

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

**«TOSHKENT IRRIGATSIYA VA QISHLOQ XO‘JALIGINI  
MEXANIZATSIYALASH MUHANDISLARI INSTITUTI» MILLIY  
TADQIQOT UNIVERSITETI**

**«AXBOROT TEXNOLOGIYALARI»**

kafedrası



**«AXBOROT TEXNOLOGIYALARI VA JARAYONLARNI  
MATEMATIK MODELLASHTIRISH»**

**FANIDAN LABORATORIYA MASHG‘ULOTLARINI BAJARISH  
BO‘YICHA**

**USLUBIY QO‘LLANMA**

**(“Suv xo‘jaligi melioratsiyasi ishlarini mexanizatsiyalash”, “Mehnat muhofazasi va texnika xavfsizligi” ( suv xo‘jaligida), “Qishloq xo‘jaligini mexanizatsiyalash”, “Qishloq va suv xo‘jaligida texnik servis”, “Qishloq xo‘jaligida innovatsion texnika va texnologiyalarni qo‘llash”, “Professional ta‘lim” (Qishloq xo‘jaligini mexanizatsiyalashtirish) bakalavriat ta‘limi yo‘nalishlari uchun)**

**Toshkent -2022**

Ushbu uslubiy qo‘llanma institut Ilmiy-uslubiy Kengashining 2022 yil “23” fevralda bo‘lib o‘tgan 2– sonli majlisida ko‘rib chiqildi va chop etishga tavsiya etildi.

Uslubiy qo‘llanmada «Axborot texnologiyalari va jarayonlarni matematik modellashtirish» fani bo‘yicha laboratoriya mashg‘ulot topshiriqlari hamda ularni bajarish bo‘yicha ko‘rsatmalar keltirilgan. Uslubiy qo‘llanma bakalavriatning “Suv xo‘jaligi melioratsiyasi ishlarini mexanizatsiyalash”, “Mehnat muhofazasi va texnika xavfsizligi” ( suv xo‘jaligida), “Qishloq xo‘jaligini mexanizatsiyalash”, “Qishloq va suv xo‘jaligida texnik servis”, “Qishloq xo‘jaligida innovatsion texnika va texnologiyalarni qo‘llash”, “Professional ta’lim” (Qishloq xo‘jaligini mexanizatsiyalashtirish) bakalavriat ta’limi yo‘nalishlari talabalari uchun mo‘ljallangan.

Tuzuvchilar: Sh.A. Aynakulov, katta o‘qituvchi  
D. Ziyodullayev, dotsent

Taqrizchilar: D. K. Sharipov, Muxammad al-Xorazmiy nomidagi  
TATU «Multimedia texnologiyalari» kafedrası  
dotsenti, t.f.f.d.(PhD)

S. S. Mirzayev, dotsent , f.-m.f.n.

© «Toshkent irrigatsiya va qishloq xo‘jaligini mexanizatsiyalash muhandislari instituti» Milliy tadqiqot universiteti, 2022 y.

## **Kirish**

Hozirgi kunda «Axborot texnologiyalari va jarayonlarni matematik modellashtirish» fanining jamiyat taraqqiyotida muhim ahamiyat kasb etmoqda. Axborot texnologiyalari jadallik bilan rivojlanib, fan, ishlab chiqarish, umuman jamiyatning turli sohalarida keng qoʻllanilmoqda. Axborot texnologiyalari sohasidagi chuqur oʻzgarishlar mamalakatimiz taʼlim tizimida ham oʻz aksini topmoqda va bu Oʻzbekiston Respublikasining «Taʼlim toʻgʻrisida»gi qonuni va «Kadrlar tayyorlash milliy dasturi»da oʻz ifodasini topgan.

«Axborot texnologiyalari va jarayonlarni matematik modellashtirish» kursida ushbu fanning asoslari, kompyuter arxitekturasi, texnik va dasturiy taʼminoti, lokal va global kompyuter tizimlari, dasturlash texnologiyalar, qishloq va suv xoʻjaligi masalalarining modelini tuzish va uni yechish usullari bilan tanishish maqsad qilib qoʻyilgan.

Mazkur uslubiy qoʻllanmada talabalarga suv xoʻjaligi va mexanizatsiya masalalarini kompyuterda yechish algoritmlari, ularni C++ dasturlash tilida bajarish qoidalari, laboratoriya topshiriqlari, mustaqil bajarish uchun topshiriqlar hamda foydalanish boʻyicha koʻrsatmalar, adabiyotlar berilgan.

## 1-laboratoriya ishi.

### Mavzu: Algoritmash asoslari. Algoritmarni grafik tasvirlash.

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

#### Misol.

#### Masalaning qo'yilishi va maqsadni aniqlash.

Ekin maydonining XOY koordinata tekisligida  $Y=0$ ,  $X=a$ ,  $X=b$  to'g'ri chiziqlar va  $Y = \sqrt{X}$  egri chiziq bilan chegaralangan yuzasi aniqlansin.

#### Masalani matematik ifodalash.

Masalaning qo'yilishidan ma'lumki, ekin maydoni yuzasi egri chizikli trapeziya shaklidir. Uning yuzasini topish aniq integral yordamida quyidagicha hisoblanadi:

$$S = \int_a^b \sqrt{x} dx$$

bu yerda:  $a$  - integralning quyi chegarasi;  $b$  - integralning yuqori chegarasi.

Misolni yechish algoritmi quyidagicha bo'ladi:

kompyuter xotirasiga  $a$  va  $b$  ning qiymatlari kiritilsin;

to'g'ri to'rtburchaklar soni  $n$  kiritilsin;

to'rtburchaklar asosi (eni) hisoblansin:  $h = (b-a)/n$

1-to'rtburchak yuzi hisoblansin:  $S_1 = \sqrt{x_1} \cdot h$ ;

$S_1$  ning qiymati eslab qolinsin;

2-to'rtburchakka o'tilsin:  $x_2 = x_1 + h$ ;

2-to'rtburchak yuzi hisoblansin:  $S_2 = \sqrt{x_2} \cdot h$ ;

$S_2$  ning qiymati  $S_1$  ning qiymatiga qo'shib qo'yilsin va yig'indi eslab qolinsin;

.....

$n$ -to'rtburchakka o'tilsin:  $x_n = x_1 + (n-1)h$

$n$ -to'rtburchak yuzi hisoblansin:  $S_n = \sqrt{x_n} \cdot h$ ;

$S_n$  ning qiymati  $S_1, S_2, \dots, S_{n-1}$  lar qiymatiga qo'shilsin;

#### Algoritmning asosiy xossalari.

Algoritm quyidagi asosiy xossalarga ega: **uzlklilik, aniqlik, natijaviylik va ommaviylik.**

**Uzlklilik.** Dastlabki berilgan malumotlarni natijaga aylantirish jarayoni uzlukli ravishda amalga oshiriladi.

**Aniqlik.** Algoritmning har bir qoidasi aniq va bir qiymatli bo'lishi zarur.

**Natijaviylik.** Algoritm masalaning yechimiga chekli sondagi qadamlarda olib kelishi yoki masalani "yechib bo'lmaydi" degan xabar bilan tugashi kerak.

**Ommaviylik.** Masalani yechish algoritmi shunday yaratilishi kerakki, uni faqat boshlang‘ich ma’lumotlar bilan farqlanadigan masalalarni yechish uchun xam qo‘llanilishi kerak.








Algoritmni bir necha xil usul bilan ifodalash mumkin. Shulardan uchtasi keng tarqalgan. Bular:

*Algoritmni so‘z bilan ifodalash;*

*Algoritmni maxsus (algoritmik) tilda yozish;*

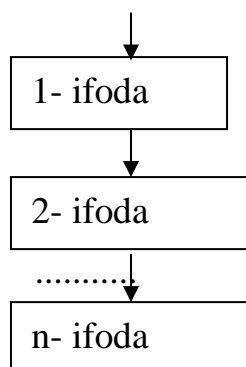
*Algoritmni grafik usul (blok-sxema) yordamida ifodalash.*

Blok- sxemada quyidagi belgilar ishlatiladi:

Shakl	Funksiyasi	Shakl	Funksiyasi
	Aloritm boshi va oxiri		Axborotni kiritish
	Hisoblash jarayoni		Natijani chop etish
	Shartini tekshirish		Takrorlanish boshi
	Boshqa algoritmgamurojaat		

Algoritmning asosiy turlariga **chiziqli**, **tarmoqlanuvchi** va **takrorlanuvchi** **algoritm**lar kiradi.

**Chiziqli algoritm**larda bloklar ketma-ket bir to‘g‘ri chiziq bo‘ylab joylashgan bo‘lib, berilgan tartibda bajariladi. Chiziqli algortmda hisoblash jarayonining quyidagicha ko‘rinishda ifodalanadi.

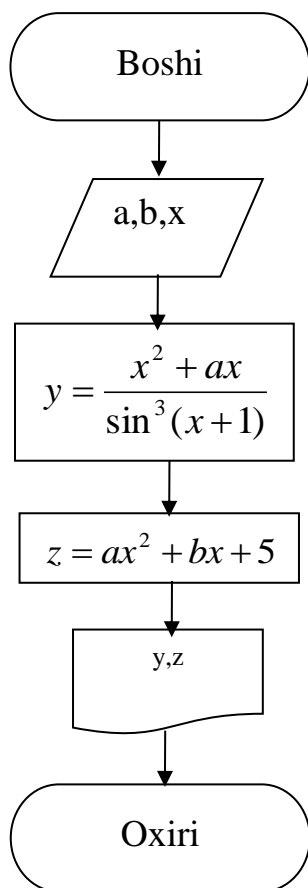


**Misol.** Quyidagi misolga blok- sxema tuzamiz:

$$y = \frac{x^2 + ax}{\sin^3(x+1)}$$

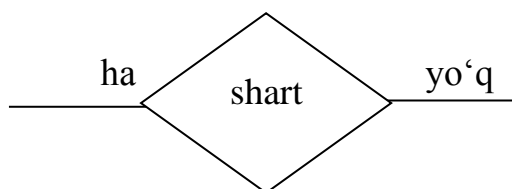
$$z = ax^2 + bx + 5$$

Bu yerda  $x=5$ ;  $a=3$ ;  $b=6$ ;



Ba'zi hollarda hisoblashlar birorta mantiqiy shartni bajarilishiga bog'liq holda u yoki bu tarmoq bo'yicha amalga oshirilishi mumkin. Bunday tuzilishdagi algoritm “**tarmoqlanuvchi algoritm**” deb ataladi.

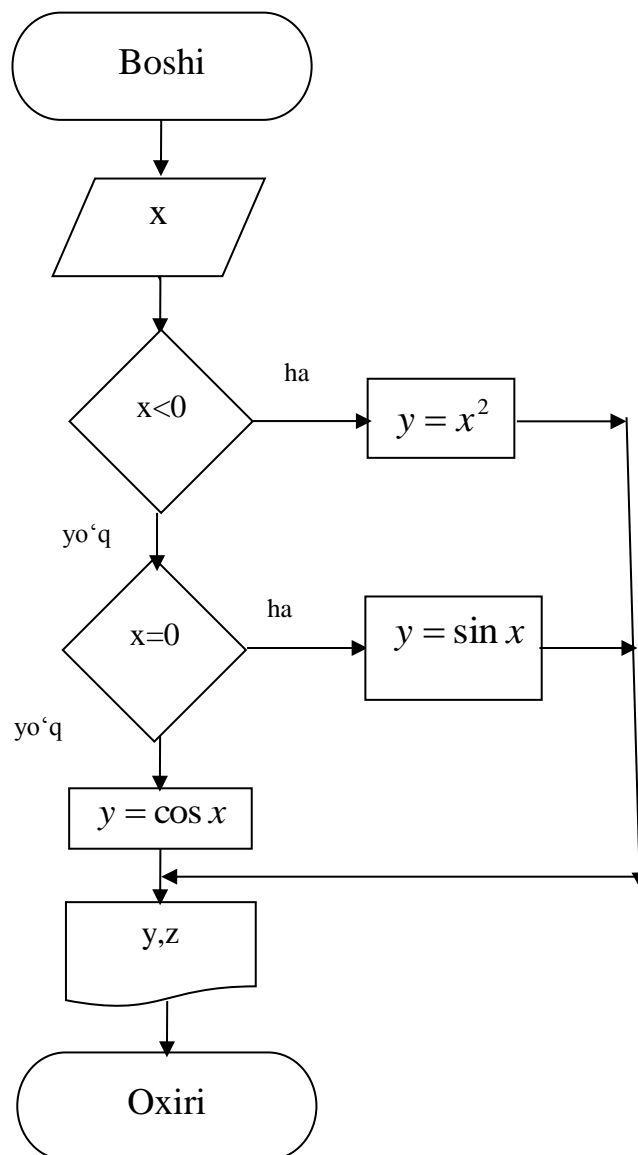
Algoritmning bu turida shartni tekshirish quyidagicha ifodalanadi:



**Misol.** Quyidagi misolga blok -sxema tuzamiz:

$$y = \begin{cases} x^2 & x > 0 \\ \sin x & x = 0 \\ \cos x & x < 0 \end{cases}$$

Bu yerda  $x=6$ .



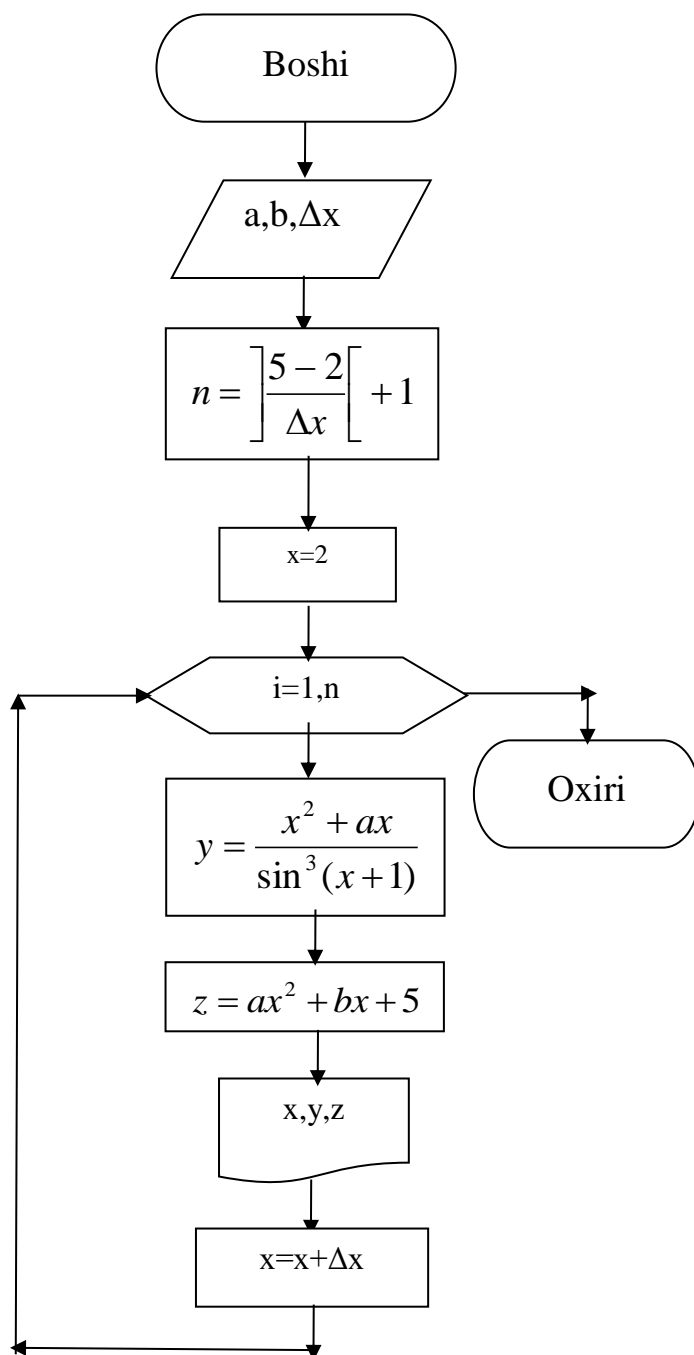
Ba'zi masalalarning yechimini hisoblashda bitta matematik ifodani o'zgaruvchilarning turli qiymatlarida ko'p marta hisoblashga to'g'ri keladi. Bunday amallar takrorlanib keladigan algoritmlar **takrorlanuvchi algoritmlar** deb ataladi. Takrorlanuvchi algoritmlar dastur hajmini va kompilyatsiya jarayonini qisqartirish imkonini beradi.

**Misol.** Quyidagi misolga blok -sxema tuzamiz:

$$y = \frac{x^2 + ax}{\sin^3(x+1)}$$

$$z = ax^2 + bx + 5$$

Bu yerda  $a=3$ ;  $b=6$ ;  $x \in [2;5]$ ;  $\Delta x = 0,5$ ;





## Laboratoriya ishi variantlari:

### 1-variant

Quyidagi masalani yechish uchun blok-sxema tuzing:

To'g'ri parallelepipedning asosining tomonlari  $a=5\text{sm}$ ,  $b=6\text{sm}$ , ular orasidagi burchak  $30^{\circ}$ , yon qirralari  $l=4\text{ sm}$  bo'lsa, parallelepipedning to'la sirti va hajmini hisoblang.

### 2-variant

Quyidagi masalani yechish uchun blok-sxema tuzing:

Piramidaning asosi teng yonli uchburchak bo'lib, ushbu uchburchakning asosi  $a=6\text{ sm}$ , balandligi  $h=9\text{ sm}$ . Piramidaning yon qirralari  $l=13\text{ sm}$  ga teng. Piramidaning to'la sirti yuzini hisoblang.

### 3-variant

Quyidagi masalani yechish uchun blok-sxema tuzing:

Silindr asosining radiusi  $r = 22\text{ sm}$ , balandligi  $h = 7\text{ sm}$  bo'lsa, silindr to'la sirti va hajmini hisoblang.

### 4-variant

Quyidagi masalani yechish uchun blok-sxema tuzing:

Konus asosining radiusi  $r = 10\text{ sm}$ , balandligi  $h = 7\text{ sm}$  bo'lsa, silindr to'la sirti va hajmini hisoblang.

### 5-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

To'g'ri burchakli uchburchakning katetlari  $15$  va  $20\text{ sm}$ . Gipotenuza o'tkazilgan balandligini toping.

### 6-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Parallelogramda diagonallar orasidagi burchak  $\alpha=120^{\circ}$ , diagonallari esa  $d_1=60\text{ sm}$  va  $d_2=32\text{ sm}$  bo'lsa, uning kichik tomonini toping.

### 7-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Parallelogrammning katta tomoniga tushirilgan balandligi  $h=24$  sm bo'lib, uni  $a_1=7$  sm va  $a_2=32$  sm bo'lgan kesmalarga ajratadi. Parallelogrammning kichik diagonalini va perimetrini toping.

### 8-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Trapetsiyaning asoslari  $a=20$  sm va  $b=60$  sm, yon tomonlari  $c=13$  sm va  $f=37$  sm bo'lsa, uning balandligini toping.

### 9-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Aylana nuqtasidan diametrga perpendikulyar o'tkazilgan bo'lib, uni  $d_1=16$  sm va  $d_2=9$  sm li kesmalarga ajratsa, perpendikulyar uzunligini toping.

### 10-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Uchburchakning tomonlari  $a=15$  sm,  $b=26$  sm va  $c=37$  sm bo'lsa, unga ichki chizilgan aylana radiusini toping.

### 11-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Agar uchburchak asosi  $a>5$  bo'lsa, uchburchak yuzi  $s=20$  ga, aks holda  $s=10$  ga teng.

### 12-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Ikkita  $R_1=4$  va  $R_2=6$  qarshiliklar uchun  $x=1$  bo'lganda, umumiy qarshilik  $R = R_1 + R$  formula bo'yicha, aks holda  $R = \frac{R_1+R_2}{R_1 \cdot R_2}$  formula bo'yicha hisoblanadi.

### 13-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Koeffitsiyentlari  $a=4$ ,  $b=5$ ,  $c=2$  bo'lgan kvadrat tenglama uchun  $d>0$  bo'lganda ildizlar soni  $n=2$  ta,  $d=0$  bo'lganda ildizlar soni  $n=1$  ta,  $d<0$  bo'lganda ildizlar soni  $n=0$  ta bo'ladi.

#### 14-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Teng tomonli uchburhak tomoni  $a=10$  sm bo'lsa, shu uchberchak yuzi va perimetrini hisodlang.

#### 15-variant

Quyidagi mulohazani ifodalash uchun blok-sxema tuzing:

$x$  va  $y$  haqiqiy sonlar berilgan. Agar  $x>0, y>0$  bo'lsa,  $ch=1$ ga teng; agar  $x<0, y>0$  bo'lsa,  $ch=2$ ga teng; agar  $x<0, y<0$  bo'lsa,  $ch=3$ ga teng; agar  $x>0, y<0$  bo'lsa,  $ch=4$ ga teng.

#### 16-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

$F_t$ - traktorning tortish kuchi,  $F_i$ -yerning traktor bilan ishqalanish kuchi berilgan. Agar  $F_t>F_i$  bo'lsa, najjaviy kuch qiymati  $F=F_t-F_i$ ; agar  $F_t=F_i$  bo'lsa,  $F=0$ ; agar  $F_t<F_i$  bo'lsa,  $F=F_i-F_t$  bo'ladi.

#### 17-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

$F_1=10$  n,  $F_2=8$  n kuchlar va ular orasidagi birchak  $a$  berilgan. Agar  $a=90^\circ$  bo'lsa, teng ta'sir etuvchi kuch  $F = \sqrt{F_1^2 + F_2^2}$  ga, aks holda  $F = \sqrt{F_1^2 + F_2^2 - 2F_1 \cdot F_2 \cdot \cos(180^\circ - a)}$  ga teng bo'ladi.

#### 18-variant.

Quyidagi misolni yechish uchun blok-sxema tuzing:

Harakatlanayotgan jism uchun harakat radiusi  $R$ , burchk tezlik  $\omega = 5$  rad/s, tezlanish  $a=2$  m/s<sup>2</sup>, harakatlanish vaqti  $t=10$  s berilgan. Agar  $R=0$  bo'lsa, chiziqli tezlik  $v = a \cdot t$  ga, aks holda  $v = \omega \cdot R$  ga teng bo'ladi.

#### 19-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Matematik mayatnik uzunligi  $l=0,5$  m, chastotasi  $\nu=10$  c<sup>-1</sup>, erkin tushish tezlanishi  $g=9,81$  m/s<sup>2</sup> va  $x$  o`zgaruvchi qymati berilgan.  $x>0$  bo'lsa, tebianish davri  $T = 2\pi \sqrt{\frac{l}{g}}$  ga, aks holda  $T = \frac{1}{\nu}$  ga teng bo'ladi.

## 20-variant

Quyidagi misolni yechish uchun blok-sxema tuzing:

Avtomobil dvigatelingning vali  $\omega=180$  rad/s burchak tezlik bilan aylanadi. Agar dvigatel vali shkivining diametri  $d_{sh}=9$  sm, ventilyatorniki esa  $d_v=6$  sm bo`lsa, tasmaning chiziqli tezligini ( $v = \omega_1.R_1$ ) va avtomobil tasmalarning burchak tezligini ( $\omega_2 = \frac{v}{R_1}$ ) aniqlang.

## 2-laboratoriya ishi.

**Mavzu: Chiziqli algoritmlarga dasturlash. Muhandislik va suv xo‘jaligi mexanizatsiyasi masalalarida ma’lumotlarni kiritish-chiqarish operatorlaridan foydalanib dastur tuzish.**

C++ alfavitiga quyidagi belgilar kiradi:

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

Izohlarda, satrlarda va belgili o‘zgaruvchilarda boshqa belgilar, masalan rus xarflarini ishlatilishi mumkin.

**Identifikator.** Identifikatorlar lotin xarflari, ostki chiziq belgisi va sonlar ketma-ketligidan iborat bo‘ladi. Identifikator lotin xarfidan yoki ostki chiziq belgisidan boshlanishi lozim.

Misol uchun:

*A1, \_MAX, adress\_01, RIM, rim.*

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

**Xizmatchi so‘zlar.** Dsturchi tomonidan o‘zgaruvchilar nomlari sifatida ishlatish mumkin bo‘lmagan identifikatorlar xizmatchi so‘zlar deyiladi.

C++ tilida quyidagi xizmatchi so‘zlar mavjud:

*int, extern, else, char, register, for, float, type, def, do, double, static, while, struct, goto, switch, union, return, case, long, sizeof, default, short, break, entry, unsigned, continue, auto, if* va boshqalar.

**O‘zgaruvchilar ob’ekt sifatida.** C++ tilining asosiy tushunchalaridan biri nomlangan xotira qismi – ob’ekt tushunchasidir. Ob’ektning xususiy holi bu

o'zgaruvchidir. O'zgaruvchiga qiymat berilganda unga ajratilgan xotira qismiga shu qiymat kodi yoziladi. O'zgaruvchi qiymatiga nomi orqali murojaat qilish mumkin, xotira qismiga esa faqat manzili orqali murojaat qilinadi. O'zgaruvchi nomi bu identifikatoridir. O'zgaruvchi nomi sifatida xizmatchi so'zlarni ishlatish mumkin emas.

**O'zgaruvchilar turlari.** O'zgaruvchilarning qo'yidagi tiplari mavjuddir:

**char** – bitta belgi;

**long char** – uzun belgi;

**int** – butun son;

**short**– qisqa butun son;

**long**– uzun butun son;

**float** - haqiqiy son;

**double** – ikkilangan haqiqiy son;

**long double** – uzun ikkilangan haqiqiy son;

O'zgaruvchilarni dasturning ixtiyoriy qismida ta'riflash yoki qayta ta'riflash mumkin.

Misol uchun:

***Int a, b1, ac; eki***

***Int a;***

***int b1;***

***int ac;***

O'zgaruvchilar ta'riflanganda ularning qiymatlari aniqlanmagan bo'ladi. Lekin o'zgaruvchilarni ta'riflashda initsializatsiyalash ya'ni boshlang'ich qiymatlarini berish mumkin.

Misol uchun:

***Int I=0;***

***Char c='k';***

**O'zgarmas-** bu dastur ishlashi davomida qiymati o'zgarmaydigan kattalikdir. C++ tilida besh turdagi o'zgarmaslar ishlatilishi mumkin: butun , haqiqiy , belgi, sanovchi o'zgarmaslar va nul ko'rsatkich. Butun sonlar o'nlik, sakkizlik yoki o'n oltilik sanoq tizimlarida berilishi mumkin. O'nlik sanoq tizimida butun sonlar 0-9 raqamlari ketma-ketligidan iborat bo'lib, birinchi raqami 0 bo'ishi kerak emas. Sakkizlik sanoq tizimida butun sonlar 0 bilan boshlanuvchi 0-7 raqamlaridan iborat ketma-ketlikdir. O'n oltilik sanoq tizimida butun son 0x yoki 0X bilan boshlanuvchi 0-9 raqamlari va a-f yoki A-F harflaridan iborat ketma- ketlikdir. Masalan, 15 va 22 o'nlik sonlari sakkizlikda 017 va 026, o'n oltilikda 0xF va 0x16 shaklida tasvirlanadi. Ma'lumotlarning haqiqiy son turi olti qismdan iborat bulishi mumkin: butun qism, nuqta, kasr qism, yoki E belgisi, o'nlik daraja, F eki f

suffikslari. Masalan :  $66.$ ;  $.0$ ;  $.12$ ;  $3.14F$ ;  $1.12e-12$ . Ma'lumolarning uzun haqiqiy son turi: Oxiriga L yoki l suffikslari qo'yilgan haqiqiy son. Masalan:  $2E+6L$ ;

### Belgili o'zgarmas.

Belgili o'zgarmas apostrof ichiga olingan bitta yoki ikkita belgidan iborat bo'ladi. Misol uchun 'x', '\*', '\012', '\0', '\n' - bitta belgili o'zgarmas; 'dd', '\n\t', '\x07\x07' ikki belgili o'zgarmaslar.

Yozilishi	Ichki kodi	Belgii (nomi)	Ma'nosi
\a	0x07	bel (audible bell)	Tovush signali
\b	0x08	Bs (bascpase)	Bir qadam qaytish
\f	0x0C	Ff (form feed)	Sahifani qaytarish
\n	0x0A	lf (line feed)	Qatorni o'tkazish
\r	0x0D	Cr (carriage return)	Karetkani qaytarish
\t	0x09	Ht (horizontal tab)	Gorizontol tabulyatsi
\v	0x0B	Vt (vertical tab)	Vertikal tabulyatsi
\\	0x5C	\ (bacslash)	Teskari chiziq
\'	0x27	' (single out)	Apostrof
\"	0x22	" (double quote)	Qo'shtirnoq
\?	0x3F	? (question mark)	Savol belgisi

### Satrlı o'zgarmas.

Satrlı o'zgarmaslar C++ tili o'zgarmaslariga kirmaydi, balki alohida tur hisoblanadi. Satrlı o'zgarmas bu qo'shtirnoqqa olingan ixtiyoriy belgilar ketma-ketligidir. Misol uchun "Men talabaman".

Satrlar orasiga eskeyp belgilar ham kirishi mumkin. Bu belgilar oldiga \ belgisi qo'yiladi. Misol uchun : "\n". Satr belgilari xotirada ketma-ket joylashtiriladi va har bir satrlı o'zgarmas oxiriga avtomatik ravishda kompilyator tomonidan '\0' belgii qo'shiladi. Shunday satrning xotiradagi xajmi belgilar soni+1 baytga tengdir. Ketma-ket kelgan va bo'shlik, tabulyatsiya yoki satr oxiri belgisi bilan ajratilgan satrlar kompilyatsiya davrida bitta satrga aylantiriladi. Misol uchun: "Salom" "Toshkent" satrlari bitta satr deb qaraladi: "Salom Toshkent". Bu qoidaga bir necha qatorga yozilgan satrlar ham bo'ysinadi. Misol uchun :

**"O'zbekistonga "**

**"bahor "**

**"keldi"**

qatorlari bitta qatorga joylashadi: "O'zbekistonga bahor keldi".

**Arifmetik amallar.** C++ dagi amallar quyidagi jadvalda berilgan. Ular ikkita operand bilan ishlatildi.

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 xususiyatlarini ko'rib chiqaylik. Butun sonli bo'lishda, yani bo'luvchi ham, bo'linuvchi ham butun son bo'lganda, javob butun son bo'ladi. Javob yaxlitlanmaydi, kasr qismi tashlanib yuborilib, butun qismining o'zi qoladi.  $x\%y$  ifodasi  $x$  ni  $y$  ga bo'lgandan keyin chiqadigan qoldiqni beradi. Demak,  $7\%4$  ifoda natijasi 3 bo'ladi. % operatori faqat butun sonlar bilan ishlaydi. Vergulli (real) sonlar bilan ishlash uchun "math.h" kutubxonasiidagi fmod funksiyasini qo'llash kerak. C++ da qavslarning ma'nosi huddi matematikadagidekdir. Undan tashqari boshqa algebraik ifodalarning ketma-ketligi ham odatdagidek. Oldin ko'paytirish, bo'lish va modul olish operatorlari bajariladi. Agar bir necha teng kuchli amallar ketma-ket kelsa, ular chapdan boshlab o'nga bajariladi. Yuqoridagi amallardan keyin qo'shish va ayirish bajariladi. Misol uchun,  $k = m * 5 + 7 \% n / (9 + x)$ ; Birinchi bo'lib  $m * 5$  hisoblanadi. Keyin  $7 \% n$  bajariladi va qoldiq  $(9 + x)$  ga bo'linadi. Chiqqan javob esa  $m * 5$  ning javobiga qo'shiladi.

### Amallar jadvali

Arifmetik amallar	Razryadli amallar	Nisbat amallari	Mantiqiy amallar
+ qo'shish	& va	= = teng	&& va
- ayirish	yoki	!= teng emas	yoki
* ko'paytirish	^ yoki inkori	> katta	! inkor
/ bo'lish	<< chapga surish	>= katta yoki teng	
% modul olish	>> o'ngga surish	< kichik	
- unar minus	~ inkor	<= kichik yoki teng	
+ unar plyus			
++ dekrement			
-- inkrement			

Amallar odatda unar, ya'ni bitta operandga qo'llaniladigan amallarga va binary, ya'ni ikki operandga qo'llaniladigan amallarga ajratiladi. Binar amallar additiv ya'ni + qo'shuv va - ayirish amallariga, hamda multiplikativ ya'ni \* ko'paytirish, / bo'lish va % modul olish amallariga ajratiladi. Additiv

amallarining ustivorligi multiplikativ amallarining ustivorligidan pastroqdir. Butun sonni butun songa bo'lganda natija butun songacha yahlitlanadi. Misol uchun  $20/3=6$ ;  $(-20)/3=-6$ ;  $20/(-3)=-6$ .

Binar arifmetik amallar bajarilganda turlarni keltirish quyidagi qoidalar asosida amalga oshiriladi: short va char turlari int turiga keltiriladi;

Agar operandlar biri long turiga tegishli bo'lsa, ikkinchi operand ham long turiga keltiriladi va natija ham long turiga tegishli bo'ladi;

Agar operandlar biri float turiga tegishli bo'lsa, ikkinchi operand ham float turiga keltiriladi va natija ham float turiga tegishli bo'ladi;

Agar operandlar biri double turiga tegishli bo'lsa ikkinchi operand ham double turiga keltiriladi va natija ham double turiga tegishli bo'ladi;

Agar operandlar biri long double turiga tegishli bo'lsa ikkinchi operand ham long double turiga keltiriladi va natija ham long double turiga tegishli bo'ladi;

Unar amallarga ishorani o'zgartiruvchi unar minus – va unar + amallari kiradi. Bundan tashqari ++ va -- amallari ham unar amallarga kiradi.

++ unar amali qiymatni 1 ga oshirishni ko'rsatadi. Amalni prefiks ya'ni ++i ko'rinishda ishlatish oldin o'zgaruvchi qiymatini oshirib so'ngra foydalanish lozimligini, postfiks, ya'ni i++ ko'rinishda ishlatish oldin o'zgaruvchi qiymatidan foydalanib, so'ngra oshirish kerakligini ko'rsatadi. Misol uchun i qiymati 2 ga teng bo'lsin, u holda  $3+(++i)$  ifoda qiymati 6 ga,  $3+i++$  ifoda qiymati 5 ga teng bo'ladi. Ikkala holda ham i qiymati 3 ga teng bo'ladi. -- unar amali qiymatni 1 ga kamaytirishni ko'rsatadi. Bu amal ham prefiks va postfiks ko'rinishda ishlatilishi mumkin. Bu ikki amalni faqat o'zgaruvchilarga qo'llash mumkindir. Unar amallarining ustivorligi binar amallardan yuqoridir.

**Razryadli amallari.** Razryadli amallar natijasi butun sonlarni ikkilik ko'rinishlarining har bir razryadiga mos mantiqiy amallarni qo'llashdan hosil bo'ladi. Masalan, 5 kodi 101 ga teng va 6 kodi 110 ga teng.  $6\&5$  qiymati 4 ga ya'ni 100 ga teng.  $6|5$  qiymati 7 ga ya'ni 111 ga teng.

$6^5$  qiymati 3 ga ya'ni 011 ga teng.

Bu misollarda amallar ustivorligi oshib borishi tartibida berilgandir.

Bu amallardan tashqari  $M\ll N$  chapga razryadli siljitish va  $M\gg N$  o'ngga razryadli siljitish amallari qo'llaniladi. Siljitish M butun sonning razryadli ko'rinishiga qo'llaniladi. N nechta pozitsiyaga siljitish kerakligini ko'rsatadi.

Chapga N pozitsiyaga surish bu operand qiymatini ikkining N chi darajasiga ko'paytirishga mos keladi. Misol uchun  $5\ll 2=20$ . Bu amalning bitli ko'rinishi:  **$101\ll 2=10100$** .

Agar operand musbat bo'lsa N pozitsiyaga o'ngga surish chap operandni ikkining N darajasiga bo'lib, kasr qismini tashlab yuborishga mosdir. Misol uchun



$5 >> 2 = 1$ . Bu amalning bitli kurinishi  $101 >> 2 = 001 = 1$ . Agarda operand qiymati manfiy bo'lsa ikki variant mavjuddir: arifmetik siljitishda bo'shatilayotgan razryadlar ishora razryadi qiymati bilan to'ldiriladi, mantiqiy siljitishda bo'shatilayotgan razryadlar nullar bilan tuldiriladi. Razryadli surish amallarining ustivorligi o'zaro teng, razryadli inkor amalidan past, qolgan razryadli amallardan yuqoridir. Razryadli inkor amali unar, qolgan amallar binar amallarga kiradi.

**Munosabat amallari.** Munosabat amallari qiymatlari 1 ga teng agar Munosabat bajarilsa va aksincha 0 ga tengdir. Munosabat amallari arifmetik tipdagi operandlarga yoki ko'rsatkichlarga qo'llaniladi.

Misollar:

$1 != 0$  qiymati 1 ga teng;

$1 == 0$  qiymati 0 ga teng;

$3 >= 3$  qiymati 1 ga teng;

$3 > 3$  qiymati 0 ga teng;

$2 <= 2$  qiymati 1 ga teng;

$2 < 2$  qiymati 0 ga teng;

Katta  $>$ , kichik  $<$ , katta yoki teng  $>=$ , kichik yoki teng  $<=$  amallarining ustivorligi bir hildir. Teng  $==$  va teng emas  $!=$  amallarining ustivorligi uzaro teng va qolgan amallardan pastdir.

**Mantiqiy amallar.** Bu amallarning natijalari quyidagicha aniqlanadi:  $(x, y)$  nuqta koordinatalar tekisligining 1-choragida joylashganligini tekshirmoqchimiz.  $x || y$  amali 1 ga teng agar  $x > 0$  va  $y > 0$  shartning kamida bittasi bajarilsa, aksincha 0 ga teng.  $x \&\& y$  amali 1 ga teng agar  $x > 0$  va  $y > 0$  bo'lsa, aksincha 0 ga teng  $!x$  amali 0 ga teng agar  $x > 0$  bo'lsa, aksincha 1 ga teng Bu misollarda amallar ustivorligi oshib borish tartibida berilgandir.

Inkor  $!$  amali unar, qolganlari binar amaldir.

Bu amallardan tashqari quyidagi amallar ham mavjuddir:

**Qiymat berish amali.** Qiymat berish amali = binar amal bo'lib, chap operand odatda o'zgaruvchi, o'ng operandi odatda ifoda bo'ladi. Misol uchun  $Z = 4.7 + 3.34$  Bu qiymati 8.04 ga teng ifodadir. Bu qiymat  $Z$  o'zgaruvchiga ham beriladi. Bu ifoda oxiriga nuqta vergul ; belgisi qo'yilganda operatorga aylanadi.  **$Z = 4.7 + 3.34$** ; Bitta ifodada bir necha qiymat berish amallari qo'llanilishi mumkin. Misol uchun:  **$C = y = f = 4.2 + 2.8$** ; Bundan tashqari  $C ++$  tili da murakkab qiymat berish amali mavjud bo'lib, umumiy ko'rinishi quyidagichadir: O'zgaruvchi\_nomi amal= ifoda; Bu yerda amal quyidagi amallardan biri  $*$ ,  $/$ ,  $\%$ ,  $+$ ,  $-$ ,  $\&$ ,  $\wedge$ ,  $|$ ,  $<<$ ,  $>>$ . Misol uchun:

$X+=4$	ifoda	$x=x+4$	ifodaga	ekvivalentdir;
$X*=a$	ifoda	$x=x*a$	ifodaga	ekvivalentdir;
$X/=a+b$	ifoda	$x=x/(a+b)$	ifodaga	ekvivalentdir;
$X>>=4$	ifoda	$x=x>>4$	ifodaga	ekvivalentdir;

**Shartli amal.** Shartli amal ternar amal deyiladi va uchta operanddan iborat bo'ladi:

$\langle 1\text{-ifoda} \rangle ? \langle 2\text{-ifoda} \rangle : \langle 3\text{-ifoda} \rangle$

Shartli amal bajarilganda avval 1- ifoda hisoblanadi. Agar 1-ifoda qiymati 0 dan farqli bo'lsa 2- ifoda hisoblanadi va qiymati natija sifatida qabul qilinadi, aks holda 3-ifoda hisoblanadi va qiymati natija sifatida qabul qilinadi. Misol uchun modulni hisoblash:  $x < 0 ? -x : x$  yoki ikkita son kichigini hisoblash  $a < b ? a : b$ .

Shuni aytish lozimki shartli ifodadan har qanday ifoda sifatida foydalanish mumkin. Agar F float turga, N –int turga tegishli bo'lsa,  $(N > 0) ? F : N$  ifoda N musbat yoki manfiyligidan qat'iy nazar double turiga tegishli bo'ladi. Shartli ifodada birinchi ifodani qavsga olish shart emas. Dastur preprocessor komandalari va bir necha funktsiyalardan iborat bo'lishi mumkin. Bu funktsiyalar orasida **main** nomli asosiy funktsiya bo'lishi shart.

Quyidagi dastur yozaylik:

```
#include <iostream>
using namespace std;
void main ()
{
    cout << "Salom \ n";
}
```

**#include** buyrug'i kompilyatorga *iostream* faylidagi standart kirish va chiqish oqimlaridan foydalanishni buyuradi. Ushbu tavsiflarsiz, **cout** << "Salom \ n" amali bajarilmaydi. **using namespace std**- std nomlar fazosodan foydalanishni bildiradi. Ushbu buyruq berilmasa ma'lumotlarni kiritish va chiqarish operatorlarida *std* so'zini yozishga to'g'ri keladi. **void main** () –dasturdagi main nomli asosiy funktsiya bo'lib, dasturda albatta yozilishi kerak. Har bir dasturda *main* deb nomlangan funktsiya bo'lishi kerak va dastur shu funktsiya bilan boshlanadi. **void**- *main* funktsiyasining turini bildirib, funktsiyaning qiymat qaytarmasligini bildiradi. **cout** << "Salom \ n" ma'lumotlarni chop etish operatori bo'lib, Salom so'zini chop etadi. \n belgisi yangi qatorga o'tishni bildiradi.

C ++ tilida yozilgan dastur natijasini olish uchun uni kompilyatsiya qilish kerak. Dastur matni o'qiladi va tahlil qilinadi, agar biron bir xato aniqlanmasa, dasturda ishlatilgan nomlar va amalar tekshiriladi (bizning holatlarimizda bu cout va <<). Iloji bo'lsa, dastur kutubxonadagi yetishmayotgan ta'riflarni to'ldiriladi. Bizning holatlarimizda cout va << oqimlari stream.h da tasvirlangan, ya'ni ularning turlari ko'rsatilgan, ammo ularning amalga oshirilishiga oid tafsilotlar berilmagan.

Quyidagi dastur uzunlikni dyuymlarda kiritishni so'raydi. Buni amalga oshirgandan so'ng, u dyuymni santimetrda ifodalab chop etadi.

```
#include <iostream>
using namespace std;
int main()
{
    int inch = 0; // inch - дюйм
    cout << "dyum";
    cin >> inch;
    cout << inch;
    cout << " in = ";
    cout << inch*2.54;
    cout << " sm\n";
    return 0;
}
```

*main()* funktsiyasining birinchi satri *inch* butun o'zgaruvchini aniqlaydi. Uning qiymati *cin* standart ma'lumotlarni kiritish oqimining >> amali yordamida o'qiladi. Dastur ishga tushgandan keyin natija quyidagicha ko'rinishi mumkin:

*dyum*

*12*

*in = 30,48 sm*

Oxirgi to'rtta chop etish buyrug'ini bitta operator bilan yozilishi mumkin:

```
cout << dyuym << "in =" << dyuym * 2.54 << "sm \ n";
```

**Sharxlar.** Dasturda faqat dasturni sharxlash uchun ishlatiladigan, kompilyator hisoblamayigan matnlar sharx deyiladi. C ++ da sharx ikki xil usulda yoziladi. Birinchi usulda sharx /\* belgilari bilan boshlanib, \*/ belgilari bilan tugaydi. Sharxning by turi ko'p satrli sharhlar uchun ishlatiladi. // belgilar bitta satrli sharxlar uchun ishlatiladi.

Har bir nom o'z turiga ega. Masalan, *int dyuym;*

Bu yerda **dyuym** turi **int** ekanligini, ya'ni dyuym butun qiymatli o'zgaruvchi ekanligini aniqlaydi. Tavsif - bu nomni dasturga kirituvchi operatoridir..

Asosiy turlari quyidagilardir:

*byte , short, int, long, float, double, boolean, char*

Dastlabki to'rtta tur butun sonlarni ifodalash uchun ishlatiladi, keyingii ikkitasi haqiqiy sonlarini ifodalash uchun ishlatiladi. **Char** tipi bitta belgini ifodalash uchun ishlatiladi, **boolean** esa mantiqiy turni aniqlaydi. Tur bilan ifodalanishi mumkin bo'lgan butun sonlarning diapazoni uning o'lchamiga bog'liq. Asosiy turlarning o'zaro bog'liqligi quyidagicha yozilishi mumkin:

$1 = \text{sizeof(char)} \leq \text{sizeof(short)} \leq \text{sizeof(int)} \leq \text{sizeof(long)}$   
 $\text{sizeof(float)} \leq \text{sizeof(double)}$

Apostrif ichida joylashgan belgi belgili kattalikdir.

Arifmetik amallar ushbu turlarning har qanday kombinatsiyasiga qo'llanilishi mumkin:

+ (*qo'shish, unar va binar*)

- (*ayirish, unar va binar*)

\* (*ko'paytirish*)

/ (*bo'lish*)

Shuningdek taqqoslash amallari:

== (*teng*)

!= (*teng emas*)

< (*kichik*)

> (*katta*)

<= (*kichik yoki teng*)

>= (*katta yoki teng*)

Butun sonli bo'linma butun natijani beradi:  $7/2= 3$ . Butun sonlar ustida qoldiqni hisoblash % amalini bajarish mumkin:  $7\% 2 = 1$ .

C ++ da juda ko'p amallar mavjud. Masalan:

~ (*qo'shimcha*)

& (*Va*)

^ (*eksklyuziv yoki*)

/ (*yoki*)

<< (*chapga siljish*)

>> (*o'ngga siljish*)

Ushbu amalar butun sonlarga nisbatan qo'llaniladi.

Amalning natijasi operandlar soniga bog'liq; & unar amali- manzlni qabul qilish operatsiyasi; & binar amali- mantiqiy qo'shishni bildiradi. Amalning natijasi operandlar turiga ham bog'liq:  $a + b$  ifodasida + amali, agar operandlar turi float bo'lsa, amal natijasi haqiqiy son bo'ladi, agar ular butun turga tegishli bo'lsa, amal natijasi butun son bo'ladi. C ++ da o'zlashtirish operatori mavjud =.

Masalan,  $a = b = c$  ifodada  $c$  ning qiymatini  $b$  o'zlashtiradi, so'ng  $a$  o'zlashtiradi.  $x * = 4$  amali  $x = x * 4$  amali bilan teng kuchli.

Ko'pincha C ++ dasturi ko'rsatkichlardan keng foydalanadi. \* unar amali bo'lib, \* p - p bilan ko'rsatilgan ob'ektni bildiradi. Ushbu amal bilvosita adreslash deb ham ataladi. Masalan, char \* p bo'lsa, u holda p- p bilan ko'rsatilgan belgidir. Ko'pincha ko'rsatkichlar bilan ishlashda ++ va -- amallari foydalidir.

C ++ da ba'zi bir keng tarqalgan matematik funksiyalar <cmath> faylida aniqlanadi. Masalan, ildizni topish, darajaga ko'tarish, sin (), cos () va boshqalar.

1-jadval - C ++ tilidagi matematik funksiyalar

Функция	Тavsifi	Misol
<b>abs( a )</b>	<b>a</b> ning moduli yoki absolyut qiymati	abs(-3.0)=3.0 abs(5.0)= 5.0
<b>sqrt(a)</b>	nomanfiy <b>a</b> kattalikdan olingan kvadrat ildiz	sqrt(9.0)=3.0
<b>pow(a, b)</b>	<b>a</b> ni <b>b</b> darajaga oshirish	pow(2,3)=8
<b>ceil( a )</b>	<b>a</b> ni <b>a</b> dan kichik bo'lmagan eng kichik butun songacha yaxlitlash	ceil(2.3)=3.0 ceil(-2.3)=-2.0
<b>floor(a)</b>	<b>a</b> ni <b>a</b> dan katta bo'lmagan eng katta butun songacha yaxlitlash	floor(12.4)=12 floor(-2.9)=-3
<b>fmod(a, b)</b>	a/b ifoda qoldig'ini hisoblash	fmod(4.4, 7.5) = 4.4 fmod( 7.5, 4.4) = 3.1
<b>exp(a)</b>	$e^a$ exponentani hisoblash	exp(0)=1
<b>sin(a)</b>	<b>a</b> ning sinusini radianlarda hisoblash	
<b>cos(a)</b>	<b>a</b> ning kosinusini radianlarda hisoblash	
<b>log(a)</b>	<b>a</b> ning natural logarifmini hisoblash	log(1.0)=0.0
<b>log10(a)</b>	<b>a</b> ning o'nli logarifmini hisoblash	log10(10)=1
<b>asin(a)</b>	<b>a</b> ning arksinusini hisoblash, $-1.0 < a < 1.0$	asin(1)=1.5708

Функция	Tavsifi	Misol
<b>acos(a)</b>	<b>a</b> ning arkkosinusini hisoblash, <b>-1.0 &lt; a &lt; 1.0</b>	
<b>atan(a)</b>	<b>a</b> ning arktangensini hisoblash	
<b>sinh(a)</b>	<b>a</b> ning giperboliksinusini hisoblash	
<b>cosh(a)</b>	<b>a</b> ning giperbolikkosinusini hisoblash	
<b>tan(a)</b>	<b>a</b> ning tangensini hisoblash	
<b>tanh(a)</b>	<b>a</b> ning giperboliktangensini hisoblash	

Ushbu funktsiyalarning operandalari har doim haqiqiy qiymatni qabul qilishi kerak.

Chiziqli algoritmni dasturlashga misol keltiramiz.

To'g'ri to'rtburchakning tomonlari  $a=4$ ,  $b=6$  berilgan. To'g'ri to'rtburchakning perimetri  $p$  va yuzi  $s$  ni hisoblash dasturini tuzing.

```
#include <iostream>
using namespace std;
int main( )
{
double a,b,p,s;
cout<< "a,b larning qiymatlarini kiriting:\n";
cin>>a>>b;
p=2(a+b);
s=a*b;
cout <<"p="<<p<<"s="<<s<<endl;
return 0;
}
```

### Laboratoriya ishi variantlari:

#### 1-variant

Quyidagi masalani yechish uchun dastur tuzing:

To'g'ri parallelepipedning asosining tomonlari  $a=5$  sm,  $b=6$  sm, ular orasidagi burchak  $\alpha=30^0$ , yon qirralari  $l=4$  sm bo'lsa, parallelepipedning to'la sirti va hajmini hisoblang.

### **2-variant**

Quyidagi masalani yechish uchun dastur tuzing:

Piramidaning asosi teng yonli uchburchak bo'lib, ushbu uchburchakning asosi  $a=6$  sm, balandligi  $= 9$  sm. Piramidaning yon qirralari  $l=13$  sm ga teng. Piramidaning to'la sirti yuzini hisoblang.

### **3-variant**

Quyidagi masalani yechish uchun dastur tuzing:

Silindr asosining radiusi  $r =22$  sm, balandligi  $h = 7$  sm bo'lsa, silindr to'la sirti va hajmini hisoblang.

### **4-variant**

Quyidagi masalani yechish uchun dastur tuzing:

Konus asosining radiusi  $r =10$  sm, balandligi  $h = 7$  sm bo'lsa, silindr to'la sirti va hajmini hisoblang.

### **5-variant**

Quyidagi masalani yechish uchun dastur tuzing:

To'g'ri burchakli uchburchakning katetlari  $a=15$  sm va  $b=20$  sm. Uchburchakning gipotenuzaga o'tkazilgan balandligini toping.

### **6-variant**

Quyidagi masalani yechish uchun dastur tuzing:

Parallelogramda diagonallar orasidagi burchak  $\alpha=120^\circ$ , diagonallari esa  $d_1=60$  sm va  $d_2=32$  sm bo'lsa, uning kichik tomonini toping.

### **7-variant**

Quyidagi masalani yechish uchun dastur tuzing:

Parallelogramning katta tomoniga tushirilgan balandligi  $h=24$  sm bo'lib, uning  $x=7$  sm va  $y=32$  sm bo'lgan kesmalarga ajratadi. Parallelogramning kichik diagonalini va perimetrini toping.

### **8-variant**

Quyidagi masalani yechish uchun dastur tuzing:

Trapetsiyaning asoslari  $b=20$  sm va  $a=60$  sm, yon tomonlari  $c=13$  sm va  $f=37$  sm bo'lsa, uning balandligini toping.

### 9-variant

Quyidagi masalani yechish uchun dastur tuzing:

Aylana nuqtasidan diametrga perpendikulyar o'tkazilgan bo'lib, diametrni  $x=16$  sm va  $y=9$  sm li kesmalarga ajratsa, perpendikulyar uzunligini toping.

### 10-variant

Quyidagi masalani yechish uchun dastur tuzing:

Uchburchakning tomonlari  $a=15$  sm,  $b=26$  sm va  $c=37$  sm bo'lsa, uchburchakka ichki chizilgan aylana radiusini toping.

### 11-variant

Quyidagi masalani yechish uchun dastur tuzing:

Avtomobil dvigatelining vali  $\omega=180$  rad/s burchak tezlik bilan aylanadi. Agar dvigatel vali shkivining diametri  $d_{sh}=9$  sm, ventilyatorniki esa  $d_v=6$  sm bo'lsa, tasmaning chiziqli tezligini ( $v = \omega_1 \cdot R_1$ ) va avtomobil tasmaining burchak tezligini ( $\omega_2 = \frac{v}{R_1}$ ) aniqlang.

### 12-variant

Quyidagi masalani yechish uchun dastur tuzing:

Asosining radiusi  $r=5$  sm, yasovchisi  $l=6$  sm bo'lgan konus to'la sirti va hajmini hisoblash dasturini tuzing.

### 13-variant

Quyidagi masalani yechish uchun dastur tuzing:

Asosining radiusi  $r=10$  sm, asoslarining qarama-qarshi nuqtalarini tutashiruvchi kesma uzunligi  $x=30$  sm bo'lgan silindr to'la sirti va hajmini hisoblash dasturini tuzing.

### 14-variant

Quyidagi masalani yechish uchun dastur tuzing:

Asosi  $a$ , yon tomoni  $b$ , ular orasidagi burchak  $60^0$  bo'lgan teng yonli uchburchak yuzi, perimetri va balandligini hisoblash dasturini tuzing.

### 15-variant

Quyidagi masalani yechish uchun dastur tuzing:

Aylana radiusi  $r=9$  sm berilgan bo'lsa, aylananing yuzi  $s$  va perimetri  $p$  ni hisoblash dasturini tuzing.



### 16-variant

Quyidagi masalani yechish uchun dastur tuzing:

Uchburchakning tomonlari  $a=15$  sm,  $b=26$  sm va  $c=37$  sm bo'lsa, uchburchak yuzini va unga tashqi chizilgan aylana radiusini toping.

### 17-variant

Quyidagi masalani yechish uchun dastur tuzing:

Avtomobil dvigatelining vali 20 m/s tezlik bilan aylanadi. Agar dvigatel vali shkivining radiusi  $r_{sh}=9$  sm, ventilyatorniki esa  $r_v=6$  sm bo'lsa, dvigatel valining burchak tezligini ( $\omega_1 = \frac{v}{R_1}$ ) va avtomobil tasmasining burchak tezligini ( $\omega_2 = \frac{v}{R_1}$ ) aniqlang.

### 18-variant

Quyidagi masalani yechish uchun dastur tuzing:

Asoslari  $a, b$  balandligi  $h$  bo'lgan trapetsiyaning  $p$  perimetri va  $s$  yuzini hisoblash dasturini tuzing.

### 19-variant

Quyidagi masalani yechish uchun dastur tuzing:

Asosining radiusi  $r$ , balandligi  $h$  bo'lgan konus to'la sirti va hajmini hisoblash dasturini tuzing.

### 20-variant

Quyidagi masalani yechish uchun dastur tuzing:

Diametrlari  $d_1, d_2$  bo'lgan rombning  $p$  perimetri va  $s$  yuzini hisoblash dasturini tuzing.

## 3 - laboratoriya ishi

**Mavzu: Tarmoqlanuvchi algoritmlarga dastur tuzish. Shartli, shrtsiz o'tish va tanlash operatorlaridan foydalanish.**

Tarmoqlanuvchi algoritm biror shartni tekshirishni o'z ichiga oladi. Shartni tekshirish natijasiga ko'ra u yoki bu amallar ketma-ketligi bajariladi. Tarmoqlanuvchi algoritmlarni dasturlashda *if* shartli o'tish operatori, *goto* shartsiz o'tish operatori va *switch* tanlash operatoridan foydalaniladi.

**Shartli o'tish operatori.** Shartli o'tish operatori ikki ko'rinishda ishlatilishi mumkin:

*if (ifoda) 1- operator; else 2- operator;*

yoki

*if (ifoda) 1-operator;*

Shartli o'tish operatori bajarilganda avval **ifoda** hisoblanadi ; agar qiymat rost, ya'ni noldan farqli bo'lsa, **1- operator** bajariladi. Agar qiymat yolg'on, ya'ni nol bo'lsa, **2-operator** bajariladi.

*if( n>0)*

*if(a>b) Z=a; else Z=b;*

Misol tariqasida uchta berilgan sonning eng kattasini aniqlash dasturini ko'ramiz:

**1-misol.**

```
#include <iostream>
using namespace std;
int main()
{
float a,b,c,max;
cout <<"\n a="; cin>>a;
cout <<"\n b="; cin>>b;
cout <<"\n c="; cin>>c;
if (a>b)
if (a>c) max=a; else max=c;
else
if (b>c) max=b; else max=c;
cout <<"\n" <<max;
return 0;
}
```

Keyingi misolda kiritilgan ball va maksimal ball asosida baho aniqlanadi:

**2-misol.**

```
#include <iostream>
using namespace std;
int main()
{ float ball,max_ball,baho,d;
cout<<"ball="; cin>>ball;
cout<<"\n max_ball="; cin>>max_ball;
d=ball/max_ball;
if (d>0.85) baho=5; else
if (d>0.75) baho=4; else
```

```

if ( $d > 0.55$ ) baho=3; else baho=2;
cout<<"\n"<< baho;
return 0;
}

```

**3-misol.** Quyidagi funksiyaning qiymati hisoblansin:

$$Y = \begin{cases} \frac{\sin^3 ax^2}{\sqrt{x^2 + 1}} & \text{agar } x < q, \\ \frac{\cos(ax) + e^{-ax^3}}{\sqrt[3]{x^2} \operatorname{arctg}x} & \text{agar } x \geq q, \end{cases}$$

bu yerda  $a=5,41$ ;  $q=3$ .

Dastur tuzilishi:

```

#include <iostream>
#include <math.h>
using namespace std;
int main()
{
float a,b,x,y;
a=5.41; b=3;
cout<<"x ning qiymatini kiriting"; cin>>x;
if (x<b) y=pow(sin(a*x*x),3)/sqrt(x*x+1);
else y=(cos(a*x)+exp(-a*pow(x,3)))/(pow(x,2./3)+atan(x));
cout<<"\n y="<<y;
return 0;
}

```

**Shartsiz o‘tish operatorining** umumiy ko‘rinishi quyidagicha: *goto n*.

Bu operator dasturning boshqa qismiga o‘tishga imkon beradi. Manzil belgi bilan aniqlanadi. Belgi identifikatordan iborat bo‘lib, undan keyin ikki nuqta(:) qo‘yiladi. Yozilishi:

```

goto belgi;

```

```

..

```

```

..

```

**belgi: Operator;**

**Tanlash operatori** *switch* bir nechta operatoridan birini tanlash yordamida u yoki bu amallar ketma-ketligining bajarilishini ta’minlaydi. Tanlash operatorining umumiy ko‘rinishi quyidagicha:

```

switch (ifoda)
{
    case 1-belgi:
        1-operatorlar ketma-ketligi;
        break;
    case 2-belgi:
        2-operatorlar ketma-ketligi;
        break;

    ...

    case n-belgi:
        n- operatorlar ketma-ketligi;
        break;
    default
        operatorlar ketma-ketligi;
}

```

bu yerda

*ifoda* - butun (int) yoki belgi (char) turidagi ifoda;

*1-doimiy, 2-doimiy, ...,n- doimiy* - ifodaning qiymati taqqoslanadigan doimiy qiymatlar. Doimiy qiymatlar butun yoki belgi turida bo‘lishi kerak;

*1-operatorlar ketma-ketligi, 2-operatorlar ketma-ketligi, ...,n- operatorlar ketma-ketligi* - belgilarga mos bo‘lgan operatorlar ketma-ketligi;

*default operatorlar ketma-ketligi* - ifoda va doimiylarning qiymati o‘rtasida hech qanday moslik topilmagan holatda bajariladiga amallar ketma-ketligi.

**4-misol.** Kiritilgan belgining unli harf yoki boshqa belgi ekanligini aniqlash masalasi.

```

#include <iostream>
using namespace std;
int baho; char c;
int main()
{cin >> c;
switch(c)
{
case 'a':
case 'u':
case'o':
case 'i': cout <<"\n Kiritilgan belgi unli harf" ;break;

```

```

default: cout <<"\n Kiritilgan belgi unli harf emas";
}
return 0;
}

```

**5-misol.** Qandaydir jarayonning davom etishi yoki davom etmasligini aniqlash dasturi.

```

#include <iostream>
using namespace std;
int main()
{
char Javob;
cout<<"Jarayon davom etsinmi? (= 'y', 'Y')";
cin>> Javob;
switch (Javob)
{
case 'Y' :
case 'y' : cout<<"Jarayon davom etadi!\n";
break;
default :
cout <<"Jarayon tygadi!\n";
return 0;
}
return 0;
}

```

**6-misol.** Kiritilgan raqam ko‘rinishidagi bahoni so‘z ko‘rinishida ifodalash dasturi.

```

# include <iostream>
using namespace std;
int main()
{
int baho;
cin>> baho;
switch(baho)
{
case 2:cout <<"\n yomon";break;
case 3:cout <<"\n o‘rta";break;
case 4:cout <<"\n yahshi";break;

```

```

case 5:cout <<"\n a'lo";break;
default: cout <<"\n baho nato'g'ri kiritilgan";
};
return 0;
}

```

**7-misol.** Berilgan koordinatalar tekisligi choragi nomeri bo'yicha unda yotuvchi nuqta koordinatalari qiymatlari ishorasini aniqlash dasturi.

```

# include <iostream>
using namespace std;
int main()
{ float chorak;
cout<< "chorak="; cin>>chorak;
switch(chorak)
{
case 1: cout<<"\n koordinatalar qiymatlari: x>0,y>0");break;
case 2: cout<<"\n koordinatalar qiymatlari: x<0,y>0');break;
case 3: cout<<"\n koordinatalar qiymatlari: x<0,y<0');break;
case 4: cout<<"\n koordinatalar qiymatlari: x>0,y<0'); break;
default: cout <<"\n chorak noto'g'ri kiritilgan";
}
return 0;
}

```

### **Laboratoriya ishi variantlari:**

#### **1-variant**

Quyidagi masalani yechish uchun dastur tuzing:

To'g'ri parallelepipedning asosining tomonlari  $a=5$  sm,  $b=6$ sm, ular orasidagi burchak  $\alpha=30^0$ , yon qirralari  $l=4$  sm va  $x$  o'zgaruvchining qiymati berilgan. Agar  $x>0$  bo'lsa, parallelepipedning to'la sirtini hisoblang,  $x\leq 0$  bo'lsa, parallelepipedning hajmini hisoblang.

#### **2-variant**

Quyidagi masalani o'rganing va uni yechish uchun dastur tuzing:

Piramidaning asosi teng yonli uchburchak bo'lib, ushbu uchburchakning asosi  $a=6$  sm, balandligi  $h=9$  sm. Piramidaning yon qirralari  $l=13$  sm ga teng.  $x$  o'zgaruvchi qiymati berilgan. Agar  $x>0$  bo'lsa, piramidaning to'la sirti yuzini hisoblang, aks holda piramidaning hajmini hisoblang.

### 3-variant

Quyidagi yechish uchun dastur tuzing:

Silindr asosining radiusi  $r = 22$  sm, balandligi  $h = 7$  sm va  $x$  o'zgaruvchining qiymati berilgan. Agar  $x < 0$  bo'lsa, silindr to'la sirti yuzini hisoblang,  $x \leq 0$  bo'lsa, silindr hajmini hisoblang.

### 4-variant

Quyidagi masalani yechish uchun dastur tuzing:

Konus asosining radiusi  $r = 10$  sm, balandligi  $h = 7$  sm va  $x$  o'zgaruvchining qiymati berilgan. Agar  $x > 0$  bo'lsa, konus to'la sirtining yuzini hisoblang,  $x \leq 0$  bo'lsa, konus hajmini hisoblang.

### 5-variant

Quyidagi masalani yechish uchun dastur tuzing:

To'g'ri burchakli uchburchakning katetlari  $a = 15$  va  $b = 20$  sm va  $x$  o'zgaruvchining qiymati berilgan. Agar  $x > 0$  bo'lsa, gipotenuzaga o'tkazilgan balandlik uzunligini hisoblang,  $x \leq 0$  bo'lsa, gipotenuzaga tushirilgan mediana uzunligini hisoblang.

### 6-variant

Quyidagi masalani yechish uchun dastur tuzing:

Parallelogramm diagonallari orasidagi burchak  $\alpha = 120^\circ$ , diagonallari esa  $d_1 = 60$  sm va  $d_2 = 32$  sm.  $x$  o'zgaruvchining qiymati berilgan. Agar  $x > 0$  bo'lsa, parallelogrammning kichik tomonini toping,  $x \leq 0$  bo'lsa, katta tomonini toping.

### 7-variant

Quyidagi masalani yechish uchun dastur tuzing:

Parallelogrammning katta tomoniga tushirilgan balandligi  $h = 24$  sm bo'lib, uni  $a_1 = 7$  sm va  $a_2 = 32$  sm bo'lgan kesmalarga ajratadi.  $x$  o'zgaruvchining qiymati berilgan. Agar  $x > 0$  bo'lsa, parallelogrammning kichik diagonalini toping,  $x \leq 0$  bo'lsa, parallelogrammning perimetrini toping.

### 8-variant

Quyidagi masalani yechish uchun dastur tuzing:

Trapetsiyaning asoslari  $b=20$  sm va  $a=60$  sm, yon tomonlari  $c=13$  sm va  $f=37$  sm.  $x$  o'zgaruvchi qiymati berilgan. Agar  $x>0$  bo'lsa, trapetsiyaning balandligini toping,  $x\leq 0$  bo'lsa, trapetsiyaning perimetrini toping.

### 9-variant

Quyidagi masalani yechish uchun dastur tuzing:

Aylana nuqtasidan diametrga perpendikulyar o'tkazilgan bo'lib, diametrni  $d_1=16$  sm va  $d_2=9$  sm uzunlikdagi kesmalarga ajratadi.  $x$  o'zgaruvchining qiymati berilgan. Agar  $x<0$  bo'lsa, perpendikulyar uzunligini toping,  $x\leq 0$  bo'lsa, diametr uzunligini toping.

### 10-variant

Quyidagi masalani yechish uchun dastur tuzing:

Uchburchakning tomonlari  $a=15$  sm,  $b=26$  sm va  $c=27$  sm.  $x$  o'zgaruvchining qiymati berilgan. Agar  $x<0$  bo'lsa, uchburchakka ichki chizilgan aylana radiusini toping,  $x\leq 0$  bo'lsa, uchburchakka tashqi chizilgan aylana radiusini toping.

### 11-variant

Quyidagi masalani yechish uchun dastur tuzing:

Agar uchburchak asosi  $a>5$  bo'lsa, uchburchak yuzi  $s=20$  ga, aks holda  $s=10$  ga teng.

### 12-variant

Quyidagi masalani yechish uchun dastur tuzing:

Ikkita  $R_1=4$  va  $R_2=6$  qarshiliklar uchun  $x=1$  bo'lganda, umumiy qarshilik  $R = R_1 + R$  formula bo'yicha, aks holda  $R = \frac{R_1+R_2}{R_1 \cdot R_2}$  formula bo'yicha hisoblanadi.

### 13-variant

Quyidagi masalani yechish uchun dastur tuzing:

Koeffitsiyentlari  $a=4$ ,  $b=5$ ,  $c=2$  bo'lgan kvadrat tenglama uchun  $d>0$  bo'lganda ildizlar soni  $n=2$  ta,  $d=0$  bo'lganda ildizlar soni  $n=1$  ta,  $d<0$  bo'lganda ildizlar soni  $n=0$  ta bo'ladi.



#### 14-variant

Quyidagi masalani yechish uchun dastur tuzing:

Kvadratning tomoni  $a=20$  sm.  $x$  o'zgaruvchi qiymati berilgan. Agar  $x>0$  bo'lsa, kvadratning perimetrini toping,  $x\leq 0$  bo'lsa, kvadratning yuzini toping.

#### 15-variant

Quyidagi masalani yechish uchun dastur tuzing:

$x$  va  $y$  haqiqiy sonlar berilgan. Agar  $x>0, y>0$  bo'lsa,  $ch=1$ ga teng; agar  $x<0, y>0$  bo'lsa,  $ch=2$ ga teng; agar  $x<0, y<0$  bo'lsa,  $ch=3$ ga teng; agar  $x>0, y<0$  bo'lsa,  $ch=4$ ga teng.

#### 16-variant

Quyidagi masalani yechish uchun dastur tuzing:

$F_t$ - traktorning tortish kuchi,  $F_i$ -yerning traktor bilan ishqalanish kuchi berilgan. Agar  $F_t>F_i$  bo'lsa, najaviy kuch qiymati  $F=F_t-F_i$ ; agar  $F_t=F_i$  bo'lsa,  $F=0$ ; agar  $F_t<F_i$  bo'lsa,  $F=F_i-F_t$  bo'ladi.

#### 17-variant

Quyidagi masalani yechish uchun dastur tuzing:

$F_1, F_2$  kuchlar va ular orasidagi burchak  $a$  berilgan. Agar  $a=90^\circ$  bo'lsa, teng ta'sir etuvchi kuch  $F = \sqrt{F_1^2 + F_2^2}$  ga, aks holda  $F = \sqrt{F_1^2 + F_2^2 - 2F_1 \cdot F_2 \cdot \cos(180^\circ - a)}$  ga teng bo'ladi.

#### 18-variant.

Quyidagi masalani yechish uchun dastur tuzing:

Harakatlanayotgan jism uchun harakat radiusi  $R$ , burchk tezlik  $\omega$ , tezlanish  $a$ , harakatlanish vaqti  $t$  berilgan. Agar  $R=0$  bo'lsa, chiziqli tezlik  $v = a \cdot t$  ga, aks holda  $v = \omega \cdot R$  ga teng bo'ladi.

#### 19-variant

Quyidagi masalani yechish uchun dastur tuzing:

Matematik mayatnik uzunligi  $l$ , chastotasi  $\nu$ , erkin tushish tezlanish  $g$  berilgan.

$x>0$  da tebranish davri  $T = 2\pi \sqrt{\frac{l}{g}}$  ga,  $x\leq 0$  bo'lganda,  $T = \frac{1}{\nu}$  ga teng bo'ladi.

## 20-variant

Quyidagi masalani yechish uchun dastur tuzing:

Avtomobil dvigatelingning vali  $\omega=180$  rad/s burchak tezlik bilan aylanadi. Agar dvigatel vali shkivining diametri  $d_{sh}=9$  sm, ventilyatorniki esa  $d_v=6$  sm bo`lsa, tasmaning chiziqli tezligini ( $v = \omega_1.R_1$ ) va avtomobil tasmalarning burchak tezligini ( $\omega_2 = \frac{v}{R_1}$ ) aniqlang.

## 4 - laboratoriya ishi

**Mavzu: Takrorlanuvchi algoritmlarga dastur tuzish. for, while, do while operatorlaridan foydalanib dastur tuzish.**

Ko'pincha masalalarni dasturlashda bir xil buyruqlar ketma-ketligini bir necha bor bajarilishga to'g'ri keladi. Bunday jarayon sikl deb ataladi. Muayyan buyruqlar ketma-ketligi yangi qiymatlar bilan bir necha marta takrorlansa, bunday algoritmlar takrorlanuvchi(siklik) algoritm deyiladi. C++ tilida takrorlanuvchi algoritmlarni dasturlash uchun maxsus sikl operatorlari mavjud: *for*, *while*, *do while*. Takrorlanuvchi algoritmlarni dasturlash uchun, shuningdek, shartli o'tish operatoridan ham foydalanish mumkin.

**For operatori** takrorlanishlar soni ma'lum bo'lganda ishlatiladi. For operatoriining umumiy yozilishi quyidagicha:

```
for (initsializatsiya; ifoda; o'sish)
```

```
{
```

```
    // operatorlar ketma-ketligi
```

```
    // ...
```

```
}
```

bu yerda

**initsializatsiya**- o'zlashtirish operatori bo'lib, unda sikl o'zgaruvchisining boshlang'ich qiymati beriladi. ushbu o'zgaruvchi sikl ishini bosharuvchi hisoblagich vazifasini bajaradi;

**ifoda** - sikl o'zgaruvchisining qiymati tekshiriladigan shartli ifoda. Ushbu bosqichda siklning keyingi bajarilishi aniqlanadi;

**o'sish** - har bir iteratsiyadan keyin sikl o'zgaruvchisining qiymati qanday o'zgarishini aniqlaydi.

**for** operatori **ifoda rost** qiymatni qabul qilsa, bajarilaveradi. **Ifodaning** qiymati **yolg'on** bo'lgandan so'ng to'xtaydi va **for** dan keyingi operator bajariladi.

**1-misol.** 100 dan 300 gacha bo'lgan butun sonlarning yig'indisini toping. Ushbu masalaning dasturi quyidagicha bo'lagi.

```
# include <iostream>  
using namespace std;  
int main()  
{  
int sum,i;  
sum = 0;  
for(i = 100; i <= 300; i++)  
sum = sum + i;  
cout<<"summa="<<sum;  
return 0; }
```

**2-misol.** n natural son berilgan. Quyidagi yig'indini hisoblang:

$$S = 1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n}$$

Masalaning dasturi:

```
# include <iostream>  
using namespace std;  
int main()  
{  
// s = 1 + 1/2 + 1/3 + ... + 1/n  
int n;  
cout<<"n ni kiriting="; cin>>n;  
float s = 0;  
for (int i = 1; i<=n; i++)  
s = s + 1.0/i;  
cout<<"\n s="<<s;  
return 0;  
}
```

**while operatori.**

Sikldagi takrorlanishlar soni noma'lum bo'lib, ma'lum shartga bog'liq bo'lsa, siklni tashkil qilishda **while** operatoridan foydalanish mumkin.

**while** operatorining umumiy shakli quyidagicha:

```
while (ifoda)  
{  
// operatorlar ketma-ketligi
```

```
// ...  
}
```

bu yerda, **ifoda**- C ++ tilidagi mantiqiy ifodadir. Operatorlar ketma-ketligi **ifoda rost** qiymatni qabul qilsa bajariladi. **Ifoda yolg'on** qiymatni qabul qilgandan so'ng, **while** operatorining bajarilishi to'xtaydi va boshqaruv **while** dan keyingi operatorga o'tadi.

**3-misol.** Haqiqiy  $a$  soni berilgan. Quyidagi shartni qanoatlantiruvchi eng kichik  $n$  ni toping

$$1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n} > a$$

Dasdrlab, yig'indining qiymati  $a$  dan kichik. Har bir iteratsiyada yig'indining qiymati oshib boradi. Ma'lum bir qadamdan keyin bu miqdor  $a$  dan katta bo'ladi.  $n$  ning shu shart bajariladigan eng kichik qiymatini topish kerak. Masalani yechish dasturi quyidagicha:

```
# include <iostream>  
using namespace std;  
int main()  
{  
float a,sum;  
int n;  
// a va nni kiritish  
a = 2.2;  
n = 1;  
sum = 1.0/n;  
while (sum < a)  
{  
n++;  
sum = sum + 1.0/n;  
}  
cout<<"\n sum="<<sum;  
return 0;  
}
```

**4-misol.** Natural son berilgan. Undagi 3 raqamlari sonini aniqlang. Masalani yechish dasturi:

```

# include <iostream>
using namespace std;
int main()
{
int n; // berilgan natural son
int k; // berilgan sondagi 3 raqamlari soni
int t, d; // qo'shimcha o'zgaruvchilar
// n ni kiritish
cout<<"n sonini kiriting"; cin>>n;
t = n; // n ning nusxasini olish
k= 0;
while (t> 0)
{
    d = t% 10; // oxirgi raqamni ajratib ko'rsatish
    if (d == 3) k ++;
    t = t / 10; //oxirgi raqamni o'chirish
}
cout<<n<<" soniagi 3 raqamlar= "<<k<<" ta";
return 0;
}

```

### **do... while operatori.**

Bu operator takrorlanishlar soni ma'lum bo'lmagan sikllarni tashkil etishda ishlatiladi. Bu operatorning ishlash jarayonida har bir qadamdan keyin shart tekshiriladi. **do ... while** operatori da takrorlanishlar kamida bir marta bajariladi. **do... while** opratorida **for** va **while** operatorlaridan farqli ravishda shart sikldan chiqishda tekshiriladi.

**do ... while** operatorining umumiy shakli:

```

do
{
    // operatorlar ketma-ketligi
    // ...
}
while (ifoda);

```

bu yerda, *ifoda* - sikl o'zgaruvchisining qiymati tekshiriladigan ifoda. Ushbu bosqichda siklning keyingi bajarilishi aniqlanadi. Katta qavslarni qo'yish shart emas.

Sikl quyidagicha ishlaydi. Dastlab, sikl tanasi bajariladi. Keyin ifoda qiymati (shartli ifoda) tekshiriladi. Agar ifoda qiymati rost (true) bo'lsa, sikl tanasi yana bajariladi. Ifodaning qiymati yolg'on(false) bo'lgandan so'ng, sikl to'xtaydi.

**5-misol.** Argument  $x$   $a$  dan  $b$  gacha  $h$  qadam bilan o'zgarganda  $Y = e^{\cos x} + \log_c(x + 4)$  funksiya qiymatlari hisoblansin. Dasturni *for*, *while* va *repeat* operatorlaridan foydalanib quyidagi uch xil ko'rinishda yozish mumkin:

```
# include <iostream>
# include <math.h>
using namespace std;
int main()
{
float a,b,h,c,y,x;
cout<<"a,b,h,c larni kiriting \n"; cin>>a>>b>>h>>c;
x=a;
do
{
y=exp(cos(x))+log(x+4)/log(c);
cout<<" x="<<x<<" y="<<y<<endl;
x+=h;
}
while (x<=b);
return 0;
}
```

### Laboratoriya ishi variantlari:

#### 1-variant

Takrorlash operatorining uchta ko'rinishining har biridan foydalanib,  $F = 10!$  hisoblang.

#### 2-variant

Quyidagini hisoblash dasturini tuzing:

$$P = \left(1 - \frac{1}{2^2}\right)\left(1 - \frac{1}{3^2}\right)\dots\left(1 - \frac{1}{3^2}\right);$$

### 3-variant

Quyidagini hisoblash dasturini tuzing:

$$y = \cos(x) + \cos(x^2) + \cos(x^3) + \dots + \cos(x^{30});$$

### 4-variant

Quyidagini hisoblash dasturini tuzing:

$$y = 1! + 2! + 3! + \dots + n!, n > 1.$$

### 5-variant

Berilgan 10 ta haqiqiy sonlarning eng kattasi va eng kichigi orasidagi ayirmani topishga dastur tuzing.

### 6-variant

Quyidagi  $y$  ni hisoblash dasturini tuzing:

$$y = (2n - 1)! = 1 * 3 * 5 * \dots * (2n - 1), n > 0;$$

### 7-variant

$n$  ta ixtiyoriy natural sondan tashkil topgan ketma-ketlik eng kichik elementining tartib nomerini aniqlang.

### 8-variant

Quyidagi  $y$  ni hisoblash dasturini tuzing:

$$y = \sin 1 + \sin 1.1 + \sin 1.2 + \dots + \sin 2.$$

### 9-variant

Uch xil siklik operatorlarni ishlatib,  $n=(a+b)!$  ni hisoblash uchun dasturlar tuzilsin.

### 10-variant

$S = \prod_{\kappa=1}^7 \sum_{n=1}^5 \cos^2 \kappa n$  ni topish uchun uch xil takrorlash operatorlari yordamida dastur tuzing.

### 11-variant

$P = \sum_{k=1}^5 \prod_{i=1}^{10} \sqrt{b+i}$  qiymatini topish uchun uch xil takrorlash operatorlari yordamida dastur tuzing.

### 12-variant

$S = \sum_{k=1}^{10} \sum_{n=1}^5 \text{Cos}^2(\kappa \cdot n)$  ni topish uchun uch xil takrorlash operatorlari yordamida dastur tuzing.

### 13-variant

$S = \sum_{k=1}^{10} \prod_{i=1}^5 (k^3 + \sin i)$  ni topish uchun uch xil takrorlash operatorlari yordamida dastur tuzing.

### 14-variant

$S = \prod_{i=1}^5 \prod_{j=1}^3 (i + j)$  ni topish uchun uch xil takrorlash operatorlari yordamida dastur tuzing.

### 15-variant

$S = \sum_{k=1}^{10} \sin k^3 + \prod_{i=1}^5 \ln i$  ni topish uchun uch xil takrorlash operatorlari yordamida dastur tuzing.

### 16-variant

1 dan 1000 gacha bo'lgan natural sonlar o'rtacha arifmetigini hisoblashga dastur tuzing.

### 17-variant

10 dan 15 gacha bo'lgan natural sonlar o'rtacha geometrigini hisoblashga dastur tuzing.

### 18-variant

Quyidagi jadvalda sutka davomida haroratning o'zgarish jadvali berilgan. Sutkalik o'rtacha haroratni hisoblash dasturini tuzing.

Vaqt(soat)	1	2	3	4	5	6	7	8	9
Harorat(gradus)	-5	-5	-7	-8	-5	-5	-3	-3	-2
Vaqt(soat)	10	11	12	13	14	15	16	17	18
Harorat(gradus)	0	+2	+4	+7	+8	+9	+7	+3	+1
Vaqt(soat)	19	20	21	22	23	24			
Harorat(gradus)	0	-1	-4	-6	-7	-8			



## 19-variant

$S = \sum_{k=1}^4 \lg k^3 + \prod_{i=1}^5 \ln^2 i$  ni topish uchun uch xil takrorlash operatorlari yordamida dastur tuzing.

## 20-variant

$S = \sum_{k=1}^4 \lg k^3 - \prod_{i=1}^5 \log_2^3 i$  ni topish uchun uch xil takrorlash operatorlari yordamida dastur tuzing.

## 5 – laboratoriya ishi

**Mavzu: Bir o‘lchovli massivlar uchun massiv operatorlaridan foydalanib dastur tuzish.**

**Massiv** - bir xil turdagi tartiblangan ma'lumotlar to‘plamidir. Massiv quyidagi ko‘rinishda yoziladi: *o‘zgaruvchi\_turi massiv\_nomi [massiv\_uzunligi]*. O‘zgaruvchining turidan keyin massiv nomi, so‘ngra kvadrat qavslarda uning o‘lchamlari ko‘rsariladi. Masalan, 4 ta sondan iborat massivni aniqlaymiz: `int mas [4]`. Ushbu massiv to‘rtta elementga ega, ammo ularning qiymatlari aniqlanmagan. Bu elementlarga o‘rta qavslar orqali boshlang‘ich qiymatlarni berishimiz mumkin: `int mas [4] = {1,2,3,4};`

Agar massiv o‘lchamiga nisbatan qiymatlar kamroq bo‘lsa, unda qiymatlar massivning dastlabki elementlariga beriladi. Agar massiv o‘lchamiga nisbatan qiymatlar ko‘proq bo‘lsa, unda kompilyatsiya paytida xato yuzaga keladi: `int mas [4] = {1, 2, 3, 4, 5, 6};`. Bu yerda massiv 4 o‘lchamga ega, ammo unga 6 qiymat berilgan. Agar massivning o‘lchami aniq ko‘rsatilmagan bo‘lsa, unda o‘lcham berilgan qiymatlar soniga qarab aniqlanadi: `int mas [] = {1, 2, 3, 4, 5, 6};`. Bu holda, massivda 6 element mavjud bo‘lib, massiv o‘lchami ham 6 ga teng.

Belgilar massivlarini initsializatsiyalash (qiymat berish) o‘ziga xos xususiyatlarga ega. Ushbu belgili massivda biz boshlang‘ich qiymatni belgilar orqali ham, satr ko‘rinishida ham berishimiz mumkin:

```
char s1 [] = {'h', 'e', 'l', 'l', 'o'};  
char s2 [] = "dunyo";
```

Bundan tashqari, ikkinchi holatda, s2 massivida 5 ta emas, balki 6 ta elementga ega bo‘ladi. Chunki belgili massivga satr orqali qiymat berganda massiv oxiriga avtomatik ravishda `\0` nol belgisi qo‘shiladi. Shu bilan birga, bitta massiv elementlarini boshqa massivga o‘lashtirishga ruxsat berilmaydi:

```
int nums1 [] = {1,2,3,4,5};
```

```
int nums2 [] = nums1; // xato
nums2 = nums1; // xato
```

Massivni aniqlagandan so‘ng, indeks orqali uning elementlariga murojaat qilishimiz mumkin. Indekslar noldan boshlanadi, shuning uchun birinchi elementga kirish uchun 0 indeksidan foydalanish kerak. Elementga indeks orqali murojaat qilib, uning qiymatini olishimiz yoki o‘zgartirishimiz mumkin:

#### 1-misol.

```
#include <iostream>
using namespace std;
int main()
{
    int numbers[4] = {1,2,3,4};
    int first_number = numbers[0];
    cout << first_number<<endl; // 1
    numbers[0] = 34;
    cout << numbers[0] << endl; // 34
    return 0;
}
```

Sikldan foydalanib, massivning elementlariga indekslar orqali murojaat qilish mumkin:

#### 2-misol.

```
#include <iostream>
using namespace std;
int main()
{
    int numbers[4] = {1,2,3,4};
    int size = sizeof(numbers)/sizeof(numbers[0]);
    for(int i=0; i < size; i++)
        cout << numbers[i] << endl;
    return 0;
}
```

Sikl yordamida massivni bo‘ylab yurish uchun avval massivning uzunligini topish kerak. Uzunlikni topish uchun sizeof operatoridan foydalaniladi. Aslida, massiv uzunligi uning elementlarining umumiy uzunligiga teng. Barcha elementlar bir xil turda va xotirada bir xil hajmni egallaydi. Shunday qilib, sizeof (numbers) ifodasidan foydalanib, butun massiv uzunligini baytlarda topamiz va sizeof (numbers [0]) yordamida bitta element uzunligini baytlarda topamiz. Birinchi qiymatni ikkinchisiga bo‘lib, massivdagi elementlar sonini aniqlash mumkin.

Soo'ng for operatoridan fodalanib,  $i$  hisoblagich qiymati massiv uzunligiga teng bo'lgunga qadar barcha elementlarga muroaat qilamiz.

**3-misol.** Haqiqiy  $a(n)$ ,  $n \leq 15$  massivning barcha musbat elementlarining o'rta arifmetigini hisoblang.

Belgilashlarni kiritamiz.  $i$  - massiv elementlari indesklarining qiymatlarini aniqlovchi o'zgaruvchi,  $i=1,2,..k$  ( $k$  - massiv elementlari soni,  $k \leq 15$ ),  $S$  va  $kn$  mos ravishda musbat elementlarning o'rta arifmatik qiymati va ularning soni.  $a[i] > 0$  shart bajarilsa  $S=S+a[i]$ ,  $kn=kn+1$  yig'indi hisoblanishi kerak. Ushbu yig'indini hisoblash tugagandan keyin  $S=S/kn$  hisoblanadi, ya'ni musbat elementlar yig'indisi musbat elementlar soniga bo'linadi. Agar  $kn=0$  bo'lsa, massivning musbat elementlari mavjud emas.

```
#include <iostream>
using namespace std;
int main()
{
    int i,k,kn;
    float s;
    cout<<"k ni kiriting \n"; cin>>k;
    float a[k]; s=0; kn=0;
    cout<<"a massivni kiriting \n";
    for (i=1;i<=k; i++) cin>>a[i];
    for (i=1;i<=k; i++)
        if (a[i]>0)
            {s=s+a[i]; kn=kn+1;}
    if (kn==0){
        cout<<"Musbat elementlari yo'q \n";
        goto t;}
    else
        {s=s/kn;
        cout<<"s="<<s;
        }
    t;;
    return 0;
}
```

**4-misol.**  $A(2,3,4)$  va  $B(3,-3,2)$  vektorlar uchun  $y = \max(|A|, |B|)$  funktsiyani hisoblang.

```
# include <iostream>
```

```

# include <math.h>
using namespace std;
int main()
{
float y, a[]={2,3,4}, b[]={3,-3,2};
for (i=1;i<=3;i++)
{
a1=a1+a[i];
b1=b1+b[i];
}
a1=sqrt(a1);
b1=sqrt(b1);
if(a1>b1)
y=a1; else y=b1;
cout<<"y="<<y<<endl;
return 0;
}

```

### Laboratoriya ishi variantlari:

#### 1-variant

A(10,2,34,-23,32) va B(1,12,-4,12,-43) vektorlarning elementlari yig'indisi hisoblash dasturini tuzing.

#### 2-variant

A(1,12,83,12) va B(1,2,4,7,-4) vektorlarning elementlari airmasini hisoblash dasturini tuzing.

#### 3-variant

A(2,3,8,12) va B(21,12,7,-4) vektorlar uchun  $y = |A + B|$  funktdiyani hisoblash dasturini tuzing.

#### 4-variant

A(4,8,18,12) va B(12,12,5,-14) vektorlar uchun  $y = |A| + |B|$  funktdiyani hisoblash dasturini tuzing.

#### 5-variant

A(22,-13,8,12) va B(21,-12,17,-4) vektorlar uchun  $y = \sqrt{|A| \cdot |B|}$  funktdiyani hisoblash dasturini tuzing.

### 6-variant

A(20,13,-8,2) va B(-11,22,17,-14) vektorlar uchun  $y = \sqrt[3]{|A| + |B|}$  funktdiyani hisoblash dasturini tuzing.

### 7-variant

A(-12,15,-18,20) va B(11,-22,15,-13) vektorlar uchun  $y = \sin\left(\frac{|A|}{|B|}\right)$  funktdiyani hisoblash dasturini tuzing.

### 8-variant

A(2,1,-8,2) va B(-1,22,7,-4) vektorlar uchun  $y = \operatorname{tg}\left(\sqrt[3]{|A| + |B|}\right)$  funktdiyani hisoblash dasturini tuzing.

### 9-variant

A(10,1,-18,22) va B(-1,12,17,-8) vektorlar uchun  $y = \cos(|A| + |B|)$  funktdiyani hisoblash dasturini tuzing.

### 10-variant

A(20,11,-28,23) va B(-15,22,-75) vektorlar uchun  $y = |A| + 2^{|B|}$  funktdiyani hisoblash dasturini tuzing.

### 11-variant

A(10,21,-18,22) va B(-1,2,7,-8) vektorlar uchun  $y = \cos^3(|A| + |B|)$  funktdiyani hisoblash dasturini tuzing.

### 12-variant

A(5,1,-4,3) va B(-11,22,-18,9) vektorlar uchun  $y = \sqrt[3]{\operatorname{tg}(|A| - |B|)}$  funktdiyani hisoblash dasturini tuzing.

### 13-variant

A(5,4,-8,2) va B(-1,2,7,-8) vektorlar uchun  $y = \ln(|A| + |B|)$  funktdiyani hisoblash dasturini tuzing.

### 14-variant

A(10,1,-8,2) va B(-1,2,7,-8) vektorlar uchun  $y = \lg|\cos(|A| + |B|)|$  funktdiyani hisoblash dasturini tuzing.

### 15-variant

A(-3,4,-8,2) va B(-11,12,7,-8) vektorlar uchun  $y = e^{|A|+|B|}$  funktdiyani hisoblash dasturini tuzing.

### 16-variant

A(6,7,-8,3) va B(4,12,-3,8) vektorlar uchun  $y = \sqrt[3]{\sin|\cos(|A| + |B|)|}$  funktdiyani hisoblash dasturini tuzing.

### 17-variant

A(16,27,18,-23) va B(3,1,-2,4) vektorlar uchun  $y = 3^{\cos(|A|+|B|)}$  funktdiyani hisoblash dasturini tuzing.

### 18-variant

A(2,1,-2,3) va B(-15,22,-7,5) vektorlar uchun  $y = \lg|A| + e^{|B|}$  funktdiyani hisoblash dasturini tuzing.

### 19-variant

A(10,11,- 8, 3) va B(-25,22,-17,15) vektorlar uchun  $y = (|A| + 2^{|B|})^{|A|}$  funktdiyani hisoblash dasturini tuzing.

### 20-variant

A(20,11,-28,23) va B(-15,22,-75,1) vektorlar uchun  $y = \operatorname{tg}(|A| + 2^{|B|})$  funktdiyani hisoblash dasturini tuzing.

## Foydalanilgan adabiyotlar

1. Z.S. Abdullaev, S.S. Mirzaev, G. Shodmonova, N.B. Shamsiddinov Informatika va axborot texnologiyalari. Darslik, T.:2012.
2. Oripov M.M., Muhammadiev J.O'. Informatika, informatsion texnologiyalar. Oliy o'quv yurtlari talabalari uchun darslik. T.: TDYUI, 2005, 275 b.
3. U. Yuldashev, Sh.K. Raxmatullaeva. Microsoft Windows-98: O'quv qo'llanma. T., 2001. 29 b.
4. G'ulomov S.S., Shermuhamedov A.T., Begalov B. A. Iqtisodiy informatika: Darslik. Akademik S. S. G'ulomovning umumiy tahriri ostida. T.: «O'zbekiston», 1999. 528 b.

5. G'ulomov S.S. va boshqalar. Axborot tizimlari va texnologiyalari: Oliy o'quv yurti talabalari uchun darslik. Akademik S. S. G'ulomovning umumiy tahriri ostida.—T.: «Sharq», 2000.—592 b.
6. Oripov M. M. va boshqalar. Informatika. Axborot texnologiyalari. Toshkent. TDTU. 2002.
7. Abduqodirov A.A. va boshqalar. «Axborot texnologiyalari» 1-2 qism. Toshkent, 2002.
8. Yuldashev U., Boqiev R. «Informatika» 1-3 qism. Toshkent, 2002 y.
9. Sattorov A. «Informatika va axborot texnologiyalari», Toshkent, O'qituvchi, 2002.
10. Файсман А. Профессиональное программирование на Турбо Паскале. Ташкент, Информекс Корпорейшн, 1992. 271-с.
11. Петров А. В. Вычислительная техника и программирование. Москва. Высшая школа. 1991. 479 с.
12. Грогоно П. Программирование на языке С++. Перевод с английского. Москва, Мир. 1982. 382с.
13. Абрамов С.А., Зима Э. В. Начало программирование на языке С++. Москва, Наука, 1987. 112с.

## Mundarija

Kirirsh.....	3
1-laboratoriya ishi. Algoritmash asoslari. Algoritmarni grafik tasvirlash.....	4
2-laboratoriya ishi.Chiziqli algoritmlarga dasturlash. Muhandislik va suv xo‘jaligi mexanizatsiyasi masalalarida ma’lumotlarni kiritish - chiqarish operatorlaridan foydalanib dastur tuzish. ....	12
3 - laboratoriya ishi. Tarmoqlanuvchi algoritmlarga dastur tuzish. Shartli, shrtsiz o‘tish va tanlash operatorlaridan foydalanish. ....	25
4 - laboratoriya ishi. Takrorlanuvchi algoritmlarga dastur tuzish. for, while, do while operatorlaridan foydalanib dastur tuzish.....	34
5 – laboratoriya ishi. Bir o‘lchovli massivlar uchun massiv operatorlaridan foydalanib dastur tuzish .....	41
Foydalanilgan adabiyotlar.....	46



Aynakulov Sharafidin Abdujalilovich  
Ziyadullayev Davron

**«Axborot texnologiyalari va jarayonlarni matematik modellashtirish»**

**fanidan laboratoriya mashg'ulotlarini bajarish bo'yicha**

### **USLUBIY QO'LLANMA**

**“Suv xo‘jaligi melioratsiyasi ishlarini mexanizatsiyalash”, “Mehnat muhofazasi va texnika xavfsizligi” ( suv xo‘jaligida), “Qishloq xo‘jaligini mexanizatsiyalash”, “Qishloq va suv xo‘jaligida texnik servis”, “Qishloq xo‘jaligida innovatsion texnika va texnologiyalarni qo‘llash”, “Professional ta’lim” (Qishloq xo‘jaligini mexanizatsiyalashtirish) bakalavriat ta’limi yo‘nalishlari uchun**

Muharrir:

M. Mustafayeva

---

Bosishga ruxsat etildi \_\_\_\_\_ 2022 y.

Qog'oz o'lchami 60x84,1/16, hajmi 3 b.t., 10 nusxa.

Buyurtma № \_\_\_\_ . TIQXMMI MTU bosmaxonasida chop etildi.

Toshkent-1000000, Qori Niyoziy ko'chasi, 39-uy.