TILIDA
KURS ISHLARINI BAJARISH BO‘YICHA
USLUBIY QO‘LLANMA**

O`quv-uslubiy qo`llanma Urganch davlat universiteti ilmiy-uslubiy kengashining 2017 yil 23-martdagi 5-sonli yig`ilishi bayonnomasi qaroriga asosan nashrga tavsiya etilgan.

Urganch-2017

SO‘Z BOSHI

O‘zbekiston Respublikasi Prezidenti I.A.Karimov «Barkamol avlod — O‘zbekiston taraqqiyotining poydevori», deb bejiz aytmaganlar. Bu avlod kelajakni quruvchi, yaratuvchi, rivojlantiruvchi va bizning otabobolarimiz, biz boshlagan ishlarning davomchisi, taraqqiy ettiruvchi, dunyo miqyosida tanituvchi vorislarimizdir. SHularni his etgan holda, biz hozirgi hayotimizni, butun borlig‘imizni va bilim salohiyatimizni shunday barkamol avlodni shakllantirishga bag‘ishlashimiz zarur. SHuning uchun xam Prezidentimiz tashabbuslari bilan ta’limni isloh qilishga kirishildi va uning milliy modeli yaratildi. Qabul qilingan «O‘zbekiston Respublikasi Kadrlar tayyorlash Milliy dasturi»da Oliy ta’limning muayyan mutaxassislik bo‘yicha ikki bosqichli - bakalavr hamda magistratura ko‘rinishida kadrlar tayyorlash amalga oshirilishi belgilandi.

O‘zbekiston milliy ensiklopediyasi 1-tomida bakalavr so‘ziga - “Bakalavr (lot. baccalaria – vassalning er-mulki) – ko‘p mamlakatlardagi dastlabki ilmiy daraja. O‘rta asrlarda G‘arbiy Evropa universitetlarida ta’limning birinchi bosqichini tugallagan talabalarga berilgan. O‘zbekiston “Ta’lim to‘g‘risida”gi qonun va “Kadrlar tayyorlash Milliy dasturi” (1997) ga muvofiq universitet va boshqa oliy o‘quv yurtlarining birinchi bosqichi – bakalavriatni tugatganlarga BAKALAVR darajasi beriladi hamda davlat tomonidan tasdiqlangan namunadagi, kasb-hunar faoliyati bilan shug‘ullanish huquqini beradigan diplom topshiriladi. Ba’zi xorijiy mamalakatlar (masalan, Fransiya)da o‘rta ma’lumotga ega bo‘lib, oliy o‘quv yurtiga kirib o‘qish huquqiga ega bo‘lgan shaxs ham Bakalavr deb ataladi” deya izoh berilgan.

O‘zbekiston Respublikasi mustaqillikka erishgandan keyin uzluksiz ta’lim tizimini joriy qilinishi hamda oliy ta’limda Bozor iqtisodiyoti sharoitidan kelib chiqib, kadrlar etkazib berish sifatini jahon talablari asosida tashkillashtirish maqsadida ikki bosqich: bakalavriat va magistratura bosqichlari joriy qilindi.

Bakalavriat – ko‘p mamlakatlarda oliy ta’limning dastlabki bosqichi. O‘zbekistonda Bakalavriat mutaxassisliklar yo‘nalishi bo‘yicha fundamental va amaliy bilim beradigan tayanch oliy ta’limdir. O‘qish muddati kamida 4 yil. Bakalavrluk dasturi tugallangandan so‘ng, bitiruvchilarga bakalavr darajasi berilib davlat attestatsiyasi yakunlariga ko‘ra davlat tomonidan tasdiqlangan namunadagi oliy ma’lumot to‘g‘risidagi diplom beriladi.

Bakalavr DTS da belgilangan maxsus o‘quv rejalari, dasturlari asosida fundamental bilimlarga ega bo‘lib, bitiruv malakaviy ishini

bajarib, himoya qilish, pedagogik bilim va mahoratga, zamonaviy ilmiy-amaliy bilimlarga, informatsion texnologiyalardan yuqori darajada foydalana bilish va kasb-hunar ta'limi yo'nalishida mustaqil faoliyat yuritish, qaror qabul qilish, o'z fikrlarini bayon qila olish qobiliyatlariga ega bo'lishi kerak.

Urganch davlat universiteti kimyo texnika fakultetidagi quyidagi yo'nalishlar bo'yicha bakalavrlar tayyorlashda o'quv rejasiga asosan quyidagi yo'nalishlarda "Informatika va AT" fanidan kurs ishi bajarilishi ko'zda tutilgan:

1. 5310500-"Avtomobilsozlik va traktorsozlik";
2. 5310600- "Yerusti transporti tizimlari va ularning ekspluatatsiyasi";
3. 5410500- "Qishloq xo'jaligi maxsulotlarini saqlash va dastlabki qayta ishlash texnologiyasi;

2001 yildan boshlab bakalavrlarni tayyorlash bo'yicha yangi yaratilgan va tasdiqlangan davlat standartlari, o'quv rejalari va o'quv dasturlari asosida o'qitish tizimiga o'tildi. Davlat standartlari, o'quv rejalari va dasturlar asosan institut mutaxassislik kafedralari tomonidan yaratildi. Ularga binoan bakalavriaturada o'qish jarayonida pedagogik va ilmiy-tadqiqot amaliyotlari bilan birgalikda kurs ishini yozish va himoya qilishga asoslangan.

ALGORITMLASH ASOSLARI

1.1 Algoritm xaqida tushuncha

Algoritm soʻzi nafaqat matematiklar uchun xos boʻlib, balki har qanday axborotni qayta ishlash jarayonlarining asosi boʻlib xisoblanadi. Mos algoritmlarning mavjudligi jarayonlarni avtomatlashtirish imkonini beradi. Algoritm nazariyasi matematik mantiq mantiq bilan birgalikda zamonaviy informatika fanining nazariy asosini hosil qiladi.

Algoritm soʻzi **algoritm** soʻzidan olingan boʻlib, u IX asrning buyuk matematigi bobokalonimiz Muxammad Al-Xorazmiy nomining lotincha shaklidir. Abu Abdulla Muxammad ibn al-Xorazmiy eramizning **783** yilda Xorazmda tavallud topgan. U oʻz davrining buyuk mutafakkir allomasi - matematik, astronom va geograf olimi boʻlgan. Arab raqamlarining sondagi oʻrniga bogʻlik holda amallar bajarish tartibini yagona tizimga keltirgan *мутафаккир*.

Informatika sohasida algoritm tushunchasi asosiy tushuncha boʻlib, u geometriya kursidagi nuqta, toʻgʻri chizik, va tekislik, matematikadagi toʻplam, kimyodagi modda, fizikadagi fazo hamda vaqt tushunchalari kabi fundamental tushuncha hisoblanadi.

Algoritmga aniq, bir taʼrif berish mushkul. Shunday boʻlsada, algoritmning mohiyatini aniq, tushuntirish mumkin. Algoritm – biror masalani echish uchun bajarilishi zarur boʻlgan buyruqlarning tartiblangan ketma-ketligi. Tuzilgan algoritmni uning yozilish qoidalarini tushunadigan va unda koʻrsatilgan buyruqlarni bajarish imkoniga ega boʻlgan insonning oʻzi yoki texnik qurilma (masalan, kompyuter) bajarishi mumkin.

Odamlar har kuni bajaradigan ishlarida oʻzlari bilmagan holda shu ishlarni bajarish algoritmlaridan foydalanadilar. Masalan, kompyuterdan foydalanish, taom tayyorlash, telefondan foydalanish, avtomobilni boshqarish, kitob oʻqish, koʻcha harakati qoidalariga rioya qilish, televizor yoki radiodan foydalanish va hokazo. Albatta, odamlar yuqorida keltirilgan yumushlarni har doim bajarib yurganligi bois, ularni bajarish uchun hech qanday aniq koʻrsatmalarga muhtojlik sezmaydilar. Lekin yuqoridagi yumushlarni birinchi marotaba bajarayotgan odam aniq, koʻrsatmalarsiz uni bajara olmaydi. Masalan, hech qachon kompyuterdan foydalanib koʻrmagan odam aniq, bir koʻrsatmasiz bu ishni bajara olmaydi. Demak, odamlar oʻzlariga tanish boʻlgan ishlarni yoki masalalarni bajarish uchun zarur boʻladigan koʻrsatmalarni qachonlardir, qaerlardandir olganlar yoki oʻrganganlar. Algoritmni bajarishda

ko'rsatmalarni berilgan tartibda bajarish kerak bo'ladi.

1.2 Algoritmning ifodalanishi

Algoritm ijrochisi EHM yoki avtomatik qurilmalar ham bo'lishi mumkin. EHM o'ziga tushunarli bo'lgan tilda tuzilgan algoritm (ya'ni dastur) asosidagina ishlaydi. Demak, EHM uchun tuzilgan dasturni ham to'liq ma'noda **algoritm** deb atash mumkin.

Algoritmni tuzish jarayonida ularning qaysi jrochiga mo'ljallab tuzilayotganligiga alohida e'tibor beriladi va sun'iy yoki tabiiy tillardan birida tuziladi. Ma'lumki, EHM tabiiy tillarni tushunmaydi va inson bilan ular o'rtasida bu tillarda muloqat o'rnatib bo'lmaydi. SHuning uchun inson bilan EHM o'rtasida muloqat o'rnatishga mo'ljallangan bir necha o'nlab tillar yaratilgan. Bu tillar sun'iy tillar jumlasiga kiradi va **dasturlash tillari** deb ataladi. Agar algoritm EHM uchun mo'ljallab tuzilayotgan bo'lsa, mazkur EHM tushunadigan sun'iy tillardan birida tuziladi va to'liq, formallashtiriladi.

Dasturlash muhiti yordamida dastur tuziladi va dasturlar quyidagi qadamlardan iborat:

1. O'ziga xos xususiyatlari (tavsif, aniqlashlar, dastur talabini shakllantirish, matematik modellashtirish);
2. Algoritmni ishlab chiqish;
3. Kodlash;
4. Dasturni ishga tushirish;
5. Dasturni testdan o'tkazish (foydalanish darajasi yoki ...).
6. Dasturni tadbiq qilish;

O'ziga xos xususiyatlari - dasturlashning eng asosiy bosqichlaridan biri bo'lib, boshlang'ich axborot elementlarini to'g'ri shakllantirib olinadi. Bu esa yaxshi natija uchun to'g'ri dastlabki qadam bo'ladi.

Algoritmni ishlab chiqish - dasturlash maqsadiga erishtiruvchi amallar ketma-ketligini to'g'ri va eng qisqa yo'l orqali shakllantirish bosqichidir.

Kodlash - aniq talablar asosida shakllantirilgan algoritmni mashinaga o'qitish uchun biror dasturlash muhitidagi tili orqali yoziladi.

Dasturni ishga tushirish - bunda ishlab chiqilgan dasturni ishga tushirib xatoliklardan xoli qilish jarayonidir. Xatoliklarni ikki turi mavjud: sintaktik (dastur matnida uchraydigan) va algoritmik. Sintaktik xatoni aniqlash bir muncha engil. Chunki, aksariyat hollarda uni dasturlash tili o'zi aniqlaydi. Algoritmik xatoni aniqlash murakkab bo'lib, katta loyihali

dasturlarda yanada murakkablashib ketadi. Sababi bunda dastur xatosizdek ishlaydi.

Testdan o'tkazish — ishlab chiqilgan dasturiy mahsulotni foydalanuvchilar ommasiga qulay shaklda taqdim etish. Bu muhim bosqich bo'lib, dasturning interfeysidan tortib, kiritiladigan o'zgaruvchilar va boshqa tomonlarini aniq hamda tushunarli bo'lishi zarur. Bu jarayonda olingan natijalar orqali dasturning to'g'ri yoki noto'g'ri ishlayotganligi aniqlanadi.

Algoritm qaysi ijrochiga mo'ljallab tuzilayotganligiga qarab quyidagi shakllardan birida ifodalanishi mumkin:

1. Matn shakli. 2. Blok-sxema shakli. 3. Dastur shakli.

Algoritmnlarni ifodalashning dastlabki ikkita shakli tabiiy tillar: matematik formulalar va simvollar vositasida ifodalanadi. Uchinchi shakli esa sun'iy tillardan birida ifodalanadi.

1.2.1 Algoritmning matn shaklida ifodalanishi

Algoritmni ifodalashning eng ko'p tarqalgan shakli bu matn shakli, ya'ni uni oddiy tilda so'zlar bilan bayon qilish hisoblanadi.

So'zlar yordamida tuzilgan algoritmning ijrochisi inson hisoblanadi. Algoritmning matn shaklida har bir ko'rsatma, jumlar orqali buyruk mazmunida beriladi. Algoritmning bunday shaklidan, odatda, kundalik hayotda uchraydigan masalalarni hal qilish algoritmlarini tuzishda foydalaniladi. Masalan, inson hayotida har kuni bajaradigan yumushlari, ya'ni har xil taomlar tayyorlash, choy damlash va hokazo algoritmlar so'zlar yordamida ifodalanadi.

Umuman olganda, ixtiyoriy turdagi masalalarning algoritmlarini so'zlar yordamida tuzish mumkin.

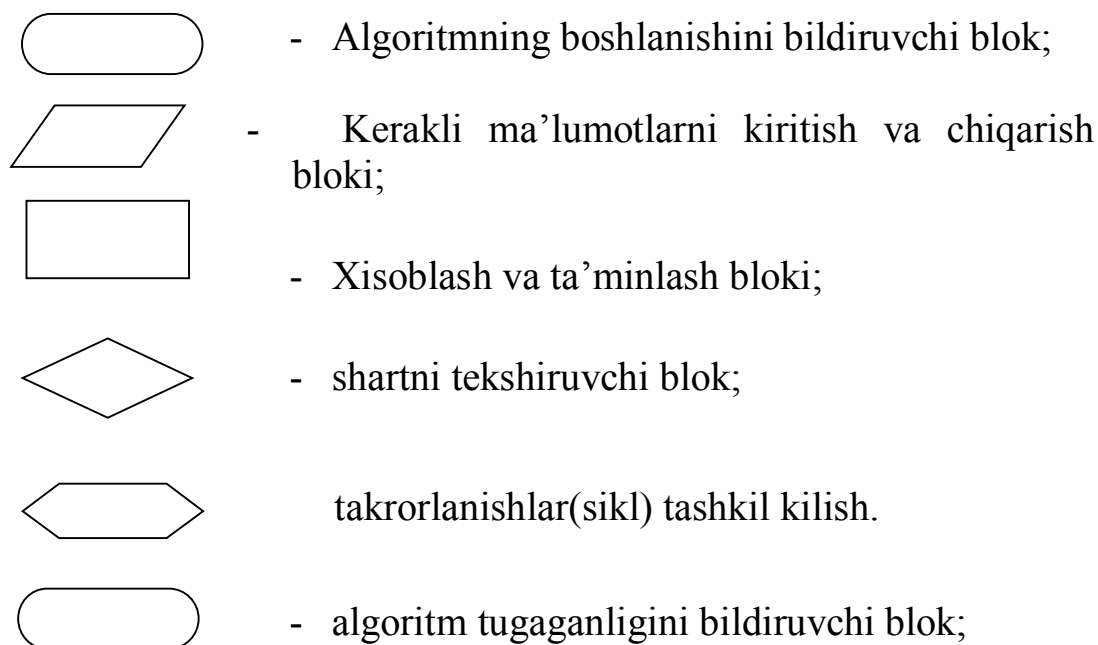
Masalan, aralash sonni noto'g'ri kasrga aylantirish algoritmi quyidagicha oddiy tilda tuziladi:

1. Sonning butun qismini mahrajiga ko'paytiramiz va uni **KI** bilan belgilaymiz.
2. **KI** ga kasrning suratida turgan son qo'shiladi va natija suratga yoziladi.
3. Kasrning maxraji o'zgarishsiz koladi.

1.2.2 Algoritmning blok-sxema shaklida ifodalanishi

Matematika masalalarini hisoblashda algoritmni blok-sxema ko‘rinishida tasvirlash uni tushunish uchun qulaylik tug‘diradi. Blok-sxema ko‘rsatkichlar bilan bog‘lanuvchi har xil bloklar ketma-ketligidan tashkil topadi. Algoritmning bajarilish tartibi ko‘rsatkich bilan ko‘rsatiladi.

Blok-sxemalarda algoritmning alohida bosqichlarini belgilash uchun quyidagi belgilardan foydalaniladi:



Har kandy algoritm blok-sxema shaklida ifodalanganda har doim uning boshlanishini bildiruvchi blok bilan boshlanib, algoritmning tugaganligini bildiruvchi blok bilan yakunlanadi.

Kerakli ma’lumotlarni kiritish va chiqarish blokida dastlabki berilganlar kiritiladi va olingan natijalar chiqariladi.

SHartni tekshirish bloki ichiga shart yoziladi, qo‘yilgan shartning bajarilish-bajarilmasligiga qarab strelka ikki tomonga yo‘naladi. Strelkalarining biriga “ha” suzi, ikkinchisiga “yo‘k,” so‘zi yoziladi. “Ha” so‘zi shartning bajarilganligini, “yo‘q” so‘zi esa shartning bajarilmaganligini bildiradi. Shartning bajarilgan yoki bajarilmaganligiga qarab, algoritmning bajarilish tartibi aniqlanadi.

Hisoblash va ta’minlash blokida hisoblanishi kerak bo‘lgan ifoda yoziladi. Ifodaning qiymati hisoblanib, tenglik belgisining chap tomonidagi o‘zgaruvchiga ta’minlanadi.

Mavzu CHiziqli, tarmoqlanuvchi va takrorlanuvchi tarkibga ega bulgan algoritmlar tuzish

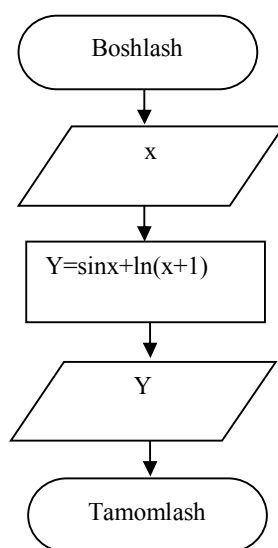
Algoritmni tasvirlashning uch turi mavjud: chiziqli algoritim, tarmoqlanuvchi algoritim, takrorlanuvchi algoritim.

3.1 Chiziqli jarayonlarga doir algoritmlar

Chiziqli tarkibli algoritmlarda algoritmlarning xar bir punkti ketma-ket bajariladi.

Masalan: $y=\sin x+\ln(x+1)$ funksiyaning qiymatini hisoblash algoritmi tuzilsin, bu erda $x=3,9$.

Blok sxema:Chiziqli jarayonga doir algoritm



Rasm-1 Chiziqli algoritim

Yuqoridagi algoritmning bajarilishi ketma-ketligi quyidagicha

1-qadam boshlash

2-qadam x o`zgaruvchiga ma`lumot kiritish

3-qadam $Y=\sin x+\ln(x+1)$ funksiyaning qiymatini topish

4-qadam **Y** funksiyaning qiymatini chop qilish.

Tarmoqlanuvchi jayonlarga doir algoritmlar

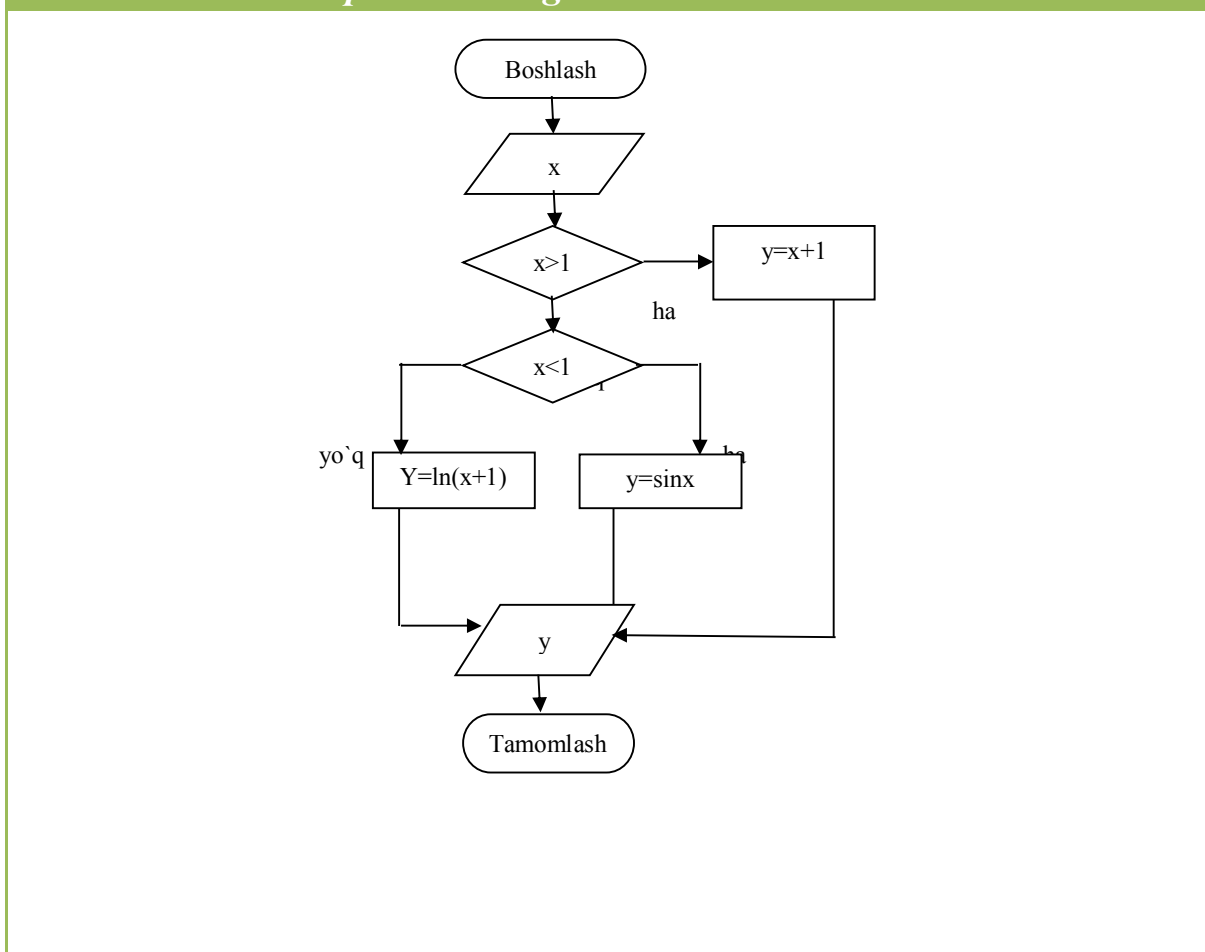
Tarmoqlanuvchi tarkibli algoritmlarda ham har bir band bir martadan bajariladi. Lekin shartning bajarilishi yoki bajarilmasligiga qarab qaysi bandlar ketma-ket bajarilishi aniqlanadi.

$$Y = \begin{cases} x + 1 & x > 1 \quad \text{bўlsa} \\ \ln(x + 1) & x = 1 \quad \text{bўlsa} \\ \sin x & x < 1 \quad \text{bўlsa} \end{cases}$$

$x = 4,5$

Masalan: Argument x ning ixtiyoriy qiymatida yuqoridagi y funksiyaning qiymatini hisoblash algoritmi tuzilsin.

Blok sxema: Tarmoqlanuvchi algoritim



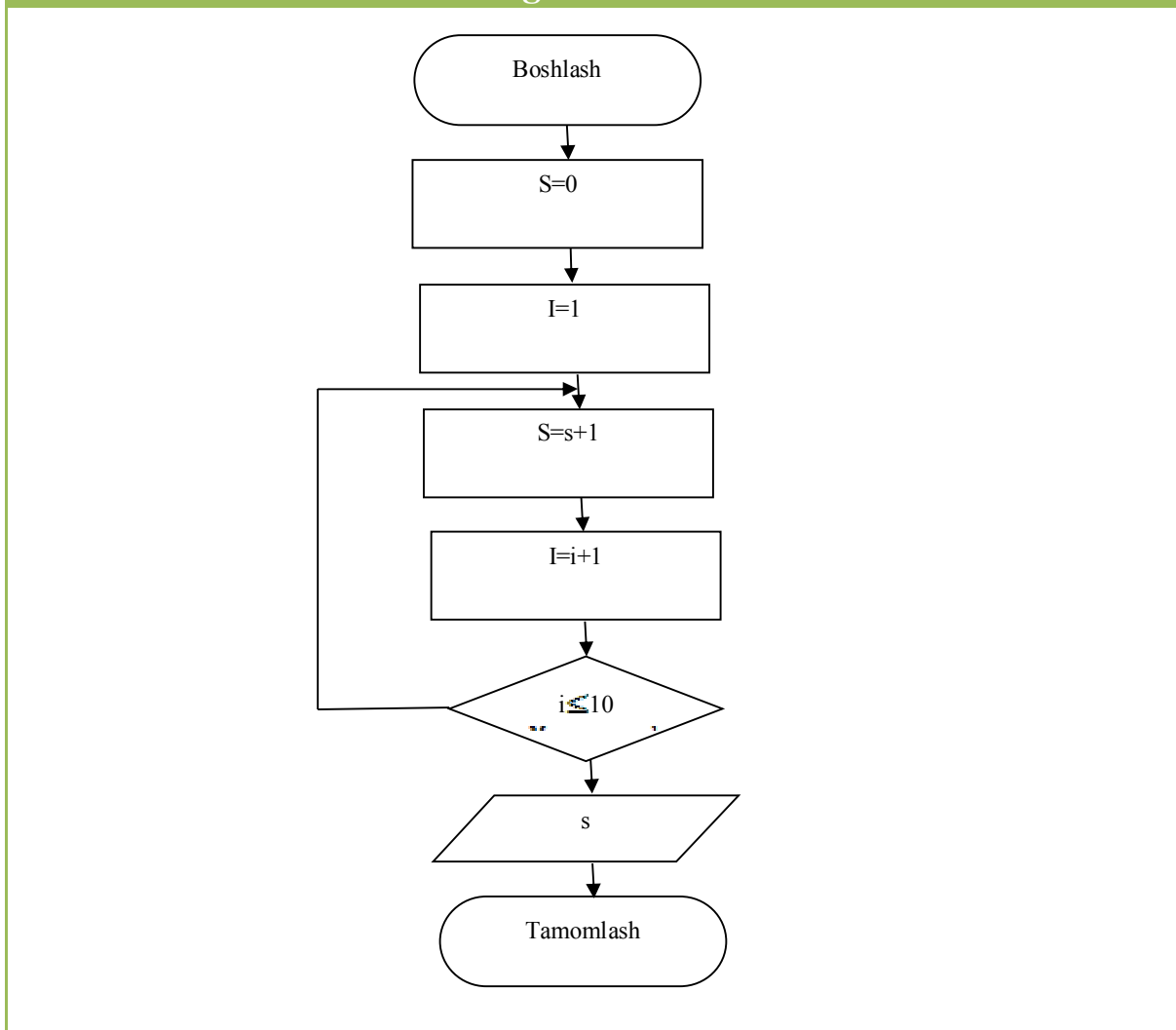
Rasm-2 Tarmoqlanuvchi algoritim

1.3.3. Takrorlanuvchi jayonlarga doir algoritmlar

Takrorlanuvchi algoritmlarda algoritmning bir bo‘lak bandlari parametrlarning qabul qilish qiymatiga qarab ketma-ket bir necha marta bajariladi.

Masalan: 1 dan 10 gacha bo‘lgan musbat butun sonlarning yig‘indisini topish algoritmi tuzilsin.

Blok sxema: Takrorlanuvchi algoritim



Rasm-3 Takrorlanuvchi jarayonlarga doir algoritim

C++ DASTURLASH TILI

C++ 1980 yillar boshida Bjarne Stroustrup tomonidan C ga asoslangan tarzda tuzildi. C++ juda ko'p imkoniyatlarni o'z ichiga olgan, lekin eng asosiysi u obyektlar asosida dasturlashga imkon beradi. Dasturlarni tez va sifatli yozishga hozirgi kunda katta ahamiyat berilmoqda . Buni ta'minlash uchun obyektli dasturlash g'oyasi ilgari surildi.

C++ Alfavit. C++ alfavitiga quyidagi simvollar kiradi.

- Katta va kichik lotin alfaviti xarflari (A,B,..,Z,a,b,...,z)
- Raqamlar: 0,1,2,3,4,5,6,7,8,9
- Maxsus simvollar: “ , { } | [] () + - / % \ ; ‘ . : ? < = > _ ! & * # ~
- Ko'rinmaydigan simvollar (“umumlashgan bushliq simvollari”).

Leksemalarni uzaro ajratish uchun ishlatiladigan simvollar (misol uchun bushlik, tabulyatsiya, yangi qatorga o'tish belgilari).

Izohlarda, satrlarda va simvolli konstantalarda boshqa literalalar, masalan rus xarflarini ishlatilishi mumkin.

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

Identifikator. Identifikatorlar lotin xarflari, ostki chiziq belgisi va sonlar ketma ketligidan iborat buladi. Identifikator lotin xarfidan yoki ostki chiziq belgisidan boshlanishi lozim.

Misol uchun:

A1, _MAX, adress_01, RIM, rim

Katta va kichik xarflar farklanadi, shuning uchun ohirgi ikki identifikator bir biridan farq qiladi.

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

Xizmatchi so'zlar. Tilda ishlatiluvchi ya'ni dasturchi tomonidan uzgaruvchilar nomlari sifatida ishlatish mumkin bulmagan identifikatorlar xizmatchi so'zlar deyiladi.

C ++ tilida quyidagi xizmatchi so'zlar mavjud:

int	else	long	unsigned
char	for	return	continue
float	do	case	auto
double	while	sizeof	if

extern	struct	default	point
register	goto	short	include
typedef	switch	Break	
static	union	entry	

O'zgaruvchilar ob'ekt sifatida. C++ tilining asosiy tushunchalaridan biri nomlangan hotira qismi – ob'ekt tushunchasidir. Ob'ektning xususiy holi bu o'zgaruvchidir. O'zgaruvchiga qiymat berilganda unga ajratilgan hotira qismiga shu qiymat kodi yoziladi. O'zgaruvchi qiymatiga nomi orqali murojat qilish mumkin, hotira qismiga esa faqat adresi orqali murojaat qilinadi. O'zgaruvchi nomi bu erkin kiritiladigan identifikatoridir. O'zgaruvchi nomi sifatida xizmatchi so'zlarni ishlatish mumkin emas.

Quyidagi jadvalda konstantalar chegaralari va mos tiplari ko'rsatilgan:

Ma'lumotlar turi	Hajm, bit	Qiymatlar chegarasi	Tip vazifasi
Unsigned char	8	0...255	Kichik butun sonlar va simvollar kodlari
Char	8	-128...127	Kichik butun sonlar va ASII kodlar
Enum	16	-32768...32767	Butun sonlar tartiblangan katori
Unsigned int	16	0...65535	Katta butun sonlar
Short int	16	-32768...32767	Kichik butun sonlar, tsikllarni boshqarish
Int	16	-32768...32767	Kichik butun sonlar, tsikllarni boshqarish
Unsigned long	32	0...4294967295	Astronomik masofalar
Long	32	-147483648... ...2147483647	Katta sonlar
Float	32	3.4E-32...3.4E+38	Ilmiy hisoblar (7 raqam haqiqiy sonlar)
Double	64	1.7E-308...1.7E+308	Ilmiy hisoblar(15 raqam haqiqiy sonlar)
Long double	80	3.4E-4932... 1.1E+4932	Moliyaviy hisoblar (19 raqam haqiqiy sonlar)

C++ tilida ma'lumotlarni bir necha xilda kiritish mumkin:

- Ma'lumotlarni konstantalar yordamida ifodalash. Bunda dasturda ishlatiladigan hamma ma'lumotlar **const** orqali ifodalangan bo'ladi;
- *O'zlashtirish operatori yordamida ifodalash.* Bunda dasturda ma'lumotlarga to'g'ridan – to'g'ri qiymatlarni o'zlashtirish yo'li bilan amalga oshiriladi.
- *Ma'lumotlarni foydalanuvchi yordamida kiritish.* Bunda dasturda o'zgaruvchilarga **cin** operatori yordamida ma'lumotlar kiritiladi.

C++ tilida ma'lumotlarni kiritishni tashkil etishda **cin** operatoridan foydalaniladi .

CIN>>argument1>> argument2 >> argument3;

C++ tilida ma'lumotlarni ekranga chiqarishni tashkil etishda **COU**T operatoridan foydalaniladi .

COUT<<argument1<< argument2 << argument3;

Misol.

```
# include <iostream.h >
int main()
{
//ekranga chiqarish
cout<< "salom dunyo!!! ";
```

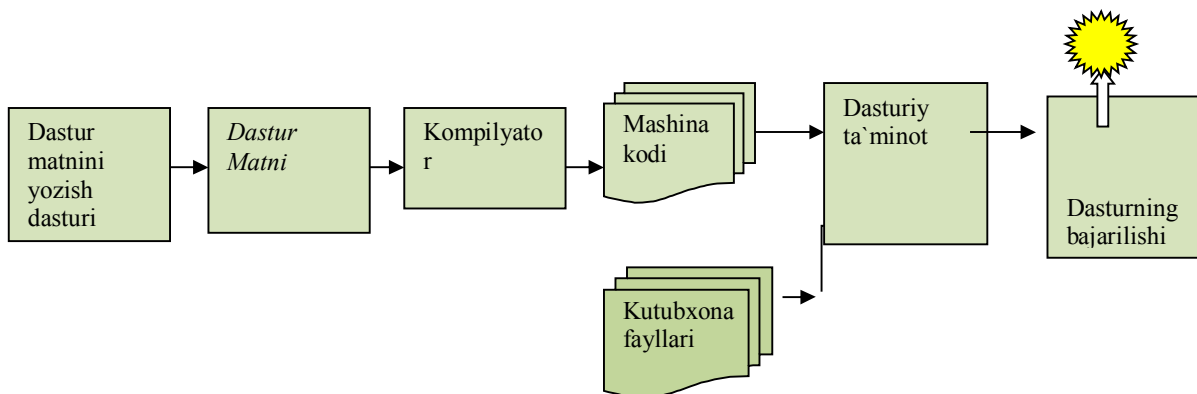
Return 0;

}

Natija salom dunyo!!! Ma'lumoti ekranga chiqadi.

Ma'lumotlarni ekranga chiqarishni tashkil etishda ishlatiladigan maxsus belgilar:

"\n"	Yangi satrga o'tkazadi;
"\t"	Tabulyatsiya bo'sh joy qoldiradi;
"\b"	Bir qadam chapdan ekranga siqiradi;
"\""	Ikkitalik qavscha
"\?"	Bittali qavscha
"\?"	So'roq belgisi
"\""	Teskari slesh
"\r"	Karetkani qaytarvdi yani chiqarmaydi
"\a"	tovush signalini chiqaradi.



Rasm-4 C++ tilida dasturning bajarilish jarayoni

C++ tilida matematik ifodalarni yozishda **math.h** kutubhonasidan foydalaniladi va uni dastur boshida `#include <math.h>` ko'rinishida yo'ziladi.

Ayrim matematik funksiyalarni C++tilida yozish

No	Funksiyani C++ da yozish	Matematik aniqlanishi	Masalan
1.	sin(x)	x ning trigonometrik sinusi x radianlarda sinx	sin(0.0)=0.0
2.	cos(x)	x ning trigonometrik kosinusi x radianlarda cosx	cos(0.0)=1.0
3.	tan(x)	x ning trigonometrik tangensi x radianlarda tgx	tan(0.0)=0
4.	exp(x)	e ning x-darajasi e^x (eksponential funksiya)	exp(1.0)=2.71828 exp(2.0)=7.38906
5.	log(x)	x ning natural logorifmi lnx (e asosga ko'ra logorifm)	log(2.718282)=1.0
6.	log10(x)	x ning 10 asosga ko'ra logorifmi lgx	log10(1000.0)=3.0
7.	sqrt(x)	x ning kvadrat ildizi \sqrt{x}	sqrt(625.0)=25.0
8.	pow(x,y)	x ning y darajasini beradi x^y	pow(3,4)=81.0 pow(16,0.25)=2
9.	ceil(x)	x ni x dan katta yoki unga teng bo'lgan eng kichik butun songacha yaxlitlaydi	ceil(12.6)=13.0 ceil(-2.4)=-2.0
10.	fabs(x)	x ning modul qiymati $ x $	x>0 fabs(x)=+x x<0 fabs(x)=+x x=0 fabs(x)=0

11.	floor(x,y)	x ni x dan kichik bo'lgan eng katta butun songacha yaxlitlaydi	floor(-15,9)=-16.0 floor(4,8)=4.0
12.	fmod(x,y)	x/y ning qoldig'ini kasr son tipida beradi	fmod(7.3,1.7)=0.5
13.	acos(x)	arkkosinus arccos(x)	funksiya acos(0)=1.57 acos(1)=0
14.	asin(x)	arksinus arcsin(x)	funksiya asin(0)= asin(1)=
15.	atan(x)	arkktangens arctg(x)	funksiya atan(1)=0.78 atan(0)=0
16.	atan2(x,y)	arkktangens arctg(x/y)	funksiya atan2(1,2)=

C++ DA ARIFMETIK AMALLAR

Ko'p programmalar ijro davomida arifmetik amallarni bajaradi. C++ dagi amallar quyidagi jadvalda berilgan. Ular ikkita operand bilan ishlatildi.

C++ dagi amal	Arifmetik operator	Algebraik ifoda	C++ dagi ifodasi:
Qo'shish	+	$h+19$	$h+19$
Ayirish	-	$f-u$	$f-u$
Ko'paytirish	*	$s1$	$s*1$
Bo'lish	/	v/d	v/d
Modul olish	%	$k \text{ mod } 4$	$k\%4$

Bularning ba'zi birlarinig hususiyatlarini ko'rib chiqaylik. Butun sonli bo'lishda, yani bo'luvchi ham, bo'linuvchi ham butun son bo'lganda, javob butun son bo'ladi. Javob yahlitlanmaydi, kasr qismi tashlanib yuborilib, butun qismining o'zi qoladi. Modul operatori (%) butun songa bo'lishdan kelib chiqadigan qoldiqni beradi. $x\%y$ ifodasi x ni y ga bo'lgandan keyin chiqadigan qoldiqni beradi. Demak, $7\%4$ bizga 3 javobini beradi. % operatori faqat butun sonlar bilan ishlaydi. Vergulli (real) sonlar bilan ishlash uchun "**math.h**" kutubhonasidagi **fmod** funksiyasini qo'llash kerak. C++ da qavslarning ma'nisi huddi algebradagidekdir. Undan tashqari boshqa boshqa algebraik ifodalarning ketma-ketligi ham odatdagidek. Oldin ko'paytirish, bo'lish va modul olish operatorlari ijro ko'radi. Agar bir necha operator ketma-ket kelsa, ular chapdan o'nga qarab ishlanadi.

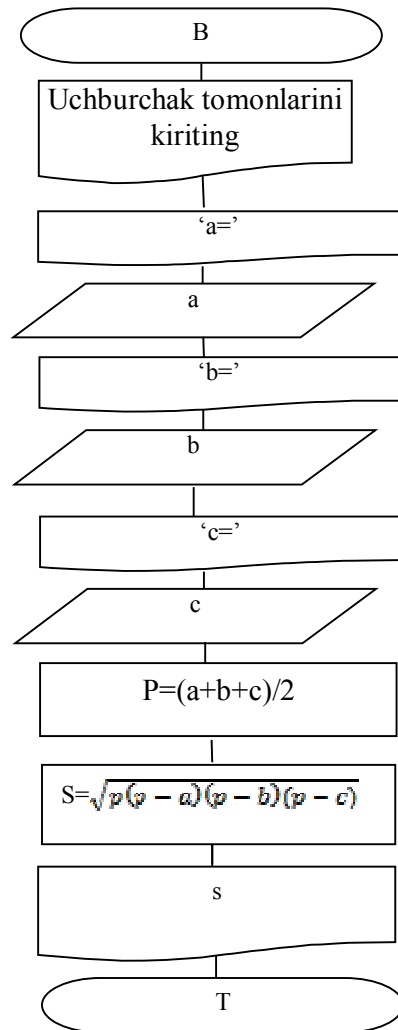
Misol keltiraylik. $k = m * 5 + 7 \% n / (9 + x);$

Birinchi bo'lib $m * 5$ hisoblanadi. Keyin $7 \% n$ topiladi va qoldiq $(9 + x)$ ga bo'linadi. Chiqqan javob esa $m * 5$ ning javobiga qo'shiladi.

CHIZIQLI JARAYONLARNI DASTURLASH

MASALA: Uchburchak tomonlari a, b, c lar berilgan. Geron formulasi yordamida ushbu uchburchak yuzasini hisoblashning algoritmini blok-sxema ko'rinishida, dasturini C++ tilida tuzing.

Masalani echish algoritmining blok-sxema ko'rinishi.



Masalani yechishning C++ da dastur ko'rinishi.

```
#include <iostream>;
#include <math.h>;
using namespace std;
int main()
{ float p,s,a,b,c;
  cout<<"a"<<endl;
  cin>>a;
  cout<<"b"<<endl;
  cin>>b;
  cout<<"c"<<endl;
  cin>>c;
  p=((a+b+c)/2);
  s=(sqrt(p*(p-a)*(p-c)*(p-b)));
  cout <<"s="<<s<< endl;
  return 0;
}
a=3
b=3
c=3
s=3.89711
```

Nazorat savollari

1. C++ dasturlash tilining umimiy tuzilishi?
2. C++ tilida amallar necha guruhga bo'linadi ?
3. C++ tilida qanday turdagi o'zgaruvchilar foydalaniladi?

TARMOQLANUVCHI ALGORITMLARNI DASTURLASH USULLARI

C++ tilida nishon deb orqasidan ikki nuqta (:) yoziladigan identifikatorga aytiladi. Nishon doimo boshqaruv o'tishi lozim bo'lgan operatoridan oldin o'rnatiladi. Kerakli nishonga o'tish uchun goto operatori qo'llaniladi. Bunda kalit so'zdan keyin nishon nomi yoziladi.

goto operatoriga misol keltiramiz.

```
1: include <iostream>;
2: int main()
3: {
4: int a, b,c;
5: a=10;
6: b=20;
7: goto belgi;
8: b=30;
9: c=a+b;
10: belgi:
11: c=a*b;
12: sout <<"c= " <<c<< "\n";
13: cout <<"c-b= " <<c-b<< "\n";
14: return 0;
15: }
```

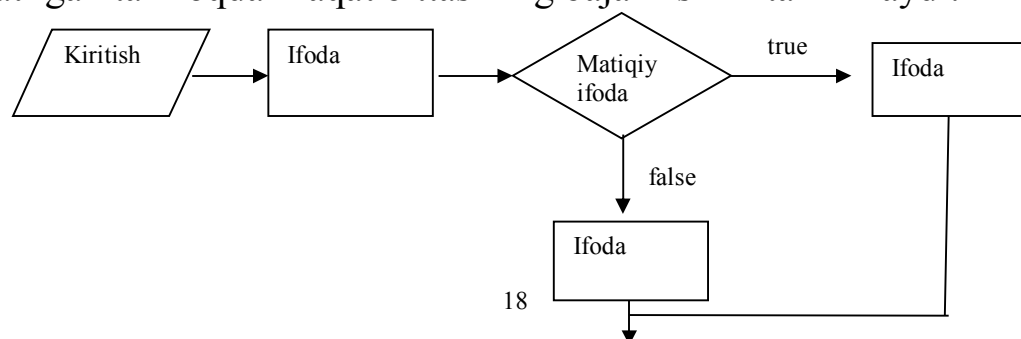
NATIJA:

s=200

s-b=180

Agar hisoblash jarayoni biror bir berilgan shartning bajarilishiga qarab turli tarmoqlar bo'yicha davom ettirilsa va hisoblash jarayonida har bir tarmoq faqat bir marta bajarilsa, bunday hisoblash jarayonlariga tarmoqlanuvchi algoritmlar deyiladi.

Tarmoqlanuvchi algoritmlar uchun ayri strukturasi ishlatiladi. Tarmoqlanuvchi strukturasi berilgan shartning bajarilishiga qarab ko'rsatilgan tarmoqdan faqat bittasining bajarilishini ta'minlaydi.



BOSHQARUV IFODALARI

Bu bo'limda biz strukturali dasturlashning asosiy prinsip va qismlarini ko'rib chiqamiz. Ma'lum bir dasturni yozish uchun belgilangan qadamlarni bosib o'tish kerak. Masala aniqlangandan so'ng uni yechish uchun mo'ljallangan algoritm tuziladi. Keyin esa psevdokod yoziladi. Psevdokod algoritmda bajariladigan qadamlarni ko'rsatadi. Bunda faqat bajariladigan ifodalar ko'rib chiqiladi. Psevdokodda o'zgaruvchi e'lonlari yoki boshqa ma'lum bir dasturlash tiliga mansub bo'lgan yordamchi amallar bo'lmaydi. Psevdo kodni yozish dasturlashni ancha osonlashtiradi, algoritm mantiq'ini tushunishga va uni rivojlanritishga katta yordam beradi. Misol uchun bir dasturning rejasi va psevdo kodi 3-4 oy yozilgan bo'lsa va yuqori darajada detallashtirilgan bo'lsa, ushbu dasturning C++ yoki boshqa tildagi kodini yozish 2-3 hafta vaqt oladi halos. Bu yozilgan programmada hato ancha kam bo'ladi, uni keyinchalik takomillashtirish arzoniga tushadi. Hozirgi paytda dastur o'zgarishi favqulotda hodisa emas, balki zamon talabidir.

DASTUR IJRO STRUKTURALARI

Asosan dasturdagi ifodalar ketma-ket, navbatiga ko'ra ijro etiladi. Gohida bir shart bajarilishiga ko'ra, ijro boshqa bir ifodaga o'tadi. Navbatdagi emas, dasturning boshqa yerida joylashgan ifoda bajariladi. Yani sakrash yoki ijro ko'chishi vujudga keladi. 60-chi yillarga kelib, dasturlardagi ko'pchilik hatolar aynan shu ijro ko'chishlarining rejasiz ishlatilishidan kelib chiqishi ma'lum bo'ldi. Bunda eng katta aybdor deb bu ko'shishlarni amalga oshiruvchi goto(..ga bor)ifodasi belgilandi. **goto** dastur ijrosini deyarli istalgan yerga ko'chirib yuborishi mumkin. Bu esa programmani o'qishni va uning strukturasi murakkablashtirib yuboradi. Shu sababli "strukturali dasturlash" atamasi "goto ni yo'q qilish" bilan tenglashtirilardi. Shuni aytib o'tish kerakki, goto kabi shartsiz sakrash amallarini bajaruvchi ifodalar boshqa dasturlash tillarida ham bor. Tadqiqotlar shuni ko'rsatdiki, istalgan programma goto siz yozilishi mumkin ekan. goto siz yozish uslubi strukturali dasturlash deb nom oldi. Va bunday dastur yozish metodi katta iqtisodiy samara beradi. Strukturali dasturlash asosi shundan iboratki, har bir programma faqatgina uch hil boshqaruv strukturalaridan iboratdir. Bular ifodalarni ketma-ket ijro etish strukturasi (sequence structure), tanlash strukturasi (selection structure) va amalni qayta ijro etish strukturasi (repetition structure).

Ifodalarni ketma-ket ijro etish strukturasi C++ tomonidan ta'minlanadi. Normal sharoitda C++ ifodalari dasturdagi navbatiga ko'ra bajariladi.

Tanlash buyruqlari uchta. Bular **if, if/else** va **switch** dir. Qayta ijro etish buyruqlari gurugiga ham uchta a'zo bor, bular **while, do/while** va **for**. Bularni har birini keyinroq tahlil qilib chiqamiz. Yuqoridagi buyruqlar nomlari C++ dasturlash tilining mahsus so'zlaridir. Dasturchi bu so'zlarni o'zgaruvchi yoki funksiyalar nomi sifatida qo'llashi ta'qiqlanadi. Quyida C++ ning ajratilgan so'zlarining to'liq ro'yhati berilgan.

C++ dagi yetita boshqaruv strukturasi aytilib o'tdik. Ular bittagina boshlanish nuqtasiga va bittagina chiqish nuqtasiga egadirlar. Demak biz bu dastur bo'laklarini ketma-ket ulab ketishimiz mumkin. Boshqaruv trukturalari-

ning bu kabi ulanishini devorning g'ishtlarini ustma-ust qalashga ham taqqoslasak bo'ladi. Yoki biz bu bloklarni bir-birining ichiga joylashtirishimiz mumkin. Bu kabi qo'llashish ikkinchi uslub bo'ladi. Mana shu ikki yo'l bilan bog'langan yetita blok yordamida biz istalgan dasturimizni yoza olamiz.

if STRUKTURASI

Biz shartga ko'ra bir necha harakat yo'lidan bittasini tanlaymiz. Misol uchun: agar bolaning yoshi 7 ga teng yoki katta bo'lsa u maktabga borishi mumkin bo'lsin. Buni C++ da if ni qo'llab yozaylik.

```
if (yosh >= 7)
    maktab();
```

Bu yerda shart bajarilishi yoki bajarilmasligi mumkin. Agar yosh o'zgaruvchisi 7 ga teng yoki undan katta bo'lsa shart bajariladi va maktab() funksiyasi chaqiriladi. Bu holat true (to'g'ri) deyiladi. Agar yosh 7 dan kichik bo'lsa, maktab() tashlab o'tiladi. Yani false (noto'g'ri) holat yuzaga keladi. Biz shart qismini mantiqiy operator-larga asoslanganligini ko'rib chiqqan edik. Aslida esa shartdagi ifodaning ko'rinishi muhim emas – agar ifodani nolga keltirish mumkin bo'lsa false bo'ladi, noldan farqli javob bo'lsa, musbatmi, manfiymi, true holat paydo bo'ladi va shart bajariladi. Bunga qo'shimcha qilib o'tish kerakki, C++ da mahsus bool tipi mavjud. Bu tipdagi o'zgaruvchilarning yordamida bul (mantiqiy) arifmetikasini amalga oshirish mumkin. bool o'zgaruvchilar faqat true yoki false qiymatlarini olishlari mumkin.

if/else STRUKTURASI

if ni qo'llaganimizda ifoda faqat shart haqiqat bo'lgandagina bajariladi, aks holda tashlanib o'tiladi. if/else yordamida esa shart bajarilmaganda (false natija chiqqanda) else orqali boshqa bir yo'ldan borishni belgilash

mumkin. Misolimizni takomillashtirsak. Bola 7 yosh yoki undan katta bo'lsa maktabga, 7 dan kichkina bo'lsa bog'chaga borsin.

```
if (yosh >= 7)
    maktab(); //nuqta-vergul majburiydir
else
    bogcha();
```

Yuqorida if ga tegishli bo'lgan blok bitta ifodadan (maktab()) iborat. Shu sababli nuqta-vergul qo'yilishi shart. Buni aytib o'tishimizning sababi, masal C++da hech narsa qo'yilmasligi shart. C++ da bitta ifosa turgan joyga ifodalar guruhini {} qavslarda olingan holda qo'ysa

bo'ladi. Masalan:

```
if (yosh >= 7){
    cout << "Maktabga!\n";
    maktab();
}
else{
    cout << "Bog'chaga!\n";
    bogcha();
}
```

Aslida har doim {} qavslarni qo'yish yahshi odat hisoblanadi, keyinchalik bir ifoda turgan joyga qo'shimcha qilinganda qavslardan biri unutilib qolmaydi.

Strukturali dasturlashning yana bir harakterli joyi shundaki tabulyatsiya, bo'sh joy va yangi satrlar ko'p qo'llaniladi. Bu programmani o'qishni osonlashtirish uchun qilinadi. C++ uchun bo'sh joyning hech ahamiyati yo'q, lekin dasturni tahrir qilayotgan odamga buyruqlar guruhini, bloklarni tabulyatsiya yordamida ajratib bersak, unga katta yordam bo'ladi. Yuqoridagini quyidagicha ham yozish mumkin:

```
if(yosh>=7){cout<<"Maktabga!\n";maktab()}else{cout<<"Bog'chaga!\n";
bogcha()};
```

Biroq buni o'qish ancha murakkab ishdir.

C++ da if/else strukturasiga o'hshash ?: shart operatori (conditional operator) ham bordir. Bu C++ ning bittagina uchta argument oluvchi operatori. Uch operand va shart operatori shart ifodasini beradi. Birinchi operand orqali shartimizni beramiz. Ikkinchi argument shart true (haqiqat) bo'lib chiqqandagi butun shart ifodasining javob qiymatidir. Uchinchi operand shartimiz bajarilmay (false) qolgandagi butun shart ifodasining qiymatidir. Masalan:

```
bool bayroq;
```

```
int yosh = 10;
bayroq = ( yosh >= 7 ? true : false );
```

Agar yosh 7 ga teng yoki katta bo'lsa, bool tipidagi o'zgaruvchimiz true qiymatini oladi, aks taqdirda false bo'ladi. Shart operatori qavslar ichida bo'lishi zarur, chunki uning kuchi katta emas. Javob qiymatlar bajariladigan funksiyalar ham bo'lishi mumkin:

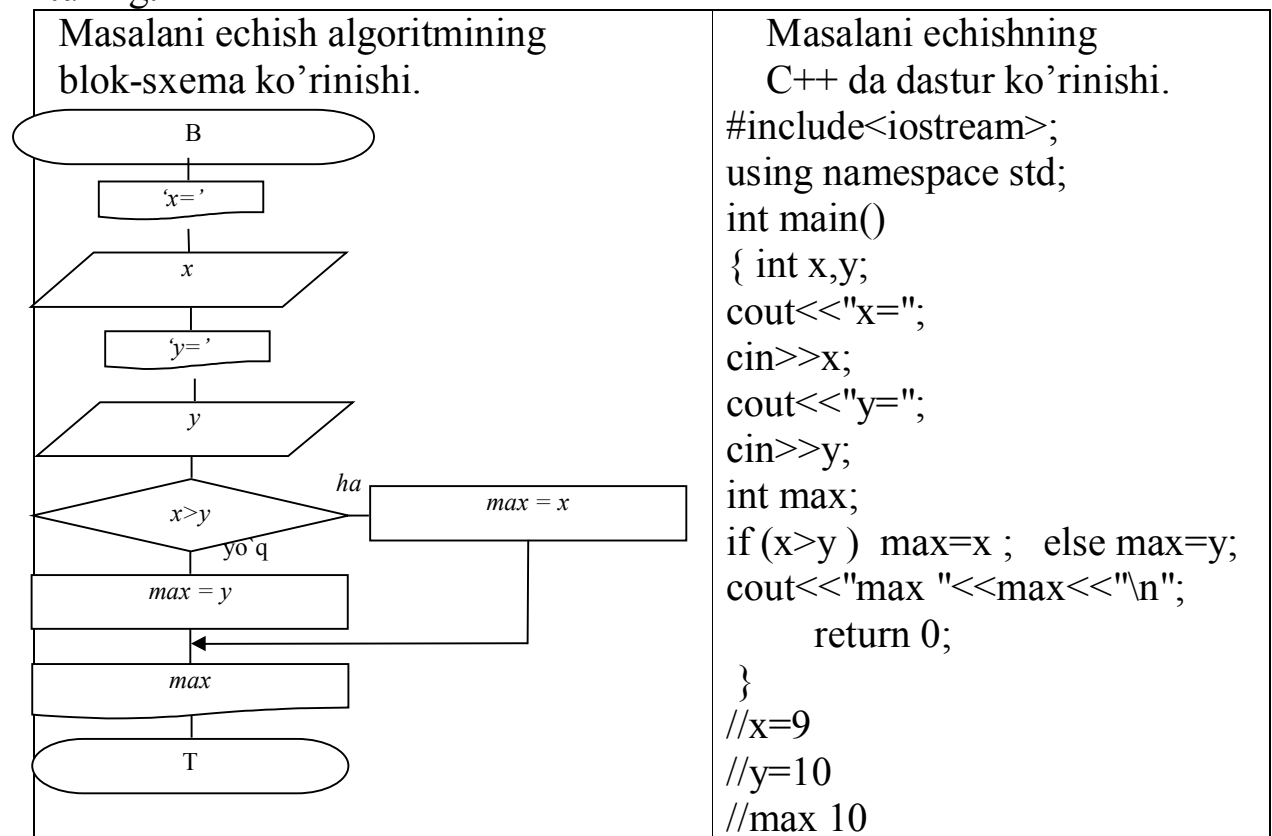
```
yosh >= 7 ? maktab() : bogcha();
```

if/else strukturalarini bir-birining ichida yozishimiz mumkin. Bunda ular bir-biriga ulanib ketadi. Misol uchun tezlikning kattaligiga qarab jarimani belgilab beruvchi blokni yozaylik.

```
if (tezlik > 120)
    cout << "Jarima 100 so'm";
else if (tezlik > 100)
    cout << "Jarima 70 so'm";
else if (tezlik > 85)
    cout << "Jarima 30 so'm";
else
    cout << "Tezlik normada";
```

Agar tezlik 120 dan katta bo'lsa birinchi if/else strukturasining haqiqat sharti bajariladi. Va bu holda albatta tezlik o'zgaruvchimizning qiymati ikkinchi va uchinchi if/else imizni ham qoniqtiradi. Lekin solishtirish ulargacha bormaydi, chunki ular birinchi if/else ning else qismida, yani noto'g'ri javob qismida joylashgandir. Solishtirish birinchi if/else da tugashi (aynan shu misolda)tanlash amalini tezlashtiradi. Yani bir-biriga bog'liq if/else lar alohida if strukturalari blokidan tezroq bajarilishi mumkin, chunki birinchi holda if/else blokidan vaqtliroq chiqish imkoni bor. Shu sababli ich-ichiga kirgan if/else lar guruhida true bo'lish imkoni ko'proq bo'lgan shartlarni oldinroq tekshirish kerak.

MASALA: Ikkita x va y sonlari berilgan bu sonlarning **MAX** ni hisoblashning algoritmini blok-sxema ko'rinishida, dasturini C++ tilida tuzing.



SWITCH strukturasi

Biz `if` va `if/else` operatorlari bilan tanishib oldik. Lekin ayrim masalalarni echishda `if` operatori ichida ko'p sondagi `if` operatorlarini qo'llashga to'g'ri keladi. Bu esa dasturni yozishni ham, uni tushinishni ham murakkablashtirib yuboradi. Bunday muammoni echish uchun C++tilida `switch` operatori qo'llaniladi. Bu operatorning `if` operatoridan asosiy farqi shuki, unda bir yo'la bir nechta shart tekshiriladi. Natijada dasturni tarmoqlanishi nisbatan samaraliroq bo'ladi. `switch` operatorining sintaksisi quyidagichadir.

```

switch(ifoda)
{
case Birinchi kiymat: operator;break;
case Ikkinchi kiymat: operator;break;
case N nchi kiymat: operator;break;
default: operator;
}
    
```

switch operatoridan keyingi qavs ichida tilning konstruksiyasi nuqtai-nazaridan to'g'ri bo'lgan ixtiyoriy ifodani ishlatish mumkin. Operator identifikatori o'rnida ham ixtiyoriy operator yoki ifoda, hamda operator va ifodalarning ketma-ketligini ishlatish mumkin. Lekin bu erda mantiqiy operatsiyalar yoki taqqoslash ifodalarini ishlatish mumkin emas.

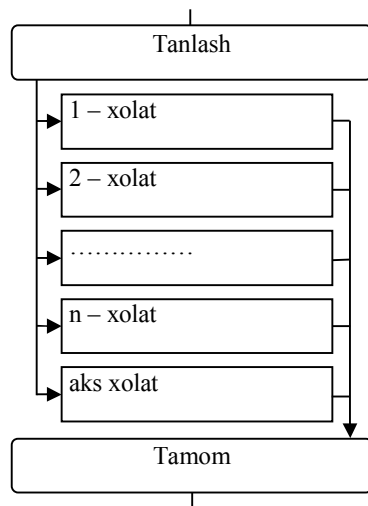
switch operatorining qo'llanilishi.

switch operatorining qo'llanilish sintaksisi quyidagicha:

```
switch(ifoda)
{
  case 1-nchi qiymat: ifoda;
  case 2-nchi qiymat: ifoda;
  ...
  case n-nchi qiymat: ifoda;
  default : ifoda;
}
```

Switch Case tanlash operatori.

Bu operator bir necha operatorlardan bittasini tanlash yuli bilan tarmoklanishni tashkil etish uchun ishlatiladi. Bu operator asosan, if operatori ichida if operatori ishlatilishining xususiy xolini ifodalaydi uni quyidagi sxema orqali tushinib olish mumkin.



switch operatori orqali dasturning tarmoqlanishi bir necha mumkin bo'lgan qiymatlarni qaytaruvchi ifodaning natijasi asosida tashkil etiladi. switch operatoridagi qavs ichida berilgan ifodaning qaytargan qiymati case operatoridan keyinda ko'rsatilgan qiymat bilan solishtiriladi. Ifodaning qiymati bilan case operatoridan keyingi qiymat mos kelsa tanlangan case operatoridan keyingi barcha satrlar bajariladi. Bunda amallarni bajarilishi break operatorigacha davom etadi.

Agarda case operatorlari qiymatidan birortasi ham qaytarilgan qiymatga mos kelmasa default operatoridan keyingi dastur satrlari bajariladi. Agarda bu operator mavjud bo`lmasa boshqaruv switch bloki tanasidan chiqadi va keyingi dastur satrlariga beriladi.

Operator yoki ifodalardan keyin break operatori qo`llanilmasa joriy case operatoridan keyingi case blokidagi barcha ifodalar bajariladi. Ko`p hollarda bunday situatsiyada xatolik ro`y beradi. Shuning uchun, break operatorini tushirib qoldirsangiz bu amalni tavsiflovchi mos izohni(komentariyni) yozishni unutmang.

1-Masala 1-7 gacha bo`lgan butun sonlar berilgan.Kiritilgan songa mos ravishda hafta kunlarini so`zda ifodolovchi programma tuzilsin.(1-Dushanba,2-Chorshanba,...h.k).

```
int kun;
#include <iostream>
using namespace std;
int main()
{
// berilgan raqamga mos hafta kunlarini ekranda namoish qilish
int kun;
// bu yerda kun xafta kunlari kun butun son
cout <<"Kunnni kiriting";
cin>>kun;
switch(kun){
case 1:cout <<"Dushanba";
break;
case 2:cout <<"Seshanba";
break;
case 3:cout <<"Chorshanba";
break;
case 4:cout <<"Payshanba";
break;
case 5:cout <<"Juma";
break;
case 6:cout <<"Shanba";break;
case 7:cout <<"Yakshanba";break;
default:cout <<"Bunday kun mavjud emas";
}
return 0;
}
```


2-MASALA Ikkita ta a va b butun sonlar ustida arifmetik amallar bajarish.

```
'+' bo'lsa a+b
 '-' bo'lsa a-b
 '*' bo'lsa a*b
 '/' bo'lsa a/b
#include <iostream>
using namespace std;
int main()
{
int a,b;
//a va b butun sonlar
char amal;
//amal +,-,*,/ belgilardan bittasi
cout <<"Arifmetik amallar"<<endl;
cout <<"a=";<<cin>>a;
cout <<"b=";<<cin>>b;
cout <<"amalni kiriting= * + - /";<<cin>>amal;
switch(amal){
case '+':cout <<a<<"+"<<b<<"="<<(a+b);break;
case '-':cout <<a<<"-"<<b<<"="<<(a-b);break;
case '*':cout <<a<<"*"<<b<<"="<<(a*b);break;
case '/':cout <<a<<"/"<<b<<"="<<(a/b);break;
default:cout<<"bunday amal yo`q";
}
return 0;
```

Nazorat savollari

1. Tarmoqlanuchi operatorlar nima uchun ishlatiladi?
2. else kalit so`zi qachon ishlatiladi?
3. if kalit so`zidan keyin qanday ifoda qo`llanilishi kerak?
4. case operatori nima uchun ishlatiladi?
5. goto nima uchun ishlatiladi?
6. Label so`zidan keyin qanday belgilar ishlatiladi?
7. Shartsiz o`tish operatorini ishlash prinsipi?
8. default nima uchun ishlatiladi?

TAKRORLANUVCHI JARAYONLARGA DASTUR TUZISH

C++ tilida takrorlanuvchi jarayonni xosil qilishning 3 ta ssuli mavjjud:

- A) «toki» – oldindan qo`yilgan shart bilan takrorlanuvchi jarayonni tashkil qilish;
- B) «gacha» – oxirida qo`yilgan shart bilan takrorlanuvchi jarayonni tashkil qilish;
- C) Parametrli tsicl yordamida takrorlanuvchi jarayonni xosil qilishni o`rganish va tsiklga doir dasturlar tuzish.

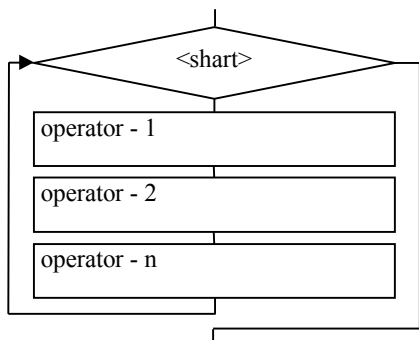
Malum bir amallar ketma-ketligining o`zgaruvchilarning turli qiymatlarida bir necha bor qayta – qayta takrorlanishi *takrorlanuvchi jarayon yoki tsicl* deyiladi. Takrorlanuvchi jarayon ichiga kiruvchi amallar *takrorlanish sohasi* deb yuritiladi. Bu jarayonda qiymatini o`zgartirib turuvchi o`zgaruvchi *tsicl parametri* deb ataladi. Takrorlanuvchi jarayonni uch xil ko`rinishda tashkil kilish mumkin.

«toki» – oldindan qo`yilgan shart bilan takrorlanuvchi jarayonni tashkil qilish

Oldindan qo`yilgan shart bilan – «toki». Bu struktura quyidagi rasmdagi ko`rinishga ega. Bu strukturada takrorlanuvchi jarayon mantiqiy ifodaning qiymatiga qarab bajaradi yoki bajarmaydi. Agar mantiqiy ifodaning qiymati **true** bo`lsa, tsicl bajarilaveradi, aks xolda tsicl bajrilmaydi. Bu strukturada tsicl bir marta ham bajarilmasligi mumkin.

WHILE TAKRORLASH STRUKTURASI

Takrorlash strukturasi bir ifoda yoki blokni ma'lum bir shart to'g'ri (true) bo'lishi davomida qaytarish imkonini beradi. Qaytarilayotgan ifoda shartga ta'sir ko'rsatishishi kerak. Ma'lum bir vaqt o'tkandan keyin shart false ga o'zgartilishi kerak. Bo'lmasa while (davomida) tugatilmaydi. while faqat o'zidan keyin kelgan ifodaga ta'sir qiladi. Agar biz bir guruh amallarni qaytarmoqchi bo'lsak, ushbu blokni {} qavslar ichiga olishimiz kerak. Shart takrorlanuvchi blokning boshida tekshirilgani sababli, agar shart noto'g'ri bo'lib chiqsa, blokni hech ijro ko'rmasligi ham mumkin.



```
while (shart)
{
    operator 1;
    operator 2;

operator n; }
```

10 ning faktorialini hisoblovchi dastur blokini keltiraylik.

```
int factorial = 1;
int son = 1;
while (son < 11) {
    factorial = factorial * son;
    son = son + 1; }
```

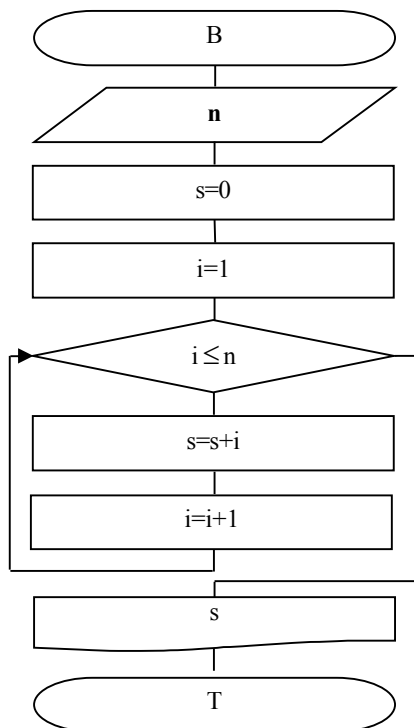
Bu yerda javobimiz factorial o'zgaruvchimizda saqlanmoqda. son o'zgaruvchimiz har takrorlanishda birga orttirilmoqda. son 11 ga yetganida while dagi shart false bo'ladi va takrorlanish tugatiladi. Yani son ning 11 qiymati javobga ta'sir ko'rsatmaydi. Biz qo'llagan son o'zgaruvchimiz sanovchi (counter)vazifasini bajaradi. Bu kabi o'zgaruvchilar vazifasiga ko'ra 1 yoki 0 ga tenglashtiriladi. Buni biz initsializatsiya deymiz. Initsializatsiya qilinmagan o'zgaruvchilar qiymatlari hotiradagi oldinroq ishlagan programmalar qoldiqlariga teng bo'ladi. Bu esa hatoga olib keladi. Shu sababli sanovchilarga boshlang'ich qiymat berib o'tilishi kerak.

MASALA: 1 dan n gacha bo'lgan sonlarni yig'indisini toping.

WHILE tsikl operatori uchun bajaramiz.

$$s = \sum_{i=1}^n i$$

Masalani echish algoritmining blok-sxema ko'rinishi.



Masalani echishning

C++ da dastur ko'rinishi

```
#include<iostream>;
using namespace std;
int main()
{
    int n,s,i;
    cout <<"n=";<<cin>>n;
    s=0; i=1;
    while(i<=n)
    {
        s=s+i;
        i++;
    }
    cout<<"s="<<s<<"\n";
    return 0;
}
//n=11
//s=66
```

«gacha» – oxirida qo`yilgan shart bilan takrorlanuvchi jarayonni tashkil qilish

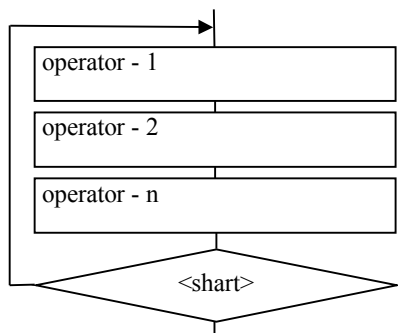
Bu strukturada mantiqiy ifodaning qiymati **false** bo`lsa, qayta bajariladi, aks xolda **true** da tsiqldan chiqib ketiladi. Bu strukturada tsikl soxasi xech bo`lmaganda bir marta bajariladi.

do/while TAKRORLASH STRUKTURASI

do/while ifodasi while strukturasi o`hshashdir. Bitta farqi shundaki while da shart boshiga tekshiriladi. do/while da esa takrorlanish tanasi eng kamida bir marta ijro ko`radi va shart strukturaning so`ngida test qilinadi. Shart true bo`lsa blok yana takrorlanadi. Shart false bo`lsa do/while ifodasidan chiqiladi. Agar do/while ichida qaytarilishi kerak bo`lgan ifoda bir dona bo`lsa {} qavslarning keragi yo`qdir. Quyidagicha bo`ladi:

```
do
  ifoda;
while (shart);
```

Lekin {} qavslarning yo`qligi dasturchini adashtirishi mumkin. Chunki qavssiz do/while oddiy while ning boshlanishiga o`hshaydi. Buni oldini olish uchun {} qavslarni har doim qo`yishni tavsiya etamiz.



```
do
operator1;
operator2;
...
operatorn;
while(shart);
```

```
int k = 1;
do {
    k = k * 5;
} while ( !(k>1000) );
```

Bu blokda 1000 dan kichik yoki teng bo`lgan eng katta 5 ga karrali son topilmoqda. while shartini ozroq o`zgartirib berdik, ! (not - inkor) operatorining ishlashini misolda ko`rsatish uchun. Agar oddiy qilib yozadigan

bo`lsak, while shartining ko`rinishi bunday bo`lardi: while (k<=1000); Cheksiz takrorlanishni oldini olish uchun shart ifodasining ko`rinishiga

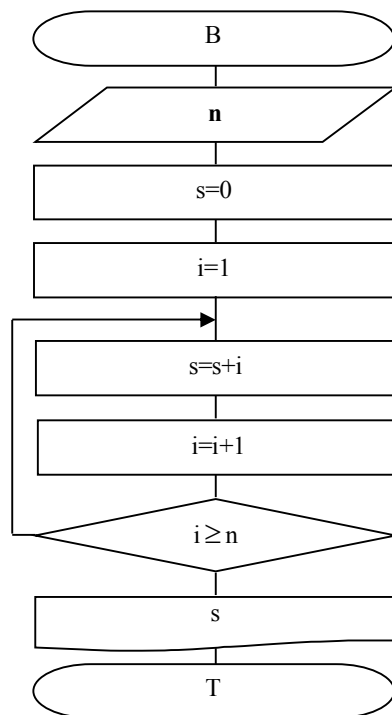
katta e'tibor berish kerak. Bir nuqtaga kelib shart true dan false qiymatiga o'tishi shart.

MASALA: 1 dan n gacha bo'lgan sonlarni yig'indisini toping.

DO-WHILE tsikl operatori uchun bajaramiz.

$$s = \sum_{i=1}^n i$$

Masalani echish algoritmining blok-sxema ko'rinishi.



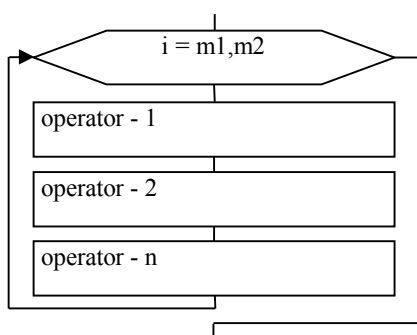
Masalani echishning C++ da dastur ko'rinishi.

```
#include<iostream>;
using namespace std;
int main()
{
  int n,s,i;
  cout <<"n=";
  cin>>n;
  s=0;
  i=1;
  do
  { s=s+i;
    i++;
  }
  while(i<=n);
  cout<<"s="<<s<<"\n";
  return 0;
}
//n=10
//s=55
```

Parametrli tsiiql operatori FOR yordamida takrorlanuvchi jarayonni xosil qilish.

FOR strukturasi sanovchi (counter) bilan bajariladigan takrorlashni bajaradi. Boshqa takrorlash bloklarida (while, do/while) takrorlash sonini control qilish uchun ham sanovchini qo'llasa bo'lardi, bu holda takrorlanish sonini o'ldindan bilsa bo'lardi, ham boshqa bir holatning vujudga kelish-kelmasligi orqali boshqarish mumkin edi. Ikkinchi holda ehtimol miqdori katta bo'ladi.

Masalan qo'llanuvchi belgilangan sonni kiritmaguncha takrorlashni bajarish kerak bo'lsa biz while ni ifodalar-ni ishlatamiz. for da esa sanovchi ifodaning qiymati oshirilib (kamaytirilib) boriladi, va chegaraviy qiymatni olganda takrorlanish tugatiladi. for ifodasidan keyingi bitta ifoda qaytariladi. Agar bir necha ifoda takrorlanishi kerak bo'lsa, ifodalar bloki {} qavs ichiga olinadi.



```
For(parametr1; parametr2; parametr3;)  
{  
  operator 1;  
  operator 2;  
  ...  
  operator n  
}  
//Ekkranda o'zgaruvching qiymatini yozuvchi  
dastur, for ni ishlatadi.
```

```
# include <iostream.h>  
int main()  
{  
  for (int i = 0; i < 5; i++){  
    cout << i << " ";  
  }  
  return 0 ;  
}  
Ekkranda:  
0 1 2 3 4
```

FOR strukturasi uch qismdan iboratdir. Ular nuqtavergul bilan bir-biridan ajratiladi. for ning ko'rinishi:

```
for( 1. qism ; 2. qism ; 3. qism ){  
    takror etiladigan blok  
}
```

1. qism - e'lon va initsializatsiya.
2. qism - shartni tekshirish (oz'garuvchini chegaraviy qiymat bilan solishtirish).
3. qism - o'zgaruvchining qiymatini o'zgartirish.

Qismlarning bajarilish ketma-ketligi quyidagichadir:

Boshida 1. qism bajariladi (faqat bir marta), keyin

2. qismdagi shart tekshiriladi va agar u true bo'lsa takrorlanish bloki ijro ko'radi, va eng ohirda 3. qismda o'zgaruvchilar o'zgartiriladi, keyin yana ikkinchi qismga

o'tiladi. for strukturamizni while struktura bilan almashtirib ko'raylik:

```
for (int i = 0; i < 10 ; i++)  
    cout << "Hello!"<< endl;
```

Ekranga 10 marta Hello! so'zi bosib chiqariladi. I o'zgaruvchisi 0 dan 9 gacha o'zgaradi. i 10 ga teng bo'lganda esa i < 10 sharti noto'g'ri (false) bo'lib chiqadi va for strukturasi nihoyasiga yetadi. Buni while bilan yozsak:

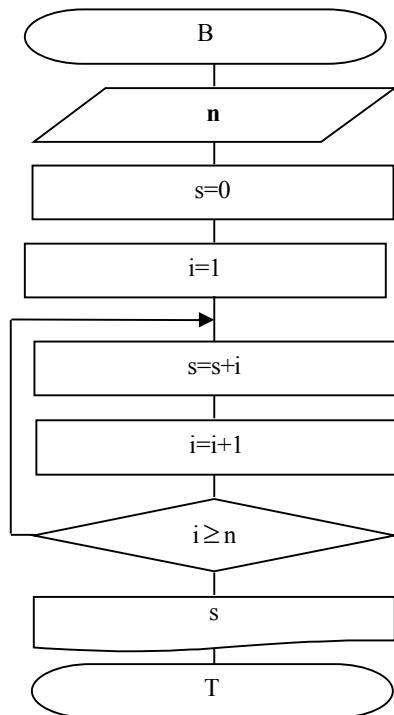
```
int i = 0;  
while ( i<10 ){  
    cout << "Hello!" << endl;  
    i++;}
```

MASALA: 1 dan n gacha bo'lgan sonlarni yig'indisini toping.

FOR tsikl operatori uchun bajaramiz.

$$S = \sum_{i=1}^n i$$

Masalani echish algoritmining blok-sxema ko'rinishi.



Masalani echishning C++ da dastur ko'rinishi.

```
using namespace std;
int main()
{
    int n,s,i;
    cout <<"n=";
    cin >>n;
    s=0;
    for ( i=1;i<=n;i++)
    {
        s=s+i;
    }
    cout<<"s="<<s<<"\n";
    return 0;
}
//n=12
//s=78
```

Nazorat sovolari:

1. FOR tsiqli qanday tashkil qilinadi?
2. FOR tsiqlini kamayish bo'yicha qanday tashkil qilish mumkin?
3. WHILE tsiqli qanday tashkil qilinadi?
4. FOR tsiqlini o'zi ichida parametrlarini o'zgartirish mumkinmi?
5. Tsiql nima uchun ishlatiladi?

Kurs ishining nazariy qismni bajarish talab va tartiblari.

Talabalar oldindan tuzilgan reja asosida o'z variantining nazariy qismi mavzusi to'liq yoritiladi va kurs ishini ximoya qilayotganda misollar yordamida tushuntirilib beriladi.

II. Kurs ishining amaliy qismni bajarish talab va tartiblari.

1. Chiziqli hisoblash jarayoni. Bu bosqichda ikkita funksiyaning qiymatlarini hisoblash jarayoni algoritmining blok sxema ko'rinishida WORD dasturida bajarilsin va dasturi C++ dasturlash tilida tuzilsin.

2. Tarmoqlanuvchi hisoblash jarayoni. Y funksiyaning qiymatlarini hisoblash algoritmi blok sxema ko'rinishida WORD dasturida bajarilsin va dasturi C++ dasturlash tilida tuzilsin. Y funksiyaning qiymatlari barcha shartlar bo'yicha topilsin.

3. Takrorlanuvchi hisoblash jarayoni. Y funksiyaning qiymatlarini hisoblash algoritmi blok sxema ko'rinishida WORD dasturida bajarilsin va dasturi C++ dasturlash tilida:

A) IF operatori yordamida tuzilsin.

B) FOR operatori yordamida tuzilsin.

C) WHILE operatori yordamida tuzilsin.

D) DO-WHILE operatori yordamida tuzilsin.

F) Funksiyaning qiymatlari barcha shartlar bo'yicha topilsin.

6. BAXOLASH MEZONI:

№	Bajarilishi lozim bo‘lgan talablar	Foyiz
1.	Oldindan tuzilgan reja asosida uz variantining nazariy kismi mavzusi to‘liq yoritilib, kurs ishini ximoya kilayotganda misollar yordamida to‘liq tushuntirib bergani uchun	10
2.	Oddiy hisoblash jarayoni. 2 ta funksiyaning qiymatlarini hisoblash algoritmi blok sxema ko‘rinishida WORD dasturida bajarilgani va dasturi C++ dasturlash tilida tuzilib natijalarni olgani uchun	15
3.	Tarmoklanuvchi hisoblash jarayoni. Y funksiyaning qiymatlarini hisoblash algoritmi blok sxema ko‘rinishida WORD dasturida bajarilgani va dasturi C++ dasturlash tilida, Y funksiyaning kiymatlari barcha shartlar bo‘yicha olgani uchun	15
4.	Takrorlanuvchi hisoblash jarayoni. Y funksiyaning qiymatlarini hisoblash algoritmi blok sxema ko‘rinishida WORD dasturida bajarilganligi va dasturi tarmoqlanuvchi C++ dasturlash tilida IF operatori yordamida tuzilib natijalarni barcha shartlar bo‘yicha olgani uchun	15
5.	Takrorlanuvchi hisoblash jarayoni. Y funksiyaning qiymatlarini hisoblash algoritmi blok sxema ko‘rinishida WORD dasturida bajarilganligi va dasturi takrorlanishlarni(siklni) C++ dasturlash tilida FOR operatori yordamida tuzilib natijalarni barcha shartlar bo‘yicha olgani uchun	15
6.	Takrorlanuvchi hisoblash jarayoni. Y funksiyaning qiymatlarini hisoblash algoritmi blok sxema ko‘rinishida WORD dasturida bajarilganligi va dasturi takrorlanishlarni(siklni) C++ dasturlash tilida WHILE operatori yordamida tuzilib natijalarni barcha shartlar bo‘yicha olgani uchun	15
7.	Takrorlanuvchi hisoblash jarayoni. Y funksiyaning qiymatlarini hisoblash algoritmi blok sxema ko‘rinishida WORD dasturida bajarilganligi va dasturi takrorlanishlarni(siklni) C++ dasturlash tilida DO-WHILE operatori yordamida tuzilib natijalarni barcha shartlar bo‘yicha olgani uchun	15
JAMI:		100

1-VARIANT**Nazariy qism mavzusi:**

**Algoritmik tillarning klassifikatsiyasi.
Algoritmik tillar to'g'risida ma'lumotlar.**

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$y = \lg|x^2 + 4| + \operatorname{tg}^2 x + e^{x+5}$$

$$z = a^x \sin(x-a) + \cos x^3$$

$$a = 4,475$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$y = \begin{cases} \cos^2 x - \sin x^2 + 4,3x & \text{agar } x < 1 \\ 2\sin^2 x^2 + x^3 + 5c & \text{agar } x > 2 \\ \sin^3 x^2 + x^4 + c & \text{agar } 1 \leq x \leq 2 \end{cases}$$

$$c = -5,575$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$y = \begin{cases} 1,5\cos^2 x & x < 1 & \text{bo'lsa} \\ 1,8ax & x = 1 & \text{bo'lsa} \\ (x+2)^2 + 6 & 1 < x < 2 & \text{bo'lsa} \\ 3\operatorname{tg} x & x > 2 & \text{bo'lsa} \end{cases}$$

$$a = 2,3;$$

$$x \in [0,2; 2,8]; \quad \Delta x = 0,2$$

2-VARIANT**Nazariy qism mavzusi:**

**Algoritmik asosiy tillari. Mantiqiy va munosabat amallari.
Malumotlarning standart turlari.**

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$R = x^2(x+1)/b - \sin^2(x+a)$$

$$s = \sqrt{xb/a} + \cos^2(x+b)^2$$

$$a=0,7; \quad b=0,05; \quad x=0,5$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} bx - \lg bx & bx < 1 \quad bo'lsa \\ 1 & bx = 1 \quad bo'lsa \\ bx + \lg bx & bx > 1 \quad bo'lsa \end{cases}$$

$$b = 1,5$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} \pi x^2 + 7/x^2 & x \leq 1,4 \quad bo'lsa \\ ax^3 - 7/\sqrt{x} & x = 1,4 \quad bo'lsa \\ \ln(x + 7\sqrt{|x+a|}) & x > 1,4 \quad bo'lsa \end{cases}$$

$$a = 1,65$$

$$x \in [0,7; 2]; \quad \Delta x = 0,1$$

3-VARIANT

Nazariy qism mavzusi:

Algoritmnining asosiy turlari. Chiziqli tarkibli turi. O'zgaruvchilar, O'zgarmaslar.

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jaroyoni:

$$y = e^{-bt} \sin(at + b) - \sqrt{|bt + a|}$$

$$s = b \sin(at^2 \cos 2t) - 1$$

$$a = -0,5; \quad b = 1,7; \quad t = 0,44$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} x^3 \sqrt{x-a} & x > a \quad bo'lsa \\ x \sin ax & x = a \quad bo'lsa \\ e^{-ax} \cos ax & x < a \quad bo'lsa \end{cases}$$

$$a = 2,5$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$Y = \begin{cases} \ln(x+4) & x > 1 \quad bo'lsa \\ e^x & x = 1 \quad bo'lsa \\ \cos^2(1+|ax|) & x < 1 \quad bo'lsa \end{cases}$$

$$a = 15,7$$

$$x \in [0,4;2]; \quad \Delta x = 0,1$$

4-VARIANT

Nazariy qism mavzusi:

Algoritmning asosiy turlari. Tarmoqlanuvchi turi. Standart funksiyalar.

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$f = \ln(a+x^2) + \sin^2(x/b)$$

$$z = e^{-cx} \frac{x + \sqrt{x+a}}{x - \sqrt{|x-b|}}$$

$$a=10,2; \quad b=9,2; \quad x=2,2; \quad c=0,5$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} \frac{a+b}{e^x + \cos x} & x < 2,8 \quad bo'lsa \\ (a+b)/x+1 & 2,8 \leq x < 6 \quad bo'lsa \\ e^x \sin x & x \geq 6 \quad bo'lsa \end{cases}$$

$$a=2,6; \quad b=-0,39$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$Y = \begin{cases} \frac{a+b}{e^x + \cos x} & x < 2,8 \quad bo'lsa \\ (a+b)/(x+1) & 2,8 \leq x \leq 6 \quad bo'lsa \\ e^x + \sin x & x \geq 6 \quad bo'lsa \end{cases}$$

$$a=2,6; \quad b=-0,39$$

$$x \in [0;7]; \quad \Delta x = 0,5$$

5-VARIANT

Nazariy qism mavzusi:

Algoritmning asosiy turlari. Takrorlanuvchi turi.

Arifmetik ifodalar.

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$f = \ln(a + x^2) + \sin^2(x/b)$$

$$z = e^{-cx} \frac{x + \sqrt{x+a}}{x - \sqrt{|x-b|}}$$

$$a = 10,2; \quad b = 9,2; \quad x = 2,2; \quad c = 0,5$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} \frac{a+b}{e^x + \cos x} & x < 2,8 \quad \text{bo'lsa} \\ (a+b)/(x+1) & 2,8 \leq x < 6 \quad \text{bo'lsa} \\ e^x + \sin x & x \geq 6 \quad \text{bo'lsa} \end{cases}$$

$$a = 2,6; \quad b = -0,39$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$Y = \begin{cases} \sin x & x \leq a \quad \text{bo'lsa} \\ \cos x & a < x < b \quad \text{bo'lsa} \\ \operatorname{tg} x & x \geq b \quad \text{bo'lsa} \end{cases}$$

$$a = 4,2; \quad b = 5,7$$

$$x \in [0; 7]; \quad \Delta x = 1$$

6-VARIANT

Nazariy qism mavzusi:

C++ tilida tuzilgan dasturlar strukturasi.

C++ tilida izoh yozish.

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$y = \begin{cases} t = \ln x + y^k x + 5a^2 \\ c = \frac{ae^{\cos x} + \sqrt{\cos x}}{1 + \ln \sqrt{x^3 - b^3}} \end{cases}$$

$$a = 5,33; \quad b = 1; \quad x = 3011; \quad y = 1; \quad k = 0,1$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} \frac{a+b}{e^x + \cos x} & x < 2,8 \quad \text{bo'lsa} \\ (a+b)/(x+1) & 2,8 \leq x < 6 \quad \text{bo'lsa} \\ e^x + \sin x & x \geq 6 \quad \text{bo'lsa} \end{cases}$$

$$a = 2,6 ; b = -0,39$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$Y = \begin{cases} \sin x & x \leq a \quad \text{bo'lsa} \\ \cos x & a < x < b \quad \text{bo'lsa} \\ \operatorname{tg} x & x \geq b \quad \text{bo'lsa} \end{cases}$$

$$a = 4,2 ; b = 5,7$$

$$x \in [0;7] ; \Delta x = 1$$

7-VARIANT

Nazariy qism mavzusi:

C++ algoritmik tilining asosiy operatorlari.

Ta'nlash operatori.

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$y = \operatorname{atg}^3 x^2 + \sqrt{\frac{z^2}{a^2 + x^2}}$$

$$p = \ln(a + x^2) + \sin^2 \frac{z}{a}$$

$$a = 0,59 ; z = -4,8 ; x = 2,1$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$y = \begin{cases} 1,5 \cos^2 x & x < 1 \quad \text{bo'lsa} \\ 1,8ax & x = 1 \quad \text{bo'lsa} \\ (x-2)^2 + 6 & 1 < x < 2 \quad \text{bo'lsa} \\ 3\operatorname{tg} x & x > 2 \quad \text{bo'lsa} \end{cases}$$

$$a = 2,3$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$Y = \begin{cases} \sqrt{at^2 + bsint + 1} & t < 0,1 \text{ bo'lsa} \\ at + b & t = 0,1 \text{ bo'lsa} \\ \sqrt{at^2 + bcost + 1} & t > 0,1 \text{ bo'lsa} \end{cases}$$

$$a = 2,5; \quad b = 0,4; \quad t \in [-1; 1]; \quad \Delta t = 0,2$$

8-VARIANT

Nazariy qism mavzusi:

Mantiqiy ifodalar.

Kiritish, chiqarish operatori.

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$z = e^{at} \cos(bt + a) - \sqrt[3]{|at + b|}$$

$$Q = \sin^2 t - t / \sqrt{a^2 + b^2}$$

$$a = 0,15; \quad t = 3,1; \quad b = 1,4$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$y = \begin{cases} bx^2 \ln x & 2 \leq x \leq 3 \text{ bo'lsa} \\ 1 & x < 1 \text{ bo'lsa} \\ \cos^2 x + a \sin x^2 & x > 3 \text{ bo'lsa} \end{cases}$$

$$a = 0,7; \quad b = -2$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$y = \begin{cases} \pi x^2 - 7/x^2 & x < 1,3 \text{ bo'lsa} \\ ax^3 + 7\sqrt{x} & x = 1,3 \text{ bo'lsa} \\ \lg(x + 7\sqrt{x}) & x > 1,3 \text{ bo'lsa} \end{cases}$$

$$a = 1,5; \quad x \in [0,82]; \quad \Delta x = 0,1$$

9-VARIANT

Nazariy qism mavzusi:

Dinamik massivlar funksiya argumenti sifatida.

Sharsiz o'tish operatori.

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$y = \sin^3(x^2 + a) - \sqrt{x/b}$$

$$z = \frac{x^2}{a} + \cos(x + b)^3$$

$$a = 0,7 ; b = 0,004 ; x = 0,2$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} a \sin(i^2 + 1) & \sin(i^2 + 1) > 0 \quad bo'lsa \\ \cos\left(i + \frac{1}{n}\right) & \sin\left(\frac{i^2 + 1}{n}\right) < 1 \quad bo'lsa \end{cases}$$

$$a = 0,3 ; n = 10$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$y = \begin{cases} bx^2 \ln x & 2 \leq x \leq 3 \quad bo'lsa \\ 1 & x < 1 \quad bo'lsa \\ \cos^2 x + a \sin x^2 & x > 3 \quad bo'lsa \end{cases}$$

$$a = 0,7; b = -2$$

$$x \in [0,1;3]; \Delta t = 0,15$$

10-VARIANT

Nazariy qism mavzusi:

Shartli o'tish operatorining to'liq va to'liqmas farmatlari.

- Tanlash operatori

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$y = \frac{a^{2x} + b^{-x} \cos(a + b)x}{x + 1}$$

$$R = \sqrt{x^2 + b} - b^2 \sin^3(x + a)/x$$

$$a = 0,3 ; b = 0,9 ; x = 0,61$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$y = \begin{cases} a \lg x + \sqrt[3]{|x|} & x > 1 \quad bo'lsa \\ 2a \cos x + 3x^2 & x \leq 1 \quad bo'lsa \end{cases}$$

$$a = 0,9$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$Y = \begin{cases} ax^3 + bx^2 + cx + d & x < 1,3 \text{ bo'lsa} \\ (a-b)/x & x = 1,3 \text{ bo'lsa} \\ a + cx/\sqrt{c+dx^2} & x < 1,3 \text{ bo'lsa} \end{cases}$$

$a = 2,7; b = -0,2; c = 5; d = 1,2$
 $x \in [1;2]; \Delta x = 0,05$

11-VARIANT

Nazariy qism mavzusi:

Tanlash operatori.

Takrorlanuvchi operatorlar

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$z = \cos^2(x^3 + b) + \sqrt[3]{x/a}$$

$$y = \operatorname{ntg}(at \cos t^2) + nc$$

$$x = 1,12; b = -1,17; c = 0,1; n = -7,8$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} e^2 + \cos x & x < 0,4 \text{ bo'lsa} \\ \cos^2 x & x = 0,4 \text{ bo'lsa} \\ e^{-ax} \cos ax & x > 0,4 \text{ bo'lsa} \end{cases}$$

$$a = 0,5;$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$y = \begin{cases} bx^2 \ln x & 2 \leq x \leq 3 \text{ bo'lsa} \\ 1 & x < 1 \text{ bo'lsa} \\ \cos^2 x + a \sin x^2 & x > 3 \text{ bo'lsa} \end{cases}$$

$$a = 0,7; b = -2$$

$$x \in [0,1;3]; \Delta t = 0,15$$

12-VARIANT

Nazariy qism mavzusi:

Sikl operatori.

Parametrlilik sikl.

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$y = \frac{a^{2x} + b^{-x} \cos(a+b)x}{x+1}$$

$$R = \sqrt{x^2 + b} - b^2 \sin^3(x+a)/x$$

$$a = 0,3 ; b = 0,9 ; x = 0,61$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} ax^2bx+c & x < 1,2 \text{ bo'lsa} \\ a/x + \sqrt{x^2+1} & x = 1,2 \text{ bo'lsa} \\ (a+bx)/\sqrt{x^2+1} & x > 1,2 \text{ bo'lsa} \end{cases}$$

$$a = 2,8 ; b = -0,3 ; c = 4$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$Y = \begin{cases} bx - \lg bx & bx < 1 \text{ bo'lsa} \\ 1 & bx = 1 \text{ bo'lsa} \\ bx + \lg bx & bx > 1 \text{ bo'lsa} \end{cases}$$

$$b = 1,5 ; x \in [0,1;1] ; \Delta x = 0,1$$

13-VARIANT

Nazariy qism mavzusi:

Sharti oldin va keyin qo'yilgan sikl operatorlari.

- Ma'lumotlar turlari

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$w = \sqrt{x^2 + b} - b^2 \sin^3(x+a)/x$$

$$y = \cos^2 x^3 - x/\sqrt{a^2 + b^2}$$

$$a = 1,5 ; b = 15,5 ; x = -2,9$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} e^x + \cos x & x < 0,4 \text{ bo'lsa} \\ \cos^2 x & x = 0,4 \text{ bo'lsa} \\ e^{-ax} \cos ax & x > 0,4 \text{ bo'lsa} \end{cases}$$

$$a = 0,5;$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$y = \begin{cases} at^2 \ln t & 1 \leq t \leq 2 \text{ bo'lsa} \\ 1 & t < 1 \text{ bo'lsa} \\ e^{at} \cos bt & t > 2 \text{ bo'lsa} \end{cases}$$

$$a = -0,5; \quad b = 2$$

$$t \in [0,3]; \quad \Delta t = 0,15$$

14-VARIANT

Nazariy qism mavzusi:

Ichma-ich joylashgan sikllar.

Standart funksiyalar

Amaliy qism masalalari.

1. Chiziqli tarkibli hisoblash jarayoni:

$$z = \frac{\sin x}{\sqrt{1+m^2 \sin^2 x}} - c \ln mx$$

$$s = e^{-ax} \sqrt{x+1} + e^{-bx} \sqrt{x+1,5}$$

$$m=0,5; \quad c=2,1; \quad x=1,7; \quad a=0,5; \quad b=1,08$$

2. Tarmoqlanuvchi tarkibli hisoblash jaroyoni.

$$Y = \begin{cases} ax^2 bx + c & x < 1,2 \text{ bo'lsa} \\ a/x + \sqrt{x^2 + 1} & x = 1,2 \text{ bo'lsa} \\ (a+bx)/\sqrt{x^2 + 1} & x > 1,2 \text{ bo'lsa} \end{cases}$$

$$a = 2,8; \quad b = -0,3; \quad c = 4$$

3. Takrorlanuvchi (siqlik) tarkibli hisoblash jarayoni.

$$y = \begin{cases} \sin x \lg ax & x > 3,5 \\ \cos^2 bx & x \leq 3,5 \end{cases}$$

$$a = 1,313; \quad b = -20,7$$

$$x \in [2,5]; \quad \Delta x = 0,25$$

15-VARIANT

Nazariy qism mavzusi:

Massivlar bilan ishlash.

Ikki o'lchovli massivlar.

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jarayoni:

$$y = \sin^3(x^2 + a) - \sqrt{x/b}$$

$$z = \frac{x^2}{a} + \cos(x + b)^3$$

$$a = 0,7 ; b = 0,004 ; x = 0,2$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$Y = \begin{cases} \sqrt{at^2 + bsint + 1} & t < 0,1 \text{ bo'lsa} \\ at + b & t = 0,1 \text{ bo'lsa} \\ \sqrt{at^2 + bcost + 1} & t > 0,1 \text{ bo'lsa} \end{cases}$$

$$a = 2,5 ; b = 0,4$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jarayoni.

$$Y = \begin{cases} a \sin\left(\frac{i^2 + 1}{n}\right) & \sin\left(\frac{i^2 + 1}{n}\right) > 0 \text{ bo'lsa} \\ \cos\left(i + \frac{1}{n}\right) & \sin\left(\frac{i^2 + 1}{n}\right) < 1 \text{ bo'lsa} \end{cases}$$

$$a = 0,3 ; n = 10$$

$$i \in [1,10] ; \Delta i = 1$$

16-VARIANT

Nazariy qism mavzusi:

Funksiyalar

Identifikatorlar va ularning turlari

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jarayoni:

$$u = \frac{a^2 x + e^{-x} \cos bx}{bx - e^{-x} \sin bx + 1}$$

$$f = e^{2x} \ln(a + x) - b^{3x} \ln(b - x)$$

$$a = 0,5 ; b = 2,9 ; x = 0,3$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$y = \begin{cases} \pi x^2 - 7/x^2 & x < 1,3 \text{ bo'lsa} \\ ax^3 + 7\sqrt{x} & x = 1,3 \text{ bo'lsa} \\ \lg(x + 7\sqrt{x}) & x > 1,3 \text{ bo'lsa} \end{cases}$$

$$a = 1,5$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jarayoni.

$$Y = \begin{cases} (\ln^3 x + x^2) / \sqrt{x+t} & x < 0,5 \text{ bo'lsa} \\ \sqrt{x+t} + 1/x & x = 0,5 \text{ bo'lsa} \\ \cos x + t \sin^2 x & x > 0,5 \text{ bo'lsa} \end{cases}$$

$$t = 2,2 ; x \in [0,2;2] ; \Delta x = 0,2$$

17-VARIANT

Nazariy qism mavzusi:

- C++ da funksiyalar
- switch tarmoqlanish operatori

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jarayoni:

$$R = x^2(x+1)/b - \sin^2(x+a)$$

$$s = \sqrt{xb/a} + \cos^2(x+b)^3$$

$$a = 0,7 ; b = 0,05 ; x = 0,5$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$Y = \begin{cases} (\ln^3 x + x^2) / \sqrt{x+t} & x < 0,5 \text{ bo'lsa} \\ \sqrt{x+t} + 1/x & x = 0,5 \text{ bo'lsa} \\ \cos x + t \sin^2 x & x > 0,5 \text{ bo'lsa} \end{cases}$$

$$t = 2,2 ;$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jarayoni.

$$Y = \begin{cases} ax^2 + bx + c & x < 1,2 \text{ bo'lsa} \\ a/x + \sqrt{x^2 + 1} & x = 1,2 \text{ bo'lsa} \\ (a+bx)\sqrt{x^2 + 1} & x > 1,2 \text{ bo'lsa} \end{cases}$$

$$a = 2,8 ; b = -0,3 ; c = 4$$

$$x \in [1;2] ; \Delta x = 0,05$$

18-VARIANT

Nazariy qism mavzusi:

**C++ tilining grafik operatorlari va funksiyalari.
Mantiqiy ifodalar**

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jarayoni:

$$z = e^{at} \cos(bt + a) - \sqrt[3]{|at + b|}$$

$$Q = \sin^2 t - t / \sqrt{a^2 + b^2}$$

$$a = 0,15; t = 3,1; b = 1,4$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$y = \begin{cases} \sin x \lg a x & x > 3,5 \\ \cos^2 b x & x \leq 3,5 \end{cases}$$

$$a = 1,313; b = -207$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jarayoni.

$$Y = \begin{cases} \lg(x+1) & x > 1 \text{ bo'lsa} \\ \sin^2 \sqrt{ax} & x \leq 1 \text{ bo'lsa} \end{cases}$$

$$a = 20,3; x \in [0,5; 2]; \Delta x = 0,1$$

19-VARIANT

Nazariy qism mavzusi:

- Statik massivlar funksiya argumenti sifatida
- if-else shart operatori

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jarayoni:

$$f = \sqrt[3]{mtgt + |c \sin t|}$$

$$z = m \cos(btsint) + c$$

$$m = 2; c = -1; t = 1,2; b = 0,7$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$Y = \begin{cases} (\ln^3 x + x^2) / \sqrt{x+t} & x < 0,5 \text{ bo'lsa} \\ \sqrt{x+t} + 1/x & x = 0,5 \text{ bo'lsa} \\ \cos x + t \sin^2 x & x > 0,5 \text{ bo'lsa} \end{cases}$$

$$t = 2,2;$$

3 Takrorlanuvchi (siqlik) tasrkibli xisoblash jarayoni.

$$Y = \begin{cases} bx - \lg bx & bx < 1 \text{ bo'lsa} \\ 1 & bx = 1 \text{ bo'lsa} \\ bx + \lg bx & bx > 1 \text{ bo'lsa} \end{cases}$$

$b = 1,5 ; x \in [0,1;1] ; \Delta x = 0,1$

20-VARIANT

Nazariy qism mavzusi:

**Strukturalar
Satrlar.**

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jarayoni:

$$y = \sin^3(x^2 + a) - \sqrt{x/b}$$

$$z = \frac{x^2}{a} + \cos(x + b)^3$$

$$a = 0,7 ; b = 0,004 ; x = 0,2$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$y = \begin{cases} at^2 \ln t & 1 \leq t \leq 2 \text{ bo'lsa} \\ 1 & t < 1 \text{ bo'lsa} \\ e^{at} \cos bt & t > 2 \text{ bo'lsa} \end{cases}$$

$$a = -0,5 ; b = 2$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jarayoni.

$$y = \begin{cases} at^2 \ln t & 1 \leq t \leq 2 \text{ bo'lsa} \\ 1 & t < 1 \text{ bo'lsa} \\ e^{at} \cos bt & t > 2 \text{ bo'lsa} \end{cases}$$

$$a = -0,5 ; b = 2$$

$$t \in [0,3] ; \Delta t = 0,15$$

21-VARIANT

Nazariy qism mavzusi:

**C++ tilida to'plamlar.
C++ tilida ifodalar**

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jarayoni:

$$y = a \sin 2x - \frac{b}{\operatorname{tg}^2\left(\frac{x}{b}\right)}$$

$$z = \sqrt{\frac{xa}{b}} + \sin^2(x+a)^2$$

$$a = 0,3 \quad b = 0,07 \quad x = 0,4$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$y = \begin{cases} at^2 \ln t & 1 \leq t \leq 2 \text{ bo'lsa} \\ 1 & t < 1 \text{ bo'lsa} \\ e^{at} \cos bt & t > 2 \text{ bo'lsa} \end{cases}$$

$$a = -0,5; \quad b = 2$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jarayoni.

$$Y = \begin{cases} ax^2 + bx + c & x < 1,2 \text{ bo'lsa} \\ a/x + \sqrt{x^2 + 1} & x = 1,2 \text{ bo'lsa} \\ (a+bx)\sqrt{x^2 + 1} & x > 1,2 \text{ bo'lsa} \end{cases}$$

$$a = 2,8; \quad b = -0,3; \quad c = 4$$

$$x \in [1; 2]; \quad \Delta x = 0,05$$

22-VARIANT

Nazariy qism mavzusi:

Ma'lumotlarning faylli toifasi.

Toifalashgan va toifalashmagan fayllar. r.

Amaliy qism masalalari.

1. Chiziqli tarkibli tarkibli xisoblash jarayoni:

$$t = \ln x + y^k x + 5a^2$$

$$c = \frac{ae^{\cos x} + \sqrt{cox}}{1 + \ln \sqrt{x^3 - b^3}}$$

$$a = 5,33; \quad b = 1; \quad x = 30,11$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$y = \begin{cases} (\lg^3 x + 2x^3) / \sqrt[3]{x+z} & x < 0,5 \text{ bo'lsa} \\ \lg x & x = 0,5 \text{ bo'lsa} \\ \sin^2 z + x \cos^2 z & x > 0,5 \text{ bo'lsa} \end{cases}$$

$$z = 37,1$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jaroyoni.

$$y = \begin{cases} \sin x \lg ax & x > 3,5 \\ \cos^2 bx & x \leq 3,5 \end{cases}$$

$$a = 1,313 ; b = -20,7$$

$$x \in [2; 5]; \Delta x = 0,25$$

23-VARIANT

Nazariy qism mavzusi:

**Matnli fayllar va ular uchun mo'ljallangan funksiyalar.
while, do-while takrorlash operatorlari**

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jaroyoni:

$$Q = \frac{\operatorname{tg}(x - \pi/6)}{2 + \cos^2 y}$$

$$t = \frac{x + b\sqrt{x+a}}{x + b\sqrt{|x-a|}}$$

$$a = 11; b = 1,9; x = 2,3; y = 0,3$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$y = \begin{cases} \pi x^2 - 7/x^2 & x < 1,3 \text{ bo'lsa} \\ ax^3 + 7\sqrt{x} & x = 1,3 \text{ bo'lsa} \\ \lg(x + 7\sqrt{x}) & x > 1,3 \text{ bo'lsa} \end{cases}$$

$$a = 1,5$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jaroyoni.

$$Y = \begin{cases} (\ln^3 x + x^2) / \sqrt{x+t} & x < 0,5 \text{ bo'lsa} \\ \sqrt{x+t} + 1/x & x = 0,5 \text{ bo'lsa} \\ \cos x + t \sin^2 x & x > 0,5 \text{ bo'lsa} \end{cases}$$

$$t = 2,2; x \in [0,2; 2]; \Delta x = 0,2$$

24-VARIANT

Nazariy qism mavzusi:

- Ma'lumotlarning murojat toifasi. Ko'rsatkich turlari va ularni tavsiflash.

Algoritmik tillar to'g'risida ma'lumotlar.

Amaliy qism masalalari.

1. Chiziqli tarkibli xisoblash jarayoni:

$$s = x^2 tg^2(x+b)^2 + \frac{a}{\sqrt{x+b}}$$

$$Q = \frac{bx^2 - a}{e^{ax} - 1}$$

$$a = 16,5 ; \quad b = 3,4 ; \quad x = 0,61$$

2. Tarmoqlanuvchi tarkibli xisoblash jaroyoni.

$$Y = \begin{cases} \sqrt{at^2 + bsint} + 1 & t < 0,18 \text{ bo'lsa} \\ at + b & t = 0,18 \text{ bo'lsa} \\ \sqrt{at^2 + bcost} + 1 & t > 0,18 \text{ bo'lsa} \end{cases}$$

$$a = 2,5 ; \quad b = 0,4$$

3. Takrorlanuvchi (siqlik) tarkibli xisoblash jarayoni.

$$Y = \begin{cases} \lg(x+1) & x > 1 \text{ bo'lsa} \\ \sin^2 \sqrt{|ax|} & x \leq 1 \text{ bo'lsa} \end{cases}$$

$$a = 20,3 ; \quad x \in [0,5; 2] ; \quad \Delta x = 0,1$$

5. FOYDALANILGAN ADABIYOTLAR VA INTERNET SAYTLAR.

Asosiy adabiyotlar:

1. M.Mominov Informatika O`quv qo`llanma Tafakur bo`stoni 2014 .344 b
2. P. Кабулов ва б. С ва С++ тили Дарслик Т.: Ворис нашриёти 2013. 408-б.
3. Richard L. Halterman Fundamentals of C++ Programming. Copyright © 2008–2016. All rights reserved. 634 pg.
4. Horstmann C.S. - C++ for Everyone, 2 edition - 2011 San Jose State University , Printed in the United States of America
5. Madraximov SH. F, G`aybnazarov S. M. C++ tilida programmalash asoslari bo`yicha uslubiy qo`llanma . M. Ulug`bek nomidagi UzMU bosmaxonasi. Toshkent 2009 y.

Qo`shimcha adabiyotlar:

1. Yusupov.F. “Informatika va dasturlash” Uslubiy qo`llanma., 2003 Urganch., 78 bet.
2. A.A. Abduqodirov., E.I.Kuznetsov "Hisoblash matematikasi va programmalashdan laboratoriya ishlari"-Toshkent "O`qituvchi", 1987 y.
3. Simonovich N.K. INFORMATIKA (dlya texnicheskix vuzov). SPb.: Piter, 1999.- 636s.
4. Aripov M. Xaydarov A. Informatika asoslari. T.: O`zbekiston, 2002
5. Informatika va axborot texnologiyalari fanidan laboratoriya mashg`ulotlarni bajarish bo`yicha uslubiy qo`llanma t.f.n. M. SHaripov, S. Jafarov ,S. Matlatipov O`zbekiston 2016 yil.

Internet saytlari

1. <http://sunschool.math.rsu.ru>
2. <http://it.mmf.rsu.ru/forum>
3. www.intuit.ru
4. www.kursovik.com
5. <http://studentbank.ru/>
6. www.dl.urdu.uz

MUNDARIJA

1.	Kirish.....	3
2.	Algoritmash asoslari.....	5
3.	Chiziqli jayonlarga doir algoritmlar.....	9
4.	Tarmoqlanuvchi jayonlarga doir algoritmlar.....	10
5.	Takrorlanuvchi jayonlarga doir (siqlik) algoritmlar.....	11
6.	C++ dasturlash tili.....	12
7.	Chiziqli jayonlarni dasturlash.....	17
8.	Tarmoqlanuvchi algoritmlarni dasturlash usullari.....	18
9.	Takrorlanuvchi jarayonlarga dastur tuzish.....	27
10.	Kurs ishini bajarishga doir talab va tartiblar.....	34
11.	Baxolash mezoni.....	35
12.	Ilova.....	36
13.	Foydalanilgan adabiyotlar va internet saytlar.....	54

Razzaqov Baxtiyor, Jafarov San`atbek Komilovich, Matkarimov Sanjar
Axmedovich

C++ DASTURLASH TILIDA BAJARISH BO'YICHA USLUBIY QO'LLANMA

Muharrir : Yo`ldoshev Ro`zimboy
Texnik muharrir: Sherali Yo`ldoshev
Musahhih: Quvondiq Ollayarov

UrDU noshirlik bo`limi O`zbekiston matbuot va axborot agentligining
2009 yil 19-avgustdagi № 148 raqamli buyrug`i bilan qayta ro`yxatdan
o`tkazilgan.

Terishga berildi: _____

Bosishga ruxsat etildi: _____

Ofset qog`ozi. Qog`oz bichimi 60x84 1/16.

Tayms garniturası. Adadi 50.

Buyurtma № _____

Hisob-nashriyot tabag`i 2,5

Shartli bosma tabag`i 2,3

UrDU noshirlik bo`limida tayyorlanadi.

Manzil: 220110. Urganch shahari,

H. Olimjon ko`chasi, 14-uy.

Telefon : (0-362)-224-66-01.

Urganch Davlat universiteti bosmaxonasida chop etildi.

Manzil: 220110. Urganch shahri,

Hamid Olimjon ko`chasi, 14 uy.

Telefon : (0-362)-224-66-01.