

**O‘ZBEKISTON RESPUBLIKASI  
OLY VA O‘RTA MAXSUS TA‘LIM VAZIRLIGI  
MIRZO ULUG‘BEK NOMIDAGI  
O‘ZBEKISTON MILLIY UNIVERSITETI**

**Sh. F. MADRAXIMOV, A. M. IKRAMOV, M. R. BABAJANOV**

**C++ TILIDA PROGRAMMALASH  
BO‘YICHA MASALALAR TO‘PLAMI**

*(O‘quv qo‘llanma)*

**Toshkent  
“Universitet”  
2014**

УДК 681.3.01(075.8)

32.973.26-018.2

M.14

Madraximov Sh. F. va boshq. C++ tilida programmalash bo'yicha masalalar to'plami. Toshkent. "Universitet" nashriyoti -2014. -160 bet.

КБК 32.973.26-018.2

Mazkur masalalar to'plami programmalash asoslarini o'zlashtirishdagi amaliy mashg'ulotlarni o'tkazishga mo'ljallangan. To'plam C++ tilida strukturali va obyektga yo'naltirilgan, umumlashgan programmalash mavzulari bo'yicha ajratilgan bo'limlardan tashkil topgan bo'lib, ular namunaviy masala echimi hamda masala va misollar majmuasidan iborat. To'plamdan bakalavriyatning informatika va unga turdosh boshqa yo'nalishlaridagi talabalar foydalanishi mumkin.

**Mas'ul muharrir f.-m.f.d., professor M.M.Aripov**

**Taqrizchilar: f.-m.f.d., professor A.A.Xoldjigitov,  
f.-m.f.d., professor N.A.Ignatev**

Ushbu o'quv qo'llanmani nashr etishga O'zbekiston Respublikasi Oliy va o'rta maxsus ta'lim vazirligining 2013 yil 13-martdagi 82-sonli buyrug'iga asosan ruxsat berilgan (ro'yxatga olish raqami 82-060).

**Madraximov Shavkat Fayzullayevich  
Ikramov Axmat Maoripovich  
Babajanov Mo'min Rajabovich**

**C++ TILIDA PROGRAMMALASH BO'YICHA MASALALAR  
TO'PLAMI  
(O'quv qo'llanma)**

Muharrirlar: D.Akmalova, S.Qurbonov.

Musahhix D.Tolipov

Bosishga ruxsat etildi:18.02.2014y. Bichimi 60x84<sup>1</sup>/6

Nashriyot hisob tabog'i 10. Shartli bosma tabog'i 16,8

Adadi 300 nusxada. Bahosi shartnoma asosida

Buyurtma № \_\_\_\_\_

"Universitet" nashriyoti. Toshkent-100174. Talabalar shaharchasi.

Mirzo Ulug'bek nomidagi O'zbekiston Milliy Universiteti. Ma'muriy bino.

O'zMU bosmaxonasida bosildi.

ISBN-978-9943-305-98-4

## Kirish

C++ tilining asosiy yutuqlaridan biri programmalash masalalarini yechishning zamonaviy usullarini qoʻllab-quvvatlashidir. Shu sababli C++ tilidan samarali foydalanishga oʻrganish uchun tilning sintaksisi va semantikasini bilishning oʻzi yetarli emas. Yanada chuqurroq oʻzlashtirish uchun yechish usullari oʻzaro kesishadigan turli-tuman masalalar qaralishi kerak boʻladi. Bunday masalalarni yechish – til vositalarini ishlatishni va ular nima uchun zarur ekanligini anglash imkonini beradi.

Oʻquv qoʻllanma C++ tilida programmalash boʻyicha nashr qilingan nazariy adabiyotlarga toʻldiruvchi hisoblanadi. Uning mazmunini OʻzMU “Programmalash va tarmoq texnologiyalari” kafedrasida oʻqilayotgan, programmalash asoslarini oʻzlashtirish bilan bogʻliq fanlarning amaliy darslarida foydalanayotgan masala va misollar tashkil qiladi.

Oʻquv qoʻllanma C++ tilida programmalash texnologiyasining asosiy uslublari – strukturali, obyektga yoʻnaltirilgan va umumlashgan programmalash usullarini qamrab oluvchi 27 ta mavzudagi masala va misollardan iborat. Mavzular ketma-ketligi soddadan murakkabga tartiblangan va undagi misollarni yechish turli algoritmlarni qoʻllashni va ilova yozishning maxsus usullaridan foydalanishni talab etadi. Har bir mavzu mos namunaviy masalani izohli yechib koʻrsatishdan boshlanadi va undan keyin topshiriqlar keltiriladi.

C++ tilining asosini tashkil qiluvchilari va ularning imkoniyatlarini ochib berishda strukturali programmalash texnologiyasi qoʻllanilgan boʻlib, qoʻllanmaning 1-21–boblari aynan shu mavzularni qamrab olgan. Bu boblarga C++ tilida berilganlarning standart turlari, til operatorlari, massivlar, satrlar va strukturalar, funksiyalar, berilganlarning dinamik strukturalari, oqimlar va fayllar bilan bogʻliq mavzular kiradi.

C++ tilida obyektga yoʻnaltirilgan va umumlashgan programmalash bilan bogʻliq mavzular qoʻllanmaning 21-27–boblarida oʻz aksini topgan va unda C++ Builder vizual muhitida sinflarni yaratish, vorislik, polimorfizm, operatorlarni qayta yuklash, qoliplar va grafika bilan bogʻliq masalalar keltirilgan.

Qoʻllanma ilovasida programmalash boʻyicha amaliy mashgʻulotlarni bajarishda talaba uchun zarur boʻladigan matematik funksiyalar kutubxonasi, C++ Builder muhiti va uning grafika komponentalari bilan ishlash boʻyicha qisqa maʼlumotlar keltirilgan.

## 1. Berilganlar turlari

Ushbu amamliy mashg'ulotlarda berilganlar turiga mos ravishda kompyuter xotirasida egallagan joyi (baytlarda), ularning qiymat diapozonlari va berilganlarni bir turdan ikkinchisiga o'tkazish bilan bog'liq masalalar qaraladi.

Jadvalning har bir satridagi "*Berilgan turi*" ustunidagi turda o'zgaruvchilar e'lon qilinib, ularga mos ravishda 8, 10, va 16 lik sanoq sistemasi ko'rinishidagi uchta qiymat klaviaturadan kiritilsin. Bu qiymatlar jadvalning keyingi ustunlarida keltirilgan turlardagi o'zgaruvchilarga o'zlashtirilsin va chop etilsin (agar mos satr va ustun kesishmasida '+' turgan bo'lsa). Chop etilgan natijalarga izoh berilsin.

### *Namunaviy masala*

short int turida berilgan o'zgaruvchining qiymatini unsigned char, unsigned short int, long int turidagi o'zgaruvchiga o'zlashtiring va ularning qiymatlariga izoh bering.

### *Yechish usuli*

Masalani yechishda C++ tilidagi bir turdan ikkinchi turga oshkor va oshkormas keltirishdan foydalaniladi.

### *Ilova matni*

```
#include<iostream.h>
int main()
{
    short int n;
    cout<<"n qiymatini kiriting =";
    cin>>n;
    unsigned char n1;
    unsigned short int n2;
    long int n3;
    n1=n; cout<<"n1="<<int(n1)<<endl;
    n2=n; cout<<"n2="<<n2<<endl;
    n3=n; cout<<"n3="<<n3<<endl;
    return 0;
}
```

Ilova ishga tushganda n ning qiymatini kiritish so'raladi va kiritilgan qiymat boshqa turdagi o'zgaruvchilarga o'zlashtiriladi. Hosil bo'lgan natijalar chop qilinadi.

Masalan, n o'zgaruvchining qiymati sifatida -1 soni kiritilsa, quyidagi natijalar chop etiladi:

n1=255  
n2=65535  
n3= -1

### Natijalarga izoh

- n=-1 sonining xotiradagi ichki ko‘rinishi "11111111 11111111" ko‘rinishida bo‘ladi.

- n1 o‘zgaruvchi xotirada 1 bayt joy egallaydi va u n o‘zgaruvchining kichik baytidagi qiymatni qabul qiladi (n1=11111111). Bu son ishorasiz butun son va o‘nlik sanoq sistemasida 255 soniga teng.

- n2 o‘zgaruvchi xotirada 2 bayt joy egallaydi va u n o‘zgaruvchining qiymatini to‘liq o‘zlashtiradi (n2=11111111 11111111). Bu son ishorasiz butun son va o‘nlik sanoq sistemasida 65535 soniga teng.

- n3 o‘zgaruvchi xotirada 4 bayt joy egallaydi va "n3=n;" ko‘rsatmasi bajarilishi natijasida n3 o‘zgaruvchida -1 sonining qo‘shimcha koddagi ko‘rinishi hosil bo‘ladi (n3=11111111 11111111 11111111 11111111).

### Amaliy topshiriqlar

Jadvalning "Berilgan turi" ustunida ko‘rsatilgan turdagi qiymatni O‘tkaziladigan turlar" turidagi qiymatga o‘tkazilsin va natija chop etilsin.

№	Berilgan turi	O‘tkaziladigan turlar										
		unsigned				short int	int	long int	float	double	bool	char
		short int	int	long int	char							
1	short int	+	+	-	+	-	+	-	+	-	+	+
2	int	+	+	+	+	+	-	-	+	+	+	+
3	long int	+	+	+	+	+	+	-	+	+	-	+
4	float	+	+	+	+	+	+	+	-	-	+	+
5	double	+	+	+	+	+	+	+	+	-	+	+
6	long double	+	+	+	+	+	+	+	+	+	-	+
7	char	+	+	-	+	+	-	-	+	-	+	-
8	unsigned short int	-	+	-	+	+	+	-	+	-	+	+
9	unsigned int	+	-	-	+	+	+	-	+	-	-	+
10	unsigned long int	+	+	-	+	+	+	+	+	+	-	+
11	unsigned char	+	+	-	-	+	-	-	+	-	+	+
12	bool	+	+	-	+	+	+	-	+	-	-	+

**Izoh.** Jadval kataklaridagi ‘+’ belgisi qiymatni o‘tkazish kerakligini, ‘-’ belgisi esa qiymatni o‘tkazish shart emasligini bildiradi.

## 2. Razryadlar ustida mantiqiy amallar

Berilgan  $a, b, n$  bayt kattaligidagi ishorali butun sonlar ustida quyidagi amallar bajarilsin:  $a \& b$ ,  $a | b$ ,  $a \wedge b$ ,  $\sim a \& b$ ,  $\sim b | a$ ,  $a = a \gg n$ ,  $a = a \ll n$ .

### *Namunaviy masala*

Bayt kattaligidagi ishorali butun  $a$  va  $b$  sonlar ustida razryadli mantiqiy ko'paytirish amali bajarilsin va natijaga izoh berilsin.

### *Yechish usuli*

Masalani yechishda C++ tilida aniqlangan mantiqiy razryadli amallarga tayaniladi.

### *Ilova matni*

```
#include <iostream.h>
int main()
{
char a=124, b=10;
cout<<"\na&b="<<int(a&b);
return 0;
}
```

Ilova ishlashi natijasida ekranga

8

qiymati chop etiladi.

### *Natijani izohlash*

Razryadli mantiqiy amallarning aniqlanish qoidasiga quyidagicha:

#### **A va B razryadlar ustida mantiqiy amallar jadvali**

A	B	A&B	A B	A^B	~A
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

Berilgan  $a$  va  $b$  o'zgaruvchilar ( $a=124$ ,  $b=10$ ) qiymatlarining bayt razryadlaridagi ikkilik ko'rinishi mos ravishda 01111100 va 00001010 ko'rinishda bo'ladi.

a :01111100 (124)

b :00001010 (10)

a&b:00001000 (8)

### *Amaliy topshiriqlar*

Operandlar	Variantlar									
	1	2	3	4	5	6	7	8	9	10
<b>A</b>	2	5	-10	64	28	55	10	-1	15	123
<b>B</b>	-3	125	10	-1	-16	-55	105	126	127	64

Operandlar	Variantlar									
	11	12	13	14	15	16	17	18	19	20
<b>A</b>	10	127	123	12	126	125	-12	60	20	32
<b>B</b>	-45	100	-100	-1	2	25	-100	15	-20	-32

### **3. Matematik ifodalarni hisoblash**

Amaliy topshiriqlarni bajarishda matematik funksiyalar va ifodalar ning C++ tilida yozilishi, butun va haqiqiy sonlar ustida arifmetik amallar bajarilishiga e'tibor berish kerak bo'ladi. Masalalar test ko'rinishida beriladi, ya'ni olingan natija oldindan berilgan qiymat bilan mos tushishi shart.

#### *Namunaviy masala*

Haqiqiy turdagi  $x$ ,  $y$  va  $z$  o'zgaruvchilarning  $x=182.5, y=18.225$  va  $z=-0.3298$  qiymatlarida

$$a = \left| x^{\frac{y}{x}} - \sqrt[3]{\frac{y}{x}} \right| + (y-x) \frac{\cos y - \frac{z}{(y-x)}}{1 + (y-x)^2}$$

ifoda hisoblanib, natijani 1.2132 qiymat bilan ustma-ust tushishi tekshirilsin.

#### *Yechish usuli*

Amallar ustunlik darajasiga rioya qilgan holda, matematik ifodalarni C++ tilida yozish qoidalari qo'llaniladi. Murakkab matematik ifodaning har bir ifoda osti alohida hisoblanib, oraliq o'zgaruvchilarga o'zlashtiriladi va ular asosida yakuniy natija olinadi. Bu usulning afzalligi shundaki, oraliq natijalar to'g'ri hisoblanganligini nazorat qilish, xatolarni lokallashtirish va ularni bartaraf etish nisbatan oson kechadi.

Standart matematik funksiyalar tilning <math.h> kutubxonasida joylashgan (2-ilova).

#### *Ilova matni*

```
#include<iostream.h>
#include<math.h>
int main()
{
```

```

float x,y,z,a,a1,a2,k,t;
x=182.5; y=18.225; z=-03.298;
k=y/x; t=y-x;
a1=fabs(pow(x,k)-pow(k,1./3));
a2=(cos(y)-z/t)/(1+pow(t,2));
a=a1+t*a2;
cout<<"a"<<a;
return 0;
}

```

Ilova bajarilishi natijasida ekranga quyidagi natija chop etiladi:

a=1.2132

### ***Amaliy topshiriqlar***

Quyidagi ifodalar berilgan qiymatlar uchun hisoblansin:

1. Berilganlar:  $x=14.26, y=-1.22,$   
 $z=3.5 \times 10^{-2},$

$$t = \frac{2 \cos\left(x - \frac{\pi}{6}\right)}{0.5 + \sin^2 y} \left(1 + \frac{z^2}{3 - z^2 / 5}\right)$$

Natija:  $t=0.564849$

3. Berilganlar:  $x=-4.5, y=0.75 \times 10^{-4},$   
 $z=0.845 \times 10^2,$

$$u = \frac{\sqrt[3]{8 + |x - y|^2 + 1}}{x^2 + y^2 + 2} - e^{|x-y|} (\operatorname{tg}^2 z + 1)^x.$$

Natija:  $u=-55.6848$

5. Berilganlar:  $x=-15.246, y=4.642 \times 10^{-2},$   
 $z=20.001 \times 10^2,$

$$\alpha = \ln\left(y^{-\sqrt{|x|}}\right) \left(x - \frac{y}{2}\right) + \sin^2 \operatorname{arctg}(z).$$

Natija:  $\alpha=-182.036$

7. Berilganlar:  $x=0.1722, y=6.33,$   
 $z=3.25 \times 10^{-4},$

$$\gamma = 5 \operatorname{arctg} x - \frac{1}{4} \operatorname{arccos} x \frac{x + 3|x - y| + x^2}{|x - y|z + x^2}.$$

Natija:  $\gamma=-205.306$

2. Berilganlar:  $x=3.74 \times 10^{-2},$   
 $y=-0.825, z=0.16 \times 10^2,$

$$v = \frac{1 + \sin^2(x + y)}{\left|x - \frac{2y}{1 + x^2 y^2}\right|} x^{|y|} + \cos^2\left(\operatorname{arctg} \frac{1}{z}\right)$$

Natija:  $v=1.0553$

4. Berilganlar:  $x=0.4 \times 10^4, y=-0.875,$   
 $z=-0.475 \times 10^{-3},$

$$w = |\cos x - \cos y|^{(1+2\sin^2 y)} \left(1 + z + \frac{z^2}{2} + \frac{z^3}{3} + \frac{z^4}{4}\right).$$

Natija:  $w=1.9873$

6. Berilganlar:  $x=16.55 \times 10^{-3}, y=-2.75,$   
 $z=0.15,$

$$\beta = \sqrt{10(\sqrt[3]{x} + x^{y+2})} (\arcsin^2 z - |x - y|).$$

Natija:  $\beta=-40.6307$

8. Berilganlar:  $x=-2.235 \times 10^{-2}, y=2.23,$   
 $z=15.221$

$$\varphi = \frac{e^{|x-y|} |x - y|^{x+y}}{\operatorname{arctg} x + \operatorname{arctg} z} + \sqrt[3]{x^6 + \ln^2 y}.$$

Natija:  $\varphi=39.374$



9. Berilganlar:  $x=1.825 \times 10^2, y=18.225,$   
 $z=-3.298 \times 10^{-2},$

$$\psi = \left| x^{\frac{y}{x}} - \sqrt[3]{\frac{y}{x}} \right| + (y-x) \frac{\cos y - \frac{z}{(y-x)}}{1 + (y-x)^2}.$$

Natija:  $\psi = 1.2131$

11. Berilganlar:  $x=6.251, y=0.827,$   
 $z=25.001,$

$$b = y^{\sqrt[3]{|x|}} + \cos^3 y \frac{|x-y| \left( 1 + \frac{\sin^2 z}{\sqrt{x+y}} \right)}{e^{|x-y|} + x/2}.$$

Natija:  $b=0.7121$

13. Berilganlar:  $x=17.421, y=10.365 \times 10^{-3},$   
 $z=0.828 \times 10^5,$

$$f = \frac{\sqrt[4]{y + \sqrt[3]{x-1}}}{|x-y|(\sin^2 z + \operatorname{tg} z)}.$$

Natija:  $f=0.33056$

15. Berilganlar:  $x=2.444, y=0.869 \times 10^{-2},$   
 $z=-0.13 \times 10^3,$

$$h = \frac{x^{y+1} + e^{y-1}}{1 + x|y - \operatorname{tg} z|} \left( 1 + |y-x| \right) + \frac{|y-x|^2}{2} - \frac{|y-x|^3}{3}.$$

Natija:  $h=-0.49871$

17. Berilganlar:  $x=1, y=1, z=3$

$$a = (1+y) \frac{x+y/(x^2+4)}{e^{-x-2} + 1/(x^2+4)};$$

$$b = \frac{1 + \cos(y-2)}{x^4/2 + \sin^2 z}.$$

Natija:  $a=9.608184; b=2.962605$

10. Berilganlar:  
 $x=3.981 \times 10^{-2},$   
 $y=-1.625 \times 10^3, z=0.512.$

$$a = 2^{-x} \sqrt{x + \sqrt[4]{|y|}} \sqrt[3]{e^{x-1/\sin z}}.$$

Natija:  $a=1.26185$

12. Berilganlar:  $x=3.251, y=0.325,$   
 $z=0.466 \times 10^{-4},$

$$c = 2^{(y^x)} + (3^x)^y - \frac{y \left( \operatorname{arctg} z - \frac{\pi}{6} \right)}{|x| + \frac{1}{y^2 + 1}}.$$

Natija:  $c=4.025$

14. Berilganlar:  $x=12.3 \times 10^{-1}, y=15.4,$   
 $z=0.252 \times 10^3,$

$$g = \frac{y^{x+1}}{\sqrt[3]{|y-2|} + 3} + \frac{x+y}{2|x+y|} (x+1)^{-1/\sin z}.$$

Natija:  $g=82.8257$

16. Berilganlar:  $x=-1, y=-1, z=3$

$$a = \left( \sqrt{|x-1|} - \sqrt[3]{|y|} \right) / \left( 1 + \frac{x^2}{2} + \frac{y^2}{4} \right);$$

$$b = x(\operatorname{arctg}(z) + e^{-(x+3)}).$$

Natija:  $a=0.2366935; b=-1.384381$

18. Berilganlar:  $x=3, y=4, z=5$

$$a = \frac{2 \cos(x - \pi/6)}{1/2 + \sin^2 y};$$

$$b = 1 + \frac{z^2}{3 + z^2/5}.$$

Natija:  $a=-1.467187; b=4.125$

19. Berilganlar:  $x=3, y=4, z=5,$

$$a = \frac{1 + \sin^2(x + y)}{2 + \left| x + 2x / (1 + x^2 y^2) \right|} + x;$$

$$b = \cos^2\left(\arctg \frac{1}{z}\right).$$

Natija:  $a=3.288716; b=0.9615385$

20. Berilganlar:  $x=5, y=4.5, z=0.5,$

$$a = \frac{\sin(xy - e^x)^2}{1 + \frac{x}{y} 2.05 + 0.001e^{x^2}};$$

$$b = \sqrt{x^2 - y^2} \ln z + \frac{\lg(y + z)}{\sqrt{x^2 - y^2 + 1}}.$$

Natija:  $a=8.46425e-10, b= -1.29084$

## 4. Hisoblashga doir sodda masalalar

### *Namunaviy masala*

O‘zaro teng bo‘lmagan  $x$  va  $y$  o‘zgaruvchilar berilgan. Ularning qiymatlari qo‘shimcha o‘zgaruvchidan foydalanmagan holda almashtirilsin.

### *Yechish usuli*

Berilgan  $x$  va  $y$  o‘zgaruvchilar ustida qo‘shish va ayirish amallarini ketma-ket bajarish orqali ularning qiymatlari o‘zaro almashtiriladi.

### *Ilova matni*

```
#include<iostream.h>
#include<math.h>
int main()
{
float x,y,z;
cout<<"x="; cin>>x; // masalan x=5;
cout<<"y="; cin>>y; // masalan y=7;
x+=y; // 12
y=x-y; // 5
x-=y; // 7
cout<<"O‘zaro almashgan qiymatlar: ";
cout<<"x="<<x<<', '<<"y="<<y;
return 0;
}
```

Agar  $x$  va  $y$  o‘zgaruvchilarning qiymati sifatida 5 va 7 sonlari kiritilsa, ekranga ularning almashgan natijasi chop etiladi:

O‘zaro almashgan qiymatlar:  $x=7, y=5.$

### *Amaliy topshiriqlar*

1.  $x^x$  -funksiyasining hosilasi  $a(a>0)$  nuqtada hisoblansin.
2. Berilgan  $a$  soni uchun  $\ln(\text{ctgx}-1)=a$  tenglamaning  $(\pi, 2\pi)$  oraliqdagi yechimi topilsin.

3. Berilgan koeffisient va o'ng tomon qiymatlarida tenglamalar sistemasi-ning determinanti noldan farqli deb hisoblab sistema yechimi topilsin:

$$\begin{cases} a_1x + b_1y = c_1 \\ a_2x + b_2y = c_2. \end{cases}$$

4. Berilgan  $R$  radiusli aylananing uzunligi, doiraning yuzasi va sharning hajmi hisoblansin.
5. Uchburchak uchlarining koordinatalari asosida uning yuzasi va perimetri topilsin.
6. Berilgan to'rt xonali son raqamlarining ko'paytmasi topilsin.
7. Massalari mos ravishda  $m_1$  va  $m_2$  bo'lgan va bir biridan  $r$  masofada joylashgan ikkita jismning o'zaro tortishish kuchi  $F$  aniqlansin.
8. To'g'ri burchakli uchburchakning gipotenuzasi va kateti berilgan bo'lsa, ikkinchi katet va ichki chizilgan aylananing radiusi hisoblansin.
9. Aylana uzunligi berilgan. Bu aylana bilan chegaralangan doira yuzasi topilsin.
10. Uchburchak burchaklarining kattaliklari va ichki chizilgan aylana radiusi bilan berilgan. Uchburchak tomonlarining uzunliklari hisoblan-  
sin.
11. Uchburchak tomonlari berilgan. Quyidagilar aniqlansin:  
a) balandliklari;    b) medianalari;  
e) bissektisarlari;    d) ichki va tashqi chizilgan aylana radiuslari.
12. Butun turdagi  $h$  o'zgaruvchiga berilgan  $k$  ( $k \geq 100$ ) musbat sonning oxir-  
idan uchinchi raqami o'zlashtirilsin (masalan, agar  $k=130985$  bo'lsa,  $h=9$ ).
13. Butun turdagi  $d$  o'zgaruvchiga haqiqiy musbat  $x$  sonining kasr qismin-  
ing birinchi raqami o'zlashtirilsin (masalan, agar  $x=32.597$  bo'lsa,  $d=5$ ).
14. Agar hozir sutkaning  $k$ -sekundi bo'lsa, sutka boshidan o'tgan aniq vaqt  
– soat( $h$ ), minut( $m$ ) va sekund( $s$ ) hisoblansin (masalan, agar  
 $k=13257=3 \cdot 3600 + 40 \cdot 60 + 57$  bo'lsa,  $h=3$ ,  $m=40$  va  $s=57$ ).
15. Soat milining sutka boshidagi holati bilan  $h$  soat,  $m$  minut va  $s$   
sekunddagi holatlari orasidagi  $f$  burchak (graduslarda) aniqlansin  
( $0 \leq h \leq 11$ ,  $0 \leq m, s \leq 59$ ).
16. Soat milining sutka boshiga nisbatan  $f$  gradus burilishiga mos keluvchi  
 $h$  soat,  $m$  minut va  $s$  sekund topilsin ( $0 \leq f \leq 360^\circ$ ,  $f$  - haqiqiy son).
17. Birinchi yanvar kuni dushanba bo'lib, kabisa bo'lmagan biror yilning  $k$ -  
kuni ( $1 \leq k \leq 365$ ) haftaning qaysi kuniga to'g'ri kelishi aniqlansin va  
bu qiymat  $n$  butun o'zgaruvchiga o'zlashtirilsin ( $1 \leq n \leq 7$ ).
18. Og'irligi bir kilogramm bo'lgan mahsulotning narxi berilgan. Uning  
og'irligi grammlarda kiritilsin va to'lash zarur bo'lgan pul miqdori

chop etilsin.

19. 10 metr radiusli silindrik shaklga ega bo'lgan suv bosimi minorasidagi suv sathining balandligi berilgan bo'lsa, suvning hajmi hisoblansin.
20. Bolalar bog'chasiga bir oylik to'lov 70000 so'm (bir oy - 22 kun). Agar bola bog'chaga  $N(0 < N < 23)$  kun kelmagan bo'lsa, bir oy uchun qancha to'lash kerak bo'ladi?
21.  $R$  radiusli doiraga ichki chizilgan muntazam  $n$ -burchakning perimetri va yuzasi hisoblansin.
22. Turg'un suvdagi qayiq tezligi  $V$  km/s. Daryo suvi oqimining tezligi  $U$  km/s ( $U < V$ ). Qayiq ko'lda  $T_1$  soat, daryoda esa (oqimga qarshi)  $T_2$  soat harakat qilgan. Qayiq suzgan umumiy  $S$  masofa topilsin.
23. Birinchi avtomobil tezligi  $V_1$  km/s, ikkinchisini  $V_2$  km/s, ular orasidagi masofa -  $S$  km. Avtomobillar bir-biridan uzoqlashsa (bir-biriga qarab harakat qilganda),  $T$  soatdan keyin ular orasidagi masofa qanday bo'ladi?
24. Asoslari  $a$  va  $b$  ( $a > b$ ), katta asosdagi burchagi  $\alpha$  bo'lgan teng yonli trapetsiyaning perimetri va yuzasi topilsin (burchak radianda beriladi).
25. Noldan farqli berilgan  $R_1, R_2, R_3$  elektr qarshiliklari uchun  $R$  hisoblansin. Bunda: 
$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2} + \frac{1}{R_3}.$$
26. Xodimning oylik ish haqiga 45% mukofot puli qo'shilsin. Hosil bo'lgan miqdordan 17% daromad solig'i, 1,5% kasaba uyushmasi va 1% nafaqa solig'i ushlab qolinsin. Qo'lga tegadigan pul miqdori chop etilsin.
27. Uch xonali butun son ( $k$ ) raqamlari yig'indisini ( $s$ ) butun o'zgaruvchiga o'zlashtirilsin.
28. Teng tomonli uchburchak tomoni berilgan, bo'lsa uchburchak yuzasi topilsin.
29. Uchta musbat son berilgan. Sonlar o'rta geometrigining kasr qismi topilsin.
30. Berilgan katetlari bo'yicha to'g'ri burchakli uchburchakning perimetri va yuzasi hisoblansin.
31. Berilgan ikki tomoni va ular orasidagi burchak (gradusda) asosida uchburchakning uchinchi tomoni va yuzasi topilsin.
32. Berilgan uch xonali son raqamlarini teskari tartibda yozishdan hosil bo'lgan son topilsin. Masalan, 345 sonining teskari tartibi 543 bo'ladi.
33. Soat milining ko'rsatgichi  $a$  ( $0 \leq a \leq 11$ ) soatda bo'lib, unga  $b$  soat (butun soat) qo'shilgandan (ayrilgandan) keyingi holati chop etilsin.

## 5. Mantiqiy ifodalar

### *Namunaviy masala*

Berilgan uch xonali butun sonning raqamlari o‘zaro teng yoki yo‘qligi aniqlansin.

### *Yechish usuli*

Masala C++ tilining butun sonlar ustidagi arifmetik amallardan foydalanilgan holda yechiladi. Berilgan butun  $a$  va  $b$  sonlar uchun  $/$  amali  $a/b$  bo‘linmaning butun qismini,  $\%$  amali  $a\%b$  bo‘linmaning butun qoldiqini beradi. Bu bo‘lishlardan foydalanib, berilgan sonning raqamlarini ajratib olish va ularni o‘zaro solishtirish mumkin.

### *Ilova matni*

```
#include <iostream.h>
int main()
{
    unsigned int n,a2,a1,a0; // n = a2a1a0 ko‘rinishida
    cout<<"\nn - qiymatini kiriting: ";
    cin>>n;
    if(n<100||n>999)
    {
        cout<<"Kiritilgan son 3 xonali emas!";
        return 1;
    }
    a2=n/100;
    a1=(n%100)/10;
    a0=n%10;
    cout<<"Berilgan son raqamali o‘zaro teng";
    if(a2!=a1 && a1!=a0 && a2!=a0) cout<<"!";
    else cout<<" emas!";
    return 0;
}
```

### *Amaliy topshiriqlar*

1. Ifodaning qiymati topilsin:

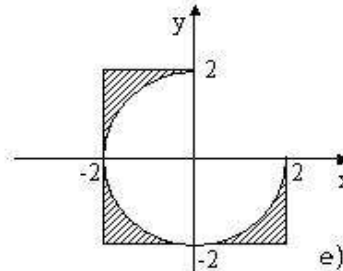
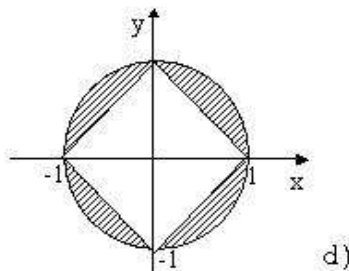
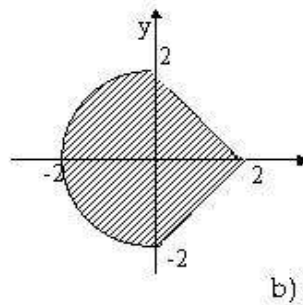
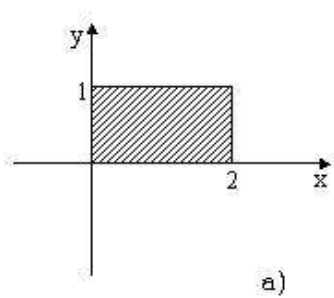
a) agar  $x=0.3$ ,  $y=-1.6$  bo‘lsa  $x^2 + 2y \leq 4$ ;

b) agar  $k=15$  bo‘lsa,  $\left\{ \frac{k}{7} \right\} = \frac{k}{5} - 1$ . Bu yerda “ $\{ \}$ ” – bo‘linma qoldig‘i;

d) agar  $p=0.182$  bo‘lsa,  $\left\{ \frac{10-p}{2} \right\} = 0$ .

2. Quyidagi shartlar bajarilganda “rost”, aks holda “yolg‘on” qiymat qabul qiluvchi mantiqiy munosabatlar C++ tilida yozilsin:
- a)  $k$  butun soni 7 ga bo‘linadi;
  - b)  $ax^2+bx+c=0$  tenglama haqiqiy ildizlarga ega emas;
  - d)  $(x,y)$  nuqta, markazi  $(1,0)$  nuqtada bo‘lgan  $r$  radiusli aylananing tashqarisida yotadi;
  - e)  $n$  natural soni – to‘liq kvadrat.
3. Ifodalarning qiymatlari hisoblansin:
- a)  $\neg \left\{ \frac{n}{2} \right\}$ , agar  $n=15$ ;
  - b)  $t \wedge \left\{ \frac{n}{2} \right\} = 0$ , agar  $t=true$ ,  $n=101010$ ;
  - d)  $(x+y \neq 0) \wedge (y > x)$ , agar  $x=2$ ,  $y=1$ ;
  - e)  $(x+y \neq 0) \vee (y > x)$ , agar  $x=2, y=1$ ;
  - f)  $a \vee \neg b$ , agar  $a=false$ ,  $b=true$ .
4. Quyidagi shartlar bajarilganda rost, aks holda yolg‘on qiymat qabul qiluvchi mantiqiy munosabatlar C++ tilida yozilsin:
- a)  $0 < x < 1$ ;
  - b)  $x = \max(x, y, z)$ ;
  - d)  $x \neq \max(x, y, z)$  (inkor amali qo‘llanilmasin);
  - e)  $a$  va  $b$  mantiqiy o‘zgaruvchilardan kamida bittasi true;
  - f) har ikkala  $a, b$  mantiqiy o‘zgaruvchilar qiymatlari true.
5. Ayniyatlar isbotlansin:
- a)  $a \wedge \neg a \equiv false$ ;
  - b)  $a \vee \neg a \equiv true$ ;
  - d)  $\neg \neg a \equiv a$ ;
  - e)  $true \vee a \equiv true$ ;
  - f)  $false \wedge a \equiv false$ ;
  - g)  $a \vee a \equiv a$ .
6. Hisoblansin:
- a)  $false \vee (1/1 > 0)$ ;
  - b)  $(1/2 > 0) \wedge true$ .
7. Ifodalardagi amallar bajarilish tartibi ko‘rsatilsin:
- a)  $a \&\& b \|| c \&\& d$ ;
  - b)  $(x >= 0) \|| t \&\& x \% 2 == 0 \|| (y * y != 4)$ .
8. Mantiqiy  $a$  va  $b$  o‘zgaruvchilarning  $a=true$  va  $b=false$  qiymatlari uchun quyidagi ifodalar hisoblansin:
- a)  $a \|| b \&\& !a$ ;
  - b)  $(a \|| b) \&\& !a$ ;
  - d)  $!a \&\& b$ ;
  - e)  $!(a \&\& b)$ .
9. Quyidagi shartlar bajarilganda true, aks holda false qiymat qabul qiluvchi ifodalar C++ tilida yozilsin:
- a)  $x \in [0, 1]$ ;
  - b)  $x \notin [0, 1]$ ;
  - d)  $x \in [2, 5] \vee x \in [-1, 1]$ ;
  - e)  $x \notin [2, 5] \vee x \in [-1, 1]$ .
  - f)  $x, y$  va  $z$  sonlaridan har biri musbat;
  - g)  $x, y$  va  $z$  sonlaridan hech bo‘lmaganda biri musbat;
  - h)  $x, y$  va  $z$  sonlaridan hech biri musbat emas;
  - i)  $x, y$  va  $z$  sonlaridan faqat biri musbat;
  - j) mantiqiy o‘zgaruvchi  $a$  true,  $b$  esa false qiymatini qabul qilgan holda;

- k)  $y$  – yil kabisa yili (kabisa yili 4 ga karrali yillar hisoblanadi. Biroq, 100 ga karrali yillar orasida faqat 400 ga karrali yillar kabisa yili deyiladi. Masalan, 1700, 1800, 1900-oddiy yillar, 2000-kabisa yili).
10. Quyidagi shartlarga mos keluvchi soha tekislikda tasvirlansin:
    - a)  $(y > x) \&\& (y + x >= 0) \&\& (y <= 1)$ ;    b)  $(x * x + y * y < 1) \|\| (y > 0) \&\& (y <= 1)$ .
  11. Ifodaning qiymati hisoblansin:
    - a) `false < true`;    b) `(32 \|\| false) == 1`;    d) `9 + 3 * true`;    e) `16 + true / 2`.
  12. Ifodaning qiymati hisoblansin:
    - a) `!(--s) \&\& ((int)(s) == 1)`, agar `s = true`;
    - b) `(p < true) == (q == false)`, agar `p = q = true`;
    - d) `a \&\& b > a \|\| b`, agar `a = true`, `b = false`.
  13. Quyidagi shartlar bajarilganda rost, aks holda yolgʻon qiymat qabul qiladigan ifoda C++ tilida yozilsin:
    - a) butun  $n$  va  $m$  sonlari bir paytda toq yoki juft sonlar;
    - b)  $a$  va  $b$  mantiqiy oʻzgaruvchilardan faqat bittasi true qiymatiga ega;
    - d)  $a, b$  va  $c$  mantiqiy oʻzgaruvchilardan faqat bittasi true qiymatini qabul qiladi.
  14. Quyidagi ifodalar oʻrinli boʻladigan soha XOY tekisligida chizilsin:
    - a)  $(\text{fabs}(x) <= 1) == (\text{fabs}(y) >= 1)$ ;    b)  $(x * x + y * y <= 4) == (y <= x)$ .
  15. Ayniyatlar isbotlansin:
    - a)  $!(a \|\| b) \equiv (!a) \&\& (!b)$ ;    b)  $a \&\& (b \|\| c) \equiv (a \&\& b) \|\| (a \&\& c)$ ;
    - d)  $a <= b \equiv !a \|\| b$ ;    e)  $a \&\& b \equiv (a < true) < b$ ;    f)  $!a \equiv a < true$ .
  16. Mantiqiy  $a$  va  $b$  oʻzgaruvchilar uchun quyidagi ifodalarni taqqoslash amallari qatnashmagan koʻrinishga keltirilsin:
    - a)  $a < b$ ;    b)  $a == b$ ;    d)  $(a < b) == a$ .
  17. Agar  $a = true$  va  $x = 1$  boʻlsa, quyidagi mantiqiy  $d$  oʻzgaruvchi qanday qiymat qabul qiladi?
    - a)  $d = x < 2$ ;    b)  $d = !a \|\| x \% 2$ ;    d)  $d = a \% 2 != x$ .
  18. Quyidagi shart bajarilganda  $t$  mantiqiy oʻzgaruvchisi true, aks holda false qiymatini oʻzlashtirsin:
    - a)  $x, y$  va  $z$  sonlar oʻzaro teng;
    - b)  $x, y$  va  $z$  sonlardan faqat ikkitasi oʻzaro teng;
    - d)  $x$  musbat son;
    - e)  $p$  soni  $q$  ga qoldiqsiz boʻlinadi ( $p$  va  $q$ -natural sonlar);
    - f)  $ax^2 + bx + c = 0$  tenglama bitta yechimga ega, bu yerda  $a, b$  va  $c$  oʻzgaruvchilar 0 boʻlishi mumkin;
    - g) uch xonali butun  $k$  sonining oʻnli yozuviga '5' raqami kiradi.
  19. Agar  $(x, y)$  nuqta boʻyalgan sohaga tegishli boʻlsa,  $t$  mantiqiy oʻzgaruvchi true qiymatini qabul qiladigan ifoda yozilsin (rasmga qarang).



20. Shaxmat taxtasining kataklari  $(x,y)$  butun sonlar juftligi bilan beriladi  $(1 \leq x, y \leq 8)$ . Berilgan  $(h_1, v_1)$  va  $(h_2, v_2)$  kataklar uchun quyidagi mu-lohazalar mantiqiy ifoda ko‘rinishida yozilsin:

- a) kataklari bir xil rangga ega;
- b)  $(h_1, v_1)$  katagida joylashgan «ot»  $(h_2, v_2)$  katakka xavf soladi;
- d)  $(h_1, v_1)$  katagida joylashgan «farzin»  $(h_2, v_2)$  katakka xavf soladi.

## 6. '?' shart amali

Quyidagi misollar '?' shart operatori yordamida yechilsin. Berilgan haqiqiy  $x$  soni uchun  $y$  hisoblansin.

### Namunaviy masala

Berilgan haqiqiy  $x$  soni uchun  $y$  qiymati hisoblansin.

$$y = \begin{cases} x^2, & \text{agar } -3 \leq x < 3; \\ x^2 + 4x, & \text{agar } x < -3; \\ 3 - x, & \text{aks holda.} \end{cases}$$

### Yechish usuli

Hisoblash shartlari nisbatan sodda bo‘lganligi sababli interval ko‘rinishida berilgan funksiya qiymatini ichma-ich joylashgan '?' shart amalini qo‘llash orqali yechiladi.

### Ilova matni

```
#include<iostream.h>
#include<math.h>
int main()
```



```

{
float y,x;
cout<<"x="; cin>>x;
y=(x>=3)?3-x: // x>=3 bo'lsa y=3-x aks holda
(x>=-3&&3>x)?x*x:x*x+4*x; // agar -3<=x<3 bo'lsa y=x*x, aks holda y=x*x+4*x;
cout<<"y="<<y;
return 0;
}

```

### *Amaliy topshiriqlar*

$$1. \ y = \begin{cases} 5x^3 + 6x^2 - 2x, & |x| \leq 1; \\ \frac{x+3}{|x|}, & x > 1; \\ 0, & x < -1. \end{cases}$$

$$2. \ y = \begin{cases} x^2, & -2 \leq x < 2; \\ x^2 + 4x + 5, & x < -2; \\ 2 - x, & x \geq 2. \end{cases}$$

$$3. \ y = \begin{cases} 0, & x \leq 0; \\ x, & 0 < x \leq 1; \\ x^4, & x > 1. \end{cases}$$

$$4. \ y = \begin{cases} \sin x + \sqrt{|x-5|}, & x < 5; \\ 5.45^2 \cos \pi + \ln x, & x = 5; \\ (x-5)^2 \operatorname{tg} \frac{x}{2}, & x > 5. \end{cases}$$

$$5. \ y = \begin{cases} -\frac{1}{x^2}, & x \leq -1; \\ x^2, & -1 < x \leq 2; \\ 4, & x > 2. \end{cases}$$

$$6. \ y = \begin{cases} \operatorname{arctg} x + \sqrt{|x-2|}, & x < -1; \\ 2.71^2 \cos \pi x, & -1 \leq x \leq 1; \\ \ln(x-1) \sin \frac{\pi x}{2}, & x > 1. \end{cases}$$

$$7. \ y = \begin{cases} \operatorname{ctg} x + 3.215 \sqrt{|x+2|}, & x < -2; \\ 13.85^2 \cos \pi, & -2 \leq x \leq 5; \\ (x-2)^{\frac{2}{3}} \sin \frac{\pi x}{2}, & x > 5. \end{cases}$$

$$8. \ y = \begin{cases} \sin x + \sqrt{|x+2|}, & x < -2; \\ x^2 \cos \pi + \ln(x+6), & -2 \leq x \leq 0; \\ (x+5)^{\frac{1}{3}} \operatorname{tg} \frac{x}{2}, & x > 2. \end{cases}$$

$$9. \ y = \begin{cases} \operatorname{tg} x + \sqrt{1+|x|}, & x < -5; \\ 6.78^{\frac{3}{4}} + 2x + \sin x, & -5 \leq x \leq 2; \\ 3x + \sin \pi x, & x > 2. \end{cases}$$

$$10. \ y = \begin{cases} \operatorname{ctg} x + \sqrt{1+|x-2|}, & x < 0; \\ (5.12+x)^{\frac{1}{3}} + \sin \pi, & 0 \leq x \leq 1; \\ 3x + \sin \pi x, & x > 1. \end{cases}$$

$$11. \ y = \begin{cases} x \operatorname{tg} x + \sqrt{\ln|x-2|}, & x < -2; \\ (x^2+3)x, & -2 \leq x \leq 2; \\ (x-2) \sin \frac{\pi x}{2}, & x > 2. \end{cases}$$

$$12. \ y = \begin{cases} \operatorname{ctg} x + \sqrt{3+|x|}, & x < -5; \\ (3x+1)^2 + \sin \pi, & -5 \leq x \leq 3; \\ (x-3) + \sin \pi x, & x > 3. \end{cases}$$

13.  $a = 1.5;$

$$y = \begin{cases} \operatorname{tg}x + \sqrt{\ln|a-3|}, & x < -3; \\ (a^2 - 3)^2 - \sin 2x, & -3 \leq x \leq 3; \\ (a+3) - \cos \pi x, & x > 3. \end{cases}$$

15.  $a = 2; b = .5;$

$$y = \begin{cases} 1, & x < 1; \\ ax^2 \ln x, & 1 \leq x \leq 2; \\ e^{ax} \cos bx, & x > 2. \end{cases}$$

17.  $a = 2.8; b = -0.3; c = 4;$

$$y = \begin{cases} ax^2 + bx + c, & x < 1.2; \\ a/x + \sqrt{x^2 - 1}, & x = 1.2; \\ (a + bx)/\sqrt{x^2 + 1}, & x > 1.2. \end{cases}$$

19.  $y = \begin{cases} 1.5 \cos^2 x, & x < 1; \\ (x-2)^2 + 6, & 1 \leq x \leq 2; \\ 3 \operatorname{tg}x, & x > 2. \end{cases}$

14.  $a = 1.5;$

$$y = \begin{cases} \pi x^2 - 7/x^2, & x < 1.3; \\ ax^3 + 7\sqrt{x}, & x = 1.3; \\ \operatorname{tg}(x + 7\sqrt{x}), & x > 1.3 \end{cases}$$

16.  $a = 1.65; b = 1.1$

$$y = \begin{cases} \pi x^2 - 7/x^2, & x < 1.4; \\ ax^3 + 7\sqrt{x^2 - 1}, & x = 1.4; \\ (a + bx)/\sqrt{x^2 + 1}, & x > 1.4. \end{cases}$$

18.  $a = 2; b = 0.5;$

$$y = \begin{cases} 1, & x < 1; \\ ax^2 \ln x, & 1 \leq x \leq 2; \\ e^{ax} \cos bx, & x > 2. \end{cases}$$

20.  $y = \begin{cases} \frac{x}{n}, & 0 \leq x \leq n; \\ n^2 \left( \frac{x}{2} - \sin x \right), & x > n; \\ \cos(nx), & x < 0. \end{cases} \quad n = 10.$

## 7. if-else, goto operatorlari

### Namunaviy masala

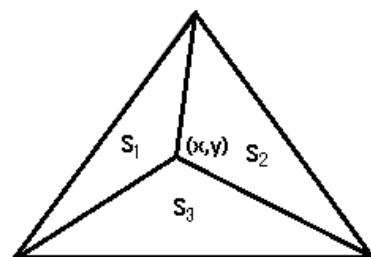
Haqiqiy turdagi  $x_1, y_1, x_2, y_2, x_3, y_3$  sonlari berilgan. Koordinata markazi uchlari  $(x_1, y_1), (x_2, y_2)$  va  $(x_3, y_3)$  nuqtalarda bo'lgan uchburchakka tegishlimi?

### Yechish usuli

Birorta  $(x, y)$  nuqta uchburchak ichiga tegishli bo'ladi, agar bu nuqtaN-ing uchburchak uchlarning juftliklari bilan hosil qiluvchi uchburchaklari yuzalarining  $(s_1, s_2$  va  $s_3)$  yig'indisi shu uchburchak yuzasiga  $(s)$  teng bo'lsa:  $s = s_1 + s_2 + s_3$ .

Shuni qayd etish kerakki, ikkita haqiqiy turdagi qiymatlarning o'zaro tengligini tekshirish ular ayirmasining absolyut qiymati berilgan aniqlikdan kichikligi orqali aniqlanadi.

Uchburchakning yuzasi Geron formulasi yordamida topiladi:



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

bu yerda a,b,c - uchburchak tomonlarining uzunliklari, p - uchburchak perimetrining yarmi.

### ***Hlova matni***

```
int main()
{
float x1,y1, x2,y2, x3,y3; // uchburchak uchlarining koordinatalari
float a,b,c ;           // uchburchak tomonlari
float s1,s2,s3;        // ichki uchburchaklar yuzalari
float s, p;            // uchburchak yuzasi va uchburchak perimetrining yarmi
const float eps=0.0001; // tekshirish aniqligi
cout<<"\n1-nuqta koordinatasini kiriting(x1,y1): ";
cin>>x1>>y1;
cout<<"\n2-nuqta koordinatasini kiriting(x2,y2): ";
cin>>x2>>y2;
cout<<"\n3-nuqta koordinatasini kiriting(x3,y3): ";
cin>>x3>>y3;
//{(x1,y1),(x2,y2),(x3,y3)} uchburchak yuzasini hisoblash(s)
a=sqrt((x1-x2)*(x1-x2)+(y1-y2)*(y1-y2));
b=sqrt((x2-x3)*(x2-x3)+(y2-y3)*(y2-y3));
c=sqrt(pow(x1-x3,2)+pow(y1-y3,2));
p=(a+b+c)/2;
s=sqrt(p*(p-a)*(p-b)*(p-c));
//{(0,0),(x1,y1),(x3,y3)} uchburchak yuzasini hisoblash (s1)
a=sqrt(x1*x1+y1*y1);
b=sqrt(x3*x3+y3*y3);
p=(a+b+c)/2;
s1=sqrt(p*(p-a)*(p-b)*(p-c));
//{(0,0),(x2,y2),(x3,y3)} uchburchak yuzasini hisoblash (s2)
a=sqrt(x2*x2+y2*y2);
c=sqrt(pow(x2-x3,2)+pow(y2-y3,2));
p=(a+b+c)/2;
s2=sqrt(p*(p-a)*(p-b)*(p-c));
//{(0,0),(x1,y1),(x2,y2)} uchburchak yuzasini hisoblash (s3)
b=sqrt(x1*x1+y1*y1);
c=sqrt(pow(x2-x1,2)+pow(y2-y1,2));
p=(a+b+c)/2;
s3=sqrt(p*(p-a)*(p-b)*(p-c));
// s=s1+s2+s3 shartni tekshirish. Bunda tenglikka tekshirish
```

```

// qiymatlarni ayirmasining absolyut qiymatini nolga
// yaqinligi bilan almashtiriladi, chunki haqiqiy sonlarni
// ustida amallar bajarilganda aniqlik yo'qotilishi mumkin
if(fabs(s-(s1+s2+s3))<eps) cout<<"Koordinata markazi uchburchak ichida.";
else cout<<"Koordinata markazi uchburchak ichida emas.";
return 0;
}

```

Ilova bajarilganda, uchlarining koordinatalari  $(-4,-1),(4,-3),(2,3)$  bo'lgan uchburchak kiritilsa, ekranga

Koordinatalar markazi uchburchak ichida.

xabari chop etiladi.

### *Amaliy topshriqlar*

1. Agar tomonlarining uzunliklari ixtiyoriy  $a$ ,  $b$  va  $c$  sonlarga teng bo'lgan uchburchakni qurish mumkin bo'lmasa  $0$ , aks holda – uchburchak teng tomonli bo'lsa  $3$ , teng yonli bo'lsa  $2$  va boshqa hollar uchun  $1$  qiymatini chop qiluvchi ilova tuzilsin.
2. Agar uchta haqiqiy, o'zaro teng bo'lmagan  $x, y$  va  $z$  sonlar yig'indisi  $1$  dan kichik bo'lsa, uchta sonning eng kichigi qolganlari yig'indisining yarmisi bilan almashtirilsin, aks holda  $x$  va  $y$  lardan kichigi qolganlari yig'indisining yarmi bilan almashtirilsin.
3. Berilgan  $50$  ta haqiqiy sonlarning eng kattasi topilsin.
4. Haqiqiy  $x, y$  va  $z$  sonlar berilgan bo'lsa, quyidagilar aniqlansin:
  - a)  $\max(x, y, z)$ ;
  - b)  $\max(x, y) + \min(y, z)$ ;
  - d)  $\max(x + y + z, x * y * z)$ ;
  - e)  $\min((x + y + z) / 2, x * z + 1)$ .
5. Uchta  $x$ ,  $y$  va  $z$  haqiqiy sonlar berilgan, agar ular monoton bo'lsa, ularning qiymatlari ikkilantirilsin, aks holda har bir o'zgaruvchining ishorasi qarama-qarshisiga almashtirilsin.
6. Butun  $n$  ( $n > 0$ ) va  $n$  ta haqiqiy sonlar berilgan. Ular orasidan manfiylari nechtaligi aniqlansin.
7.  $Ox$  va  $Oy$  o'qlarida yotmaydigan nuqta koordinatalari bilan berilgan. Bu nuqta joylashgan koordinata choragi aniqlansin.
8. Bo'sh bo'lmagan va oxiri  $0$  soni bilan tugaydigan musbat butun sonlar ketma-ketligi berilgan ( $0$  soni ketma-ketlikka kirmaydi va uning tugaganligini bildiradi). Ketma-ketlikning o'rta geometrik qiymati hisoblansin.
9. Sonlar o'qida uchta  $A$ ,  $B$  va  $C$  nuqtalar joylashgan.  $B$  va  $C$  nuqtalardan qaysi biri  $A$  nuqtaga yaqin masofada joylashgan bo'lsa, shu masofa chop etilsin.
10. Berilgan uch xonali son raqamlari orasida bir xillari bor yoki yo'qligi

aniqlansin.

11. Berilgan  $x$  uchun quyidagi ifodalarning qiymatlari o‘shish tartibida chop etilsin:  $chx$ ,  $1+|x|$  va  $(1+x^2)^x$ .
12.  $a_1x + b_1y = c_1$  va  $a_2x + b_2y = c_2$  tenglamalar bilan berilgan chiziqlarning kesishishi, ustma-ust tushishi, yoki o‘zaro paralleligi aniqlansin. Bu yerda  $a_1, b_1, c_1, a_2, b_2$  va  $c_2$  – berilgan sonlar.
13.  $ax^4 + bx^2 + c = 0$  tenglamaning haqiqiy ildizlari topilsin yoki ildizlar yo‘qligi haqida xabar berilsin.
14. Shaxmat taxtasidagi maydonlar sakkizdan katta bo‘lmagan sonlar juftligi bilan aniqlanadi: birinchi son shaxmat taxtasi maydonining vertikal bo‘yicha tartib nomeri (chapdan o‘nga), ikkinchisi – gorizontal bo‘yicha tartib nomeri (pastdan yuqoriga). Sakkizdan katta bo‘lmagan  $k, l, m$  va  $n$  sonlari berilgan. Quyidagi masalalar yechilsin:
  - a)  $(k, l)$  maydonidagi ruh bir yurishda  $(m, n)$  maydoniga o‘tishi mumkinmi? Agar mumkin bo‘lmasa, ikkita yurishda o‘tish yo‘llari ko‘rsatilsin;
  - b)  $(k, l)$  maydonidagi farzin bir yurishda  $(m, n)$  maydoniga o‘tishi mumkinmi? Agar mumkin bo‘lmasa, ikkita yurishda o‘tish yo‘llari ko‘rsatilsin.
15. Berilgan  $a_1, a_2, a_3$  va  $a_4$  butun sonlar ichida uchta bir-biriga teng. Boshqalaridan farqli bo‘lgan sonning tartib nomeri chop etilsin.
16. Butun turdagi  $a, b$  va  $c$  o‘zgaruvchilar qiymati shunday almashtirilsinki, natijada  $a \leq b \leq c$  munosabat o‘rinli bo‘lsin.
17. Natural  $n (n \leq 9999)$  soni berilgan. Sonni to‘rt xonali deb hisobga olgan holda ushbu sonning palindrom ekanligi aniqlansin (chapdan va o‘ngdan bir xil o‘qiladigan sonlar, masalan, 1221, 5555, 440 sonlari palindrom sonlar hisoblanadi).
18. O‘lchamlari  $a \times b \times c$  bo‘lgan to‘g‘riburchakli qutiga  $p \times r \times q$  o‘lchamdagi to‘g‘riburchakli taxta bo‘lagini joylashtirish mumkinmi?
19. Radiusi  $r$  bo‘lgan doira ko‘rinishidagi alyumin plastinkadan o‘lchamlari  $a \times b$  va  $p \times q$  bo‘lgan ikkita to‘g‘ri to‘ritburchak shaklidagi plastinkalarni qirqib olish mumkin yo‘ki yo‘qligi aniqlansin.
20. Tomonlari koordinata o‘qlariga parallel (perpendikulyar) bo‘lgan ikkita kvadratlar bosh diagonallarining koordinatalari bilan berilgan:  $(x_1, y_1)$  va  $(x_2, y_2)$  - birinchi kvadrat;  $(x_3, y_3)$  va  $(x_4, y_4)$  - ikkinchi kvadrat. Agar kvadratlar o‘zaro kesishmasa 0, urunsa 1 va kesishsa 2 qiymati chop etilsin.
21. Ikki xonali sonlar ketma-ketligining (1011121314..9899)  $k$  - o‘rindagi

- ( $1 \leq k \leq 180$ ) raqami aniqlansin.
22. 10 soni darajalaridan tuzilgan ketma-ketlikning (10100100010000...) k - o'ringdagi raqami aniqlansin.
  23. Berilgan yilga mos keluvchi asr nomeri chop etilsin. Bunda quidagi holat inobatga olinsin: masalan, 21 asr boshi 2001 yildan hisoblanadi.
  24. Qiymati [-999,999] oraliqda yotuvchi butun son berilgan. Son qiymatiga mos ravishda "manfiy ikki xonali son", "nol soni", "uch xonali musbat son" kabi satrlar chop qilinsin.
  25. Qiymati  $1 \leq x \leq 9999$  bo'lgan x butun soni berilgan. Bu sonning qiymatiga mos ravishda quydagi satrlar chop etilsin: "to'rt xonali juft son", "ikki xonali toq son" va hokazo.
  26. Berilgan p, a va b ( $a < b$ ) sonlar uchun  $\arctg(2x - |p|) = \sqrt{2}$  tenglamaning ildizi [a, b] kesmaga tegishli ekanligi aniqlansin.
  27. Berilgan to'rt xonali sonning boshidagi ikkita raqamlari yig'indisi qolgan raqamlari yig'indisiga teng yoki yo'qligi aniqlansin.
  28. Berilgan haqiqiy musbat son kasr qismining boshidagi uchta raqamlari orasida 0 raqami bormi?

## 8. switch tarmoqlanish operatori

### *Namunaviy masala*

Butun turdagi ixtiyoriy ikkita son berilgan. Bu sonlar ustida tanlangan arifmetik amalga ('+', '-', '\*', '/') mos hisoblash amali bajarilsin. Agar arifmetik amallardan farqli belgi kiritilsa, bu haqda xabar berilsin va amalni qayta tanlash taklif qilinsin.

### *Yechish usuli*

Butun a va b sonlari klaviaturadan kiritiladi. Keyin foydalanuvchiga arifmetik amallarni tanlash taklif etiladi. Tanlangan amal amal o'zgaruvchisiga o'zlashtiriladi. switch operatori yordamida qaysi amal ekanligi aniqlanib, sonlar o'rtasida mos amal bajariladi, break operatori bajarilishi orqali tanlash operatoridan tashqariga chiqiladi va natija chop etiladi.

### *Ilova matni*

```
#include<iostream.h>
#include<math.h>
int main()
{
```

```

int a,b;
float natija;
char amal;
cout<<"a="; cin>>a;
cout<<"b="; cin>>b;
nishon: cout<<"Arifmetik amallar:";
cout<<"+: c=a+b\n"<<"- : c=a-b\n"<<"* : c=a*b\n"<<"/ : c=a/b\n";
cout<<"Amalni tanlang(+,-,*,/): ";
cin>>amal;
switch(amal)
{
case '+': natija=a+b; break;
case '-': natija=a-b; break;
case '*': natija=a*b; break;
case '/': natija=a/b; break;
default : cout<<"bunday amal yo'q!"; goto nishon;
}
cout<<"Hisoblash natijasi="<<natija;
return 0;
}

```

### *Amaliy topshiriqlar*

1. Berilganlar turi va o'zgaruvchilar quyidagicha aniqlangan:  
enum Oy{yan,fev,mar,apr,may,iyn,iyl,avg,sen,okt,noy,dek};  
int d1,d2; Oy m1,m2; bool t;  
Agar d1, m1 sana (yil hisobida) d2, m2 sanadan oldin kelsa, t o'zgaruvchiga true, aks holda false qiymat berilsin.
2. Oy m,m1; (Oy turi yuqoridagi masalada berilgan).  
int k,n;  
Quyidagi shart bo'yicha m1 o'zgaruvchiga qiymat berilsin:  
a) m oydan keyingi oyning nomi (dekabrdan keyin yanvar kelishini hisobga olgan holda);  
b) m oydan keyingi k-oyning nomi;  
d) yilning n - oy nomi berilsin.
3. enum Nota{do,re,mi,fa,sol,lya,si};  
enum Oraliq{secund,tersia,qvart,kvint,sekst,septima};  
Nota n1,n2; Oraliq i;  
Berilgan n1 va n2 ( $n1 \neq n2$ ) notalardan tashkil topgan i-oraliq aniqlansin; secund - bu ikkita qo'shni (aylana bo'ylab) notalardan tashkil topgan oraliq (masalan, re va mi, si va do), tersia – bu bitta notadan keyingi oraliq (masalan, fa va lya, si va re) va hokazo.

4. enum Mavsum {qish,bahor,yoz,kuz};  
Oy m; Mavsum s; (Oy turi 1-masalada aniqlangan).  
Berilgan m oyga mos keluvchi s mavsum aniqlansin.
5. enum Davlat{Germaniya,Quba,Laos,Monaqo,Nepal,Polsha};  
enum Qita{Osiyo,Amerika,Evropa};  
Davlat davlat; Qita qita;  
Davlatning nomi bo'yicha u joylashgan qit'a nomi aniqlansin.
6. enum Birlik{desimetr,kilometr,metr,millimetr,santimetr};  
float x; Birlik r;  
Berilgan r birlikdagi x o'zgaruvchining qiymati metrlarda aniqlansin.
7. Berilgan k o'zgaruvchi qiymati ( $0 \leq k \leq 15$ ) rim raqamlari ko'rinishida chop qilinsin.
8. enum Kelishik{bosh,qar,tush,jun,ur\_payt,chiq};  
enum Suz{ruchka,qalam,daftar,eshik};  
Suz s; Kelishik k;  
Berilgan s so'zni k kelishikda chop qilinsin. Masalan, s=daftar va k=jun bo'lganda "*daftarga*" so'zi chop qilinsin.
9. enum Yunalish{shimol,sharq,janub,garb};  
enum Buyruq{oldinga,unga,orqaga,chapga};  
Yunalish k1,k2; Buyruq br;  
Kema avvaliga k1 yo'nalish bo'yicha ketayotgan edi, keyin uning yo'nalishi br buyruqqa asosan o'zgartirildi. Kemaning yangi k2 yo'nalishi aniqlansin.
10. Oy oy; (1-masalaga qaralsin).  
int kun;  
Berilgan oydagi kunlar soni kun o'zgaruvchiga o'zlashtirilsin (yil kabisa yili emas deb hisoblansin).
11. int yil,kun;Oy oy; (Oy turi 1-masalada aniqlangan).  
bool t;  
Agar yil, oy, kun uchlik to'g'ri sanani aniqlasa, t o'zgaruvchiga true, aks holda false qiymat berilsin (masalan, 31 iyun va hokazolarda false).
12. int yil,yil1,kun,kun1;  
Oy oy,oy1; (Oy turi 1-masalada aniqlangan).  
Berilgan yil, oy, kun sanasi bo'yicha keyingi kun sanasi – yil1,oy1,kun1 aniqlansin.
13. int yil\_kuni,oy\_kuni;  
Oy oy; (Oy turi 1-masalada aniqlangan).  
a) Kabisa yilning oy, oy\_kuni sanasiga mos keluvchi kunning yildagi tartib nomeri yil\_kuni aniqlansin.



- b) Kabisa yilining hisobi bo'yicha yil kuniga mos keluvchi oy sanasi aniqlansin.
14. enum Hafta\_Kuni{yaksh,dush,sesht,chor,paysh,juma,shanba};  
int kun,k13; Oy oy; (Oy turi 1-masalada aniqlangan).  
Hafta\_Kuni h\_kun1,h\_kun2;  
Agar yil kabisa yili bo'lmasa, va uning 1 yanvari haftaning h\_kun1 kuniga to'g'ri kelsa, quyidagilar aniqlansin:  
a) kun, oy sanaga mos keluvchi haftaning h\_kun2- kuni;  
b) k13 -yildagi oyning 13-kuniga mos keluvchi dushanba kunlar soni.
15. Eski yapon kalendarida 60 yillik takrorlanish qabul qilingan va bu takrorlanish o'z navbatida beshta 12 yillik takrorlanish qismlaridan iborat. Qism takrorlanishlar quyidagi ranglarning nomi bilan belgilangan: yashil, qizil, sariq, oq va qora. Har bir takrorlanish ostining ichidagi yillar hayvonlarning nomi bilan belgilangan: sichqon, sigir, yo'lbars, quyon, ajdarho, ilon, ot, qo'y, maymun, tovuq, it va to'ng'iz (1984-yil (yashil sichqon yili) - keyingi takrorlanishning boshi bo'lgan).  
Eramizning biror yili kiritilib, uning eski yapon kalendaridagi nomi topilsin.

## 9. for takrorlash operatori

### *Namunaviy masala*

Fibonachchi sonlari quyidagicha aniqlanadi:  $f_0 = f_1 = 1, f_n = f_{n-1} + f_{n-2}$ ,  $n = 2, 3, \dots$ . Fibonachchi sonlaridan hosil bo'lgan ketma-ketlikning 40 - hadi topilsin.

### *Yechish usuli*

Fibonachchi sonlari ketma-ketligida har bir hadi (2-hadidan boshlab) o'zidan oldingi ikkita hadning yig'indisiga teng. Shu sababli, ketma-ketlikni hosil qilish uchun uchta o'zgaruvchi etarli:  $f_0, f_1, f_2$ . Ketma-ketlikning 2-hadi  $f_2 = f_0 + f_1$  ko'rinishda hisoblanadi, 3-hadni topish uchun esa  $f_0 = f_1, f_1 = f_2$  amallari bajarilib, yana  $f_2 = f_0 + f_1$  topiladi. Bu yerda garchi yana  $f_2$  qiymat hisoblansa ham, amalda u ketma-ketlikdagi  $f_3$  qiymati bo'ladi. Jarayon 39 marta takrorlanib,  $f_{40}$ - had topiladi.

### *Ilova matni*

```
#include<iostream.h>
int main()
{
const int n=40;
```

```

int f0,f1,f2,i;
f0=f1=1;
for(i=2;i<=40;i++) { f2=f0+f1; f0=f1; f1=f2;}
cout<<"Fibonachchi sonlari ketma-ketligining "<<n<<"-hadi."<<f2;
return 0;
}

```

### *Amaliy topshiriqlar*

- Berilgan  $n$  ta haqiqiy sonlar orasida qo'shnilaridan (o'zidan oldingi va keyingi sonlardan) katta bo'lgan sonlar miqdori topilsin.
- Berilgan 10 ta natural sonlarning eng katta umumiy bo'luvchisi topilsin.
- Berilgan  $n$  va  $m$  natural sonlari uchun  $s = \sum_{i=1}^n \prod_{j=5}^m (i+j)$  ifodaning qiymati hisoblansin.
- Berilgan  $n$  ta natural sonlardan iborat ketma-ketlikdagi tartib nomerlari Fibonachchi sonlari bo'lgan hadlar yig'indisi hisoblansin.
- Quyidagi satr chop etilsin: 1 – a A , 2 – b B, 3 – c C, ..., 26 – z Z.
- Berilgan natural  $n$  va  $m$  uchun  $s = \prod_{i=3}^n \sum_{j=2}^m (2*i + j*j)$  ifodaning qiymati hisoblansin.
- Berilgan natural  $n$ ,  $m$  va haqiqiy  $a$  sonlari uchun  $s = \sum_{i=4}^n \sum_{j=2}^m (a^i + j)$  ifodaning qiymati hisoblansin.
- 0 bilan tugaydigan sonlar ketma-ketligi berilgan (0 soni ketma-ketlikka kirmaydi). Ketma-ketlikdagi barcha musbat sonlar yig'indisi topilsin.
- Natural  $n$  soni va  $n$  ta haqiqiy sonlardan iborat ketma-ketlik berilgan. Toq o'rinda turgan sonlar maximumi va juft o'rindagilarning minimumi topilsin.
- Berilgan  $n$  natural sondagi turli raqamlar miqdori aniqlansin.
- Natural  $n$  soni va  $n$  ta sonlar juftligi berilgan -  $x_1, y_1, x_2, y_2, \dots, x_n, y_n$ . Har bir sonlar juftligi tekislikdagi nuqta koordinatasi deb hisoblab, berilgan nuqtalarni o'z ichiga oluvchi, markazi koordinata boshida bo'lgan eng kichik aylananing radiusi topilsin.
- Raqamlari yig'indisi  $n$  soniga teng ( $1 \leq n \leq 27$ ) bo'lgan uch xonali natural sonlar soni -  $k$  topilsin. Butun sonlarni bo'lish amallaridan ( $/$ ,  $\%$ ) foydalanilmasin.
- O'nlik yozuvida bir xil raqamlari bo'lmagan uch xonali sonlar o'sish tartibida chop etilsin (butun sonlarni bo'lish amalidan foydalanilmasin).

14. Quyida keltirilgan to‘g‘ri to‘rtburchaklar usulidan foydalangan holda

$\int_0^{3.14} \ln(2 + \sin(x))dx$  integrali taqribiy hisoblansin:

$$\int_a^b f(x)dx \approx h[f(x_1) + \dots + f(x_n)],$$

bu yerda  $h = \frac{b-a}{n}$ ,  $x_i = ih - h/2$ ,  $i=1..n$ ,  $n=100$ .

15. Butun  $n$  ( $n > 1$ ) soni va  $n$  ta haqiqiy sonlardan iborat ketma-ketlik berilgan. Ketma-ketlik o‘svuchi yoki yo‘qligi aniqlansin.

16. Butun  $n$  ( $n > 1$ ) soni va  $n$  ta haqiqiy sonlardan iborat ketma-ketlik berilgan. Ketma-ketlikning manfiy elementlari orasidan eng kattasi topilsin.

17. Trapetsiya formulasidan -

$$\int_a^b f(x)dx \approx I_n = h \left[ \frac{f(a)}{2} + f(a+h) + f(a+2h) + \dots + f(b-h) + \frac{f(b)}{2} \right], \quad h = \frac{b-a}{n}$$

foydalanib,  $\int_c^d \cos e^x dx$  integrali  $\varepsilon$ -aniqlikda hisoblansin. Bu yerda  $c$ ,  $d$

va  $\varepsilon$  ( $s < d, \varepsilon > 0$ ) berilgan sonlar. Zaruriy aniqlikka erishish uchun Runge qoidasidan foydalanilsin: agar  $I_n$  integralning taqribiy qiymati  $n=2n_0, 4n_0, 8n_0$  va hokazolarda hisoblangan bo‘lib (bu yerda  $n_0$  - oraliqni boshlang‘ich bo‘lishlar soni, masalan  $n_0=10$ ), ular uchun  $|I_{2n} - I_n|/3 < \varepsilon$  shart bajarilganda,  $I_{2n}$  - integralning natijaviy qiymati sifatida olinsin.

18. Berilgan 80 ta haqiqiy sonlar orasidan biror butun songa eng yaqinining tartib nomeri topilsin.

19. Butun  $n$  ( $n > 1$ ) soni va  $n$  ta butun sonlardan iborat ketma-ketlik berilgan. Ulardan nechtasi eng katta qiymat qabul qilishi aniqlansin.

20. Berilgan  $n$  ta ( $n > 0$ ) haqiqiy  $x_0, x_1, \dots, x_{n-1}$  sonlardan foydalanib, quyidagi

kattaliklar hisoblansin:  $M = \frac{\sum x_i}{n}$ ,  $D = \sqrt{\frac{\sum (x_i - M)^2}{n-1}}$ .

21. Berilgan  $x_i$  ( $i = \overline{1,55}$ ) sonlari uchun  $x_1(x_2 + x_3)(x_4 + x_5 + x_6) \dots (x_{46} + x_{47} + \dots + x_{55})$  ifodaning qiymati hisoblansin.

22. 100 ta butun sonlardan iborat ketma-ketlik berilgan. Shu ketma-ketlikdagi faqat nollardan iborat eng katta ketma-ketlik ostisining uzunligi topilsin.

23. Berilgan,  $a_i < b_i$  shartni qanoatlantiruvchi  $a_1, b_1, a_2, b_2, \dots, a_n, b_n$ , ( $n > 2$ ) sonlarni bir chiziqda yotuvchi kesmalarining chap ( $a_i$ ) va o‘ng ( $b_i$ ) chegaralari deb qarab, barcha kesmalar kesishmasidan hosil bo‘lgan

kesma uchlari -  $(a_k, b_k)$  topilsin, agar bunday kesma mavjud bo'lsa, bu haqda xabar berilsin.

24. Berilgan natural sonning mukammalligi, ya'ni o'zidan boshqa musbat bo'luvchilarining yig'indisi shu sonning o'ziga tengligi aniqlansin. (misol uchun, 6- mukammal, chunki  $6=1+2+3$ ).
25.  $[2, n]$  ( $n > 2$ ) oraligidagi barcha tub sonlar chop etilsin.
26. Berilgan natural sonning barcha tub bo'luvchilari chop etilsin.
27. Hadlar soni ikkitadan kam bo'lmagan nol bilan tugaydigan natural sonlar ketma-ketligi berilgan. Tartib nomerlari tub sonlar bo'lgan hadlar yig'indisi hisoblansin.
28. Berilgan natural sonning raqamlarini teskari tartibda yozishdan hosil bo'ladigan son topilsin.
29. Butun  $n$  ( $n > 0$ ) soni va  $n$  ta haqiqiy sonlar ketma-ketligi berilgan. Ketma-ketlik monotonlikka tekshirilsin.
30. O'nlik sanoq sistemasida berilgan son o'n oltilik sanoq sistemasiga o'tkazilsin.
31. Natural  $n$  soni va o'zaro teng bo'lmagan  $n$  ta butun sonlar ketma-ketligi berilgan. Bu ketma - ketlikning eng katta elementidan kichik bo'lgan sonlarning eng kattasi topilsin.
32. Berilgan ixtiyoriy  $n$  ta sonlar ichidan to'la kvadratlar miqdori aniqlansin.
33. Haqiqiy turdagi  $n$  ta sonlar juftligi berilgan.  $x_1, y_1, x_2, y_2, \dots, x_n, y_n$ . Har bir sonlar juftini tekislikdagi nuqta koordinatalari deb qarab, berilgan nuqtalarni ichiga oladigan eng kichik radiusli aylana radiusi  $r$  aniqlansin.
34. Musbat  $a$  haqiqiy soni va  $s = 1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{n} + \dots$  qonuniyat berilgan.  $s > a$  shartni qanoatlantiruvchi birinchi  $n$  soni chop etilsin.
35. Berilgan musbat  $a$  haqiqiy sonning raqamlari yig'indisi topilsin.

## 10. while, do-while takrorlash operatorlari

### *Namunaviy masala*

Berilgan haqiqiy  $x$  va  $\varepsilon > 0$  uchun standart funksiyalardan foydalanmagan holda (fabs bundan mustasno) quyidagi ifoda hisoblansin.

$$s = e^x = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!} + \dots$$

### ***Yechish usuli***

Ifoda  $s = s + \frac{a}{b}$  ko‘rinishdagi yig‘indi masalasidir. Yig‘indini hisoblash

esa, har qadamda  $s$  yig‘indiga  $\frac{a}{b}$  qiymatini qo‘shishni amalga oshiradigan takrorlash jarayoni vositasida bajariladi. Bu yerda  $a$  va  $b$  parametrlarni hisoblash qadamiga bog‘liqligini topish kerak bo‘ladi. Jarayon boshlanishida ( $i=0, s=1, a=1, b=1$ ) qiymatlarni qabul qilsin. Takrorlashning  $i$ -qadamida ( $i \geq 1$ )  $a$  qiymati  $i-1$  chi qadamdagi  $a$  qiymatini  $x$  soniga ko‘paytirishdan, maxraj esa oldingi  $b$  qiymatiga  $i$  qiymatini kopaytirishdan hosil bo‘ladi. Yig‘indini hisoblash jarayoni navbatdagi qo‘shiladigan hadning absolyut qiymati  $\varepsilon$  sonidan kichik bo‘lguncha davom etadi ( $\left| \frac{a}{b} \right| < \varepsilon$ ).

### ***Hlova matni***

```
int main()
{
    float eps,x,a=1.0,b=1.0,s=0.0;
    int i=0;
    cout<<" eps="; cin>>eps;
    cout<<"\n x="; cin>>x;
    do { s+=a/b; a*=x; b*=++i; } while(abs(a/b)>=eps);
    cout<<"exp(x)="<<s;
    return 0;
}
```

### ***Amaliy topshiriqlar***

1. Berilgan, 7 so‘mdan katta bo‘lgan har qanday tiyinsiz pul miqdorini 3 va 5 so‘mliklar yig‘indisi bilan qaytimsiz to‘lash mumkinligi isbotlansin, ya’ni berilgan  $n > 7$  uchun  $3a + 5b = n$  shartni qanoatlantiruvchi musbat butun  $a$  va  $b$  sonlar juftliklari topilsin.
2. Berilgan natural  $n$  va  $m$  soni uchun  $t = \sum_{i=2}^n \prod_{j=3}^m (i * i / j)$  hisoblansin.
3. Berilgan natural  $n$  va  $m$  soni uchun  $t = \prod_{i=3}^{n-1} \sum_{j=2}^{m-3} (2 * i + j)$  hisoblansin.
4. Berilgan  $n$  ta sonning maksimumidan farqli sonlar ichida eng kattasi topilsin.
5. Berilgan 10 ta natural sonlarning eng katta umumiy bo‘luvchisi topilsin.
6. Berilgan natural  $n$  sonining barcha bo‘luvchilari topilsin.
7. Berilgan  $n$  uchun  $s$  ning qiymati hisoblansin:

a)  $s=1! - 2! + 3! - 4! + \dots + (-1)^{n+1}n!$ ;

b)  $s=-2!+4!+\dots+(-1)^n(2n)!$ .

8. Ketma-ketlik quyidagi qonuniyat bilan berilgan:

$$y_0 = 0; \quad y_k = \frac{y_{k-1} + 1}{y_{k-1} + 2}, \quad k = 1, 2, \dots$$

Berilgan  $\varepsilon > 0$  uchun  $y_n - y_{n-1} < \varepsilon$  shartni qanoatlantiruvchi birinchi  $y_n$  had topilsin.

9. Berilgan natural  $n$  soni raqamlarining yig'indisi hisoblansin.
10. Standart funksiyalardan foydalanmagan holda (fabs()-bundan mustasno) berilgan  $\varepsilon > 0$  aniqlikda  $y$  qiymati hisoblansin. Yig'indini hisoblashda navbatdagi qo'shiluvchi had moduli bo'yicha  $\varepsilon$  dan kichik bo'lgan holda hisoblash jarayoni to'xtatilsin.
- a)  $y = \operatorname{sh}x = x + x^3/3! + x^5/5! + \dots + x^{2n+1}/(2n+1)! + \dots$ ;
- b)  $y = \cos x = 1 - x^2/2! + x^4/4! - \dots + (-1)^n x^{2n}/(2n)! + \dots$ ;
- d)  $y = \ln(1+x) = x - x^2/2 + x^3/3 - \dots + (-1)^{n-1} x^n/n + \dots, (|x| < 1)$ ;
- e)  $y = \operatorname{arctg}x = x - x^3/3 + x^5/5 - \dots + (-1)^n x^{2n+1}/(2n+1) + \dots, (|x| < 1)$ .
11. Bir-biridan farqli, uchtadan kam bo'lmagan natural sonlar ketma-ketligi berilgan bo'lib, u 0 bilan tugallanadi. Shu sonlar ichidan uchta eng kattasi topilsin.
12. Nol bilan tugaydigan, noldan farqli butun sonlar ketma-ketligida ishora o'zgarishlar soni aniqlansin. (Masalan, 1,-34,8,4,-5,0 ketma-ketlikda ishora 3 marta o'zgaradi).
13. Hadlar soni ikkitadan kam bo'lmagan nol bilan tugaydigan natural sonlar ketma-ketligi berilgan. Tartib nomerlari tub son bo'lgan hadlarning yig'indisi aniqlansin.
14. Quyida keltirilgan ketma-ketliklarning  $k$ -raqami chop etilsin:
- a) 12345678910111213...-ketma-ket yozilgan natural sonlar;
- b) 149162536... - natural sonlar kvadratlari;
- d) 1123581321...- Fibonachchi sonlari.
15. O'nlik sanoq sistemasida natural  $p$  soni berilgan bo'lib, uning  $q(2 \leq q \leq 16)$  sanoq sistemasidagi ko'rinishi hosil qilinsin.
16. O'nlik kasr soni  $z$  uchun uning  $q(2 \leq q \leq 16)$  sanoq sistemasidagi verguldan keyingi to'rt xona aniqlikdagi ko'rinishi hosil qilinsin.
17. O'nlik sanoq sistemasida butun  $m$  soni berilgan bo'lib, uning ikkilik sanoq sistemasidagi ko'rinishidagi sonda 0 o'rniga 1 va 1 o'rniga 0 almashtirishdan hosil bo'lgan sonning o'nlik sanoq sistemasidagi ko'rinishi aniqlansin.
18. O'nlik sanoq sistemasidagi butun  $p$  sonining o'n oltilik sanoq siste-

- masidagi ko‘rinishida ‘E’ raqami bor yoki yo‘qligi aniqlansin.
19. O‘nlik sanoq sistemasidagi  $p(p < 1)$  kasr sonining oltilik sanoq sistemasiga o‘tkazilganda ‘4’ raqami necha marta uchrashi aniqlansin.
  20. O‘nlik sanoq sistemasidagi butun  $a$  va  $b$  sonlarning uchlik sanoq sistemasidagi ko‘rinishida birinchi raqamlari mos tushish yoki tushmasligi aniqlansin.
  21. Butun  $m$  va  $n$  sonlar berilgan. Umumiy bo‘luvchiga ega bo‘lmagan  $\frac{p}{q} = \frac{m}{n}$  shartni qanoatlantiruvchi  $p$  va  $q$  butun sonlar topilsin.
  22. Musbat haqiqiy  $a, x$  va  $\varepsilon$  sonlar berilgan.  $y_1, y_2, \dots$  ketma-ketlik quyidagi qonuniyat bo‘yicha hosil qilingan:

$$y_0 = a; y_i = \frac{1}{2} \left( y_{i-1} + \frac{x}{y_{i-1}} \right), i = 1, 2, \dots$$

Ketma-ketlikning  $|y_n^2 - y_{n-1}^2| < \varepsilon$  shartni qanoatlantiruvchi birinchi  $y_n$  hadi topilsin.

23. Berilgan, ketma-ketlikni hosil qilish qonuniyati

$$x_0 = 1; x_k = \frac{2 - x_{k-1}^2}{5}, k = 1, 2, \dots$$

uchun  $|x_n - x_{n-1}| < 10^{-5}$  o‘rinli bo‘lgan birinchi  $x_n$  hadi toplisin.

24. Haqiqiy  $x, a$  va  $\varepsilon (\varepsilon > 0, |x| < 1)$  sonlar berilgan.

$$1 + \sum_{k=1}^{\infty} \frac{a(a-1) \dots (a-k+1)}{k!} x^k$$

ifodaning qiymati  $\varepsilon$  aniqlikda hisoblansin.

25. Haqiqiy  $\varepsilon (\varepsilon > 0)$  soni va  $a_1, a_2, \dots$  ketma-ketlik quyidagi qonuniyat bilan berilgan:

$$a_n = \left(1 - \frac{1}{2}\right) \left(1 - \frac{1}{3}\right) \dots \left(1 - \frac{1}{n+1}\right).$$

Ketma-ketlikning  $|a_n - a_{n-1}| < \varepsilon$  shartni qanoatlantiruvchi birinchi  $a_n (n \geq 2)$  hadi topilsin.

26. Haqiqiy  $a, b, \varepsilon (a > b > 0, \varepsilon > 0)$  sonlar va  $x_1, x_2, \dots, y_1, y_2, \dots$  ketma-ketliklar quyidagi qonuniyat bilan berilgan:

$$x_1 = a, y_1 = b, x_k = \frac{1}{2}(x_{k-1} + y_{k-1}), y_k = \sqrt{x_{k-1} y_{k-1}}.$$

Ketma-ketliklarning  $|x_n - y_n| < \varepsilon$  shartni qanoatlantiruvchi birinchi  $x_n$  hadi topilsin.

27. O'nlik  $\frac{a}{b}$  ko'rinishidagi kasrning davriy yoki yo'qligini aniqlash uchun  $b$  soni tub bo'luvchilar ko'paytmasi ko'rinishida yoziladi. Agar ko'paytma faqat 2 va 5 sonlardan iborat bo'lsa, kasr davriy emas, aks holda davriy hisoblanadi.

Masalan:

- a)  $\frac{17}{20}$ ;  $20 = 2^2 * 5$  - kasr davriy emas; b)  $\frac{17}{36}$ ;  $36 = 2^2 * 3^2$  - kasr davriy.

## 11. Funksiyalar

### *Namunaviy masala*

Berilgan butun sonning raqamlari yig'indisi hisoblansin.

### *Yechish usuli*

Berilgan butun sonning raqamlari yig'indisini hisoblash butun turdagi qiymat qaytaruvchi funksiya ko'rinishida amalga oshiriladi (int Raqamlar\_Summasi()). Funksiya bitta butun turdagi  $n$  parametriga, yani raqamlar yig'indisi hisoblanishi zarur bo'lgan songa ega. Funksiya tanasida dastlab  $n$  soni manfiylikka tekshiriladi va u manfiy bo'lsa  $n$  ning qiymati qarama-qarshi songa aylantiriladi. Shundan keyin, toki  $n$  nolga teng bo'lmaguncha  $n$  sonini 10 ga bo'lishdagi qoldig'i  $s$  o'zgaruvchiga yig'iladi,  $n$  o'zgaruvchining yangi qiymati  $n/10$  bo'linmaning butun qismi bo'ladi. Funksiya natija sifatida  $s$  qiymatni qaytaradi.

### *Ilova matni*

```
#include<iostream.h>
int Raqamlar_Summasi(int);
int main()
{
    int b;
    cout<<"\nButun sonni kiritig:"; cin>>b;
    cout<<"\n"<<Raqamlar_Summasi(b);
    return 0;
}
int Raqamlar_Summasi(int n)
{
    int s=0;
    if(n<0)n=-n;
    while(n>0) { s+=n%10; n/=10; }
    return s;
}
```



### ***Amaliy topshiriqlar***

1. Ixtiyoriy  $n$  ta sonning yig'indisini hisoblash funksiyasi tuzilsin.
2. Ixtiyoriy  $n$  ta sonning maksimumini topish funksiyasi tuzilsin.
3. Ikkita uchburchak uchlarining koordinatalari bilan berilgan. Bu uchburchaklardan qaysi birining yuzasi katta ekanligini aniqlovchi funksiya tuzilsin.
4.  $n$  natural soni va  $n$  ta sonlardan iborat ketma-ketlik berilgan. Ketma-ketlikdagi ishora almashinishlar sonini aniqlash funksiyasi tuzilsin.
5. Berilgan  $n$  va  $m$  ( $n \geq m$ ) natural sonlar uchun Nyuton binomi koeffisientlarini hisoblash funksiyasi tuzilsin:  $C_n^m = \frac{n!}{(n-m)!m!}$ .
6. Kompleks  $a_1 + ib_1$  va  $a_2 + ib_2$  sonlar ustida arifmetik amallar ('+', '-', '\*', '/') bajaruvchi funksiya tuzilsin, bu yerda  $a_1, b_1, a_2, b_2$  - berilgan butun sonlar.
7. Tekislikda  $a, b, c$  va  $d$  kesmalar berilgan. Ularning har bir uchtasidan uchburchak qurish mumkinligini aniqlaydigan mantiqiy Uchburchak() funksiya tuzilsin. Agar Uchburchak() funksiyasi birorta uchlik uchun true qiymat qaytarsa, shu uchlikdan hosil bo'lgan uchburchakning yuzasini hisoblovchi haqiqiy turdagi Uchburchak\_Yuzasi() funksiya tuzilsin.
8. Butun  $a, b, p, q$  ( $b \neq 0$ ) parametriga ega qisqart(a,b,p,q) funksiya  $a/b$  kasrni qisqarmaydigan  $p/q$  ko'rinishiga olib kelsin va undan  $1+1/2+1/3+\dots+1/20$  kasrni qisqarmaydigan  $c/d$  ko'rinishiga keltirishda foydalanilsin.
9. Uchburchak  $a, b$  va  $c$  tomonlari bilan berilgan. Uchburchakning medianalaridan iborat uchburchakning medianalari topilsin. (Izoh: uchburchakning  $a$  tomoniga o'tkazilgan mediana  $0.5\sqrt{2b^2 + 2c^2 - a^2}$  ga teng).
10. Haqiqiy  $c, d$  ( $c < d$ ) sonlari berilgan. Quyidagi integral hisoblansin:

$$\int_c^d \arctg^2 x dx + \int_0^\pi \sin e^{10x} dx.$$

Birinchi integral  $n=20$  da, ikkinchisi  $n=100$  da trapetsiya formulasidan foydalanib hisoblansin:

$$\int_a^b f(x) dx \approx h \cdot \left[ f(a)/2 + \sum_{i=1}^{n-1} f(a+ih) + f(b)/2 \right]$$

bu yerda  $h = (b-a)/n$ .

11. Berilgan  $1/(1+x^2) = x, 3e^x + x = 0$  va  $x * \ln(1+x) = 0.5$  tenglamalarning yechimlari  $\varepsilon > 0$  aniqlikda topilib, o'sish tartibida chop etilsin.
12. Uchburchak uchlari va shu uchburchak ichidagi biror nuqta koordinatasi bilan berilgan. Berilgan nuqtadan uchburchak tomonlarigacha

bo'lgan eng yaqin masofa topilsin.

13. Tekislikda uchta to'g'ri chiziq  $a_k x + b_k y = c_k$  ( $k = 1, 2, 3$ ) tenglamalari bilan berilgan. Agar bu to'g'ri chiziqlar juft-jufti bilan kesishib, uchburchak hosil qilsa, shu uchburchak yuzasi hisoblansin.
14.  $n$  dan  $2n$  ( $n > 2$ ) gacha bo'lgan sonlar ichidan "egizaklar" jufti chop etilsin. (Ikkita tub sonlar "egizak" deyiladi, agarda ular bir-biri bilan 2 ga farq qilsa, masalan: 41 va 43 sonlari).
15. Berilgan  $n$  va  $k$  uchun  $n$  sonidan boshlanuvchi  $k$  ta "tug'ma sonlar" ketma-ketligi hosil qilinsin. Ketma-ketlikning hadi oldingi hadga uning raqamlari yig'indisini qo'shish orqali hosil bo'ladi. Masalan, ketma-ketlik hadi 13 bo'lsa undan keyingi son  $13 + (1 + 3) = 17$  bo'ladi va hokazo.
16. Berilgan natural sondan katta bo'lmagan barcha "do'st" sonlar juftiligi chop etilsin (Ikkita natural son "do'st" deyiladi, agarda ularning har biri ikkinchisining bo'luvchilari (o'zidan tashqari) yig'indisiga teng bo'lsa, masalan: 220 va 284 sonlari).
17. Berilgan  $a > 0$  haqiqiy son uchun

$$\frac{\sqrt[3]{a - \sqrt{a^2 + 1}}}{1 + \sqrt[3]{3 + a}}$$

kattalik hisoblansin. Bu yerda  $y = \sqrt[k]{x}$  ko'rinishidagi ildiz quyida keltirilgan iterativ formuladan foydalanib,  $\varepsilon$  ( $\varepsilon = 0.0001$ ) aniqlikda hisoblansin.

$$y_0 = 1; y_{n+1} = y_n + (x / y_n^{k-1} - y_n) / k \quad (n = 0, 1, 2, \dots).$$

Yechim sifatida  $|y_{n+1} - y_n| < \varepsilon$  shartni qanoatlantiruvchi  $y_{n+1}$  olinsin.

18. Berilgan haqiqiy  $\varepsilon > 0$  va  $t$  sonlari uchun

$$\sqrt[4]{1 - \frac{\cos^4 t}{4}} + \sqrt[5]{1 + \frac{\arctg t}{2}} \cdot \sqrt[9]{\frac{1}{3 + t^2}}$$

ifoda  $\varepsilon$  aniqlikda hisoblansin. Ildizlarni hisoblashda quyidagi Teylor qatoridan foydalanilsin:

$$(1 + x)^a = 1 + \frac{a}{1!}x + \frac{a(a-1)}{2!}x^2 + \frac{a(a-1)(a-2)}{3!}x^3 + \dots, (|x| \leq 1, a > 0).$$

19. 1, 2, ...,  $k$  nuqtalarda (bu yerda  $k - 2$  dan 70 gacha bo'lgan butun son) quyidagi funksiyalarning grafiklari chop etilsin:
  - a)  $\varphi(n)$  -  $1..n$  gacha  $n$  soni bilan o'zaro tub bo'lgan butun sonlar miqdori;
  - b)  $\tau(n)$  -  $n$  sonining musbat bo'luvchilari soni;
  - d)  $\pi(n)$  -  $n$  sonidan oshmaydigan tub sonlar soni.

Funksiya grafigini chizishda koordinata sistemasida  $X$  o'qi pastga,  $Y$  o'qi chap tomonga yo'nalgan deb hisoblanadi. Ekrandagi belgi o'rni

o‘q birligi qilib olinadi. Har bir  $x$  nuqta uchun ( $x=1,2,\dots,k$ ) funksiya qiymati -  $y$  hisoblanadi va ekrandagi ayni satrga  $y-1$  ta probel, undan keyin “\*” belgisi chop qilinadi va keyingi qatorga o‘tiladi ( $x$  o‘zgaruvchisining keyingi qiymati uchun).

20.  $A(x_1, y_1), B(x_2, y_2)$  va  $C(x_3, y_3)$  nuqtalar berilgan. Masalalar yechilsin:
- shu nuqtalardan uchburchak hosil bo‘lsa true, aks holda false qiymat qaytaruvchi funksiya tuzilsin;
  - agar  $ABC$  uchburchak mavjud bo‘lsa, bu uchburchak turini aniqlovchi funksiya (teng tomonli, teng yonli, to‘g‘riburchakli) tuzilsin.
21. Uchlari  $A(x_1, y_1), B(x_2, y_2), C(x_3, y_3)$  va  $D(x_4, y_4)$  nuqtalarda bo‘lgan to‘rtburchakni yasash mumkinmi? Agar mumkin bo‘lsa, uning turi aniqlansin: romb, parallelogram, trapetsia, to‘g‘riburchakli yoki ixtiyoriy to‘rtburchak.

## 12. Vektorlar

### *Namunaviy masala*

Berilgan  $n$ -darajali  $a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0$  va  $m$ -darajali  $b_m x^m + b_{m-1} x^{m-1} + \dots + b_1 x + b_0$  ko‘phadlar ko‘paytmasidan hosil bo‘lgan ko‘phadning koeffitsiyentlari topilsin.

### *Yechish usuli*

Darajalari  $n$  va  $m$  bo‘lgan ko‘phadlarni ko‘paytirish natijasida  $n+m$  darajali ko‘phad hosil bo‘ladi:  $c_{n+m} x^{n+m} + c_{n+m-1} x^{n+m-1} + \dots + c_1 x + c_0$ . Asosi bir xil darajali sonlarni ko‘paytirganda ularning darajalari qo‘shilishi xossasidan foydalangan holda mos koeffitsientlar topiladi:  $c_{i+j} = \sum a_i * b_j$ . Har bir ko‘phadning ozod hadini inobatga olgan holda, ular uchun ajratiladigan massivlar o‘lchami mos ko‘phad darajasidan bittaga ortiq bo‘lishi kerak.

### *Ilova matni*

```
int main()
{
    const int n=2+1,m=4+1;
    int a[n],b[m],c[n+m-1],i,j;
    for(i=0;i<n;i++) cin>>a[i];
    for(j=0;j<m;j++) cin>>b[j];
    for(i=0;i<n+m-1;i++) c[i]=0;
    for(i=0;i<n;i++)
        for(j=0;j<m;j++) c[i+j]+=a[i]*b[j];
    for(i=n+m-2;i>=0;i--)
```

```

{
if(c[i]==0) continue;
if(c[i]>0&&i!=(n+m-2)) cout<<" ";
if(c[i]!=1) cout<<c[i];
if(i>0) cout<<"x"; if(i>1) cout<<"^"<<i;
}
return 0;
}

```

Yuqoridagi ilova da 2 va 4 - darajali ko'phadlarni ko'paytirishdan hosil bo'lgan ko'phad koeffisientlarini hisoblash ko'rsatilgan. Ilova ishga tushirilganda  $x^2 + 2x + 3$  va  $5x^4 + 4x^3 + 3x^2 + 2x + 1$  ko'phadlar koeffisientlari kiritilsa, natija sifatida quyidagi ko'phad chop etiladi:

$$5x^6 + 14x^5 + 26x^4 + 20x^3 + 14x^2 + 8x + 3$$

### *Amaliy topshiriqlar*

- Bo'sh bo'lmagan, raqamlardan iborat va nuqta bilan tugaydigan matn berilgan. Matndagi eng ko'p uchraydigan raqam chop qilinsin (agar bunday raqamlar bir nechta bo'lsa, ulardan ixtiyoriy bittasi chop qilinsin).
- const int n=100;  
char s[n];  
Berilgan s vektor kattalikning indekslari quyidagilarga teng bo'lgan elementlari chop qilinsin:  
a) ikkining darajalari: (1,2,4,8,16,...);  
b) to'liq kvadratlar: (1,4,9,16,25,...);  
d) fibonachchi sonlari: (1,2,3,5,8,13,...).
- const int k=5,m=6,n=11;//n=k+m  
float x[k],y[m],z[n];  
Berilgan x va y vektorlarning har birida elementlar kamaymaydigan tartibda joylashgan. Bu ikki vektorni birlashtirishdan hosil bo'ladigan z vektorning elementlari ham kamaymaydigan tartibda joylashuvchi ilova tuzilsin.
- char m[9],p[2];  
float x;  
Berilgan m va p vektorlar raqamlardan iborat, x o'zgaruvchiga  $0.m_1m_2\dots m_9 \cdot 10^{p_1p_2}$  ko'rinishidagi haqiqiy son o'zlashtirilsin.
- enum Oy{yan,fev,mart,apr,may,iyun,iyul,avg,sen,okt,noy,dek};  
float t[365];  
Oy oy;  
Kabisa bo'lmagan biror yilning har bir kuni haroratini bildiruvchi t

vektor bo'yicha o'rtacha oylik harorati eng katta bo'lgan oyning nomi m aniqlansin.

6. `int x[50];`  
`bool t;`  
Berilgan  $x$  vektorning elementlari orasida quyidagi sonlar bor yoki yo'qligiga qarab  $t$  o'zgaruvchiga `true` yoki `false` qiymat berilsin:
  - a) kamida bitta Fibonachchi soni;
  - b) kamida ikkita ikkining darajasi ko'rinishidagi son.
7. Har biri 30 ta butun sondan iborat ikkita ketma-ketlik berilgan. Birinchi ketma-ketlikning ikkinchi ketma-ketlikka kirmagan sonlar ichidagi eng kichigi topilsin (bunaqa sonlardan kamida bittasi mavjud deb faraz qilinsin).
8. Berilgan matn 30 ta belgidan tashkil topgan. Takrorlanuvchi belgilarni o'chirishdan hosil bo'lgan matn chop qilinsin.
9. Belgilari 100 tadan ortiq bo'lmagan va nuqta bilan tugaydigan (nuqtaning o'zi matnga kirmaydi) matndagi turli belgilar soni aniqlansin.
10. Qiymati 0 dan 20 gacha bo'lgan  $k$  butun soni berilgan.  $k$ -tartibli Chebishev ko'phadi koeffitsientlari topilsin (Izoh: Chebishev ko'phadlari  $T_n(x)$  quyidagi formula bilan aniqlanadi:
$$T_0(x) = 1, T_1(x) = x, T_n(x) = 2xT_{n-1}(x) - T_{n-2}(x), n = 2, 3, \dots$$
11. Haqiqiy  $a_0, a_1, \dots, a_{15}$  sonlari berilgan.  $(x-a_0)(x-a_1)\dots(x-a_{15})$  ko'phadning koeffitsientlari topilsin.
12. Berilgan 10-darajali  $P(x)$  va 6-darajali  $Q(x)$  ko'phadning koeffitsientlari bo'yicha  $P(Q(x))$  ko'phadning koeffitsientlari topilsin.
13. 10 ta tosh bo'lib, ularning og'irliklari mos ravishda  $a_1, \dots, a_{10}$  butun sonlarga teng. Berilgan  $m_1, m_2, \dots, m_{10}$  og'irliklar uchun  $c_1, \dots, c_{10}$  hosil qilinsin. Bu yerda,  $c_k - m_k$  og'irlikni hosil qilish usullari soni, yani  $a_1x_1 + \dots + a_{10}x_{10} = m_k$  tenglamaning yechimlari, bu yerda  $x_i (i = \overline{1, 10})$  o'zgaruvchisi 0 yoki 1 qiymat qabul qiladi.
14. Sonlar o'qida  $n (n > 1)$  sondagi  $(a_1, a_2), (a_3, a_4), \dots, (a_{2n-1}, a_{2n})$  intervallar juftliklar ko'rinishda berilgan:
  - a) intervallarning umumiy nuqtalari bormi?
  - b) intervallar birlashmasi interval hosil qilsa, shu interval uchlari ko'rsatilsin ( $L_1$  va  $L_2$  intervallar birlashmasi deb shunday  $L_3$  intervalga aytiladiki,  $L_1$  va  $L_2$  intervallarga tegishli barcha nuqtalar  $L_3$  ga ham tegishli bo'lsa).
  - d) intervallar birlashmasini  $k$  ta kesishmaydigan intervallar ko'rinishida

taqdim etish mumkinmi?

e) kamida uchta intervalga tegishli butun sonlar bor bo'lsa, shu sonlardan birortasi ko'rsatilsin.

15. Tekislikda  $n$  ta ( $n \geq 4$ ) nuqta  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$  koordinatalari bilan berilgan. Ularning ichida kvadrat hosil qiluvchi nuqtalar bor yoki yo'qligi aniqlansin.
16. Markazi  $M(x_i, y_i)$  nuqtada bo'lgan  $r_i$  radiusli aylanalar ( $i=1, \dots, n$ ) berilgan. Quyidagilar aniqlansin:
  - a) aylanalar ichida uchta kesishuvchi aylanalar bormi?
  - b) alohida turgan aylanalar topilsin, ya'ni boshqa aylanalar bilan umumiy nuqtalari yo'q, birorta aylana ichida joylashmagan va boshqa aylanalarni o'z ichiga olmagan aylanalar).
17.  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$  koordinatalari bilan berilgan nuqtalar to'plami-ning medianalar soni topilsin ( $n > 2$  va  $n$  juft son). To'plam medianasi deb to'plamning ikkita ixtiyoriy nuqtasini tutashtiruvchi shunday to'g'ri chiziqqa aytiladiki, uning ikki tomonida bir hil sondagi nuqtalar joylashadi va bu to'g'ri chiziqda hech qanday uchinchi nuqta yotmaydi.
18. Arqon tortish musobaqasida ishtirok etmoqchi bo'lgan  $n$  ta o'quvchilar ( $n$ -juft son) ikki guruhga bo'linish uchun aylana shaklida joylashib, birdan to  $k$  sonigacha sanay boshladilar. Bunda har  $k$ -o'quvchi davradan chiqib, ikkinchi guruhga qo'shiladi. Sanoq davrada guruhlarda o'quvchilar soni teng bo'lguncha davom etadi. Har bir o'quvchining tartib nomeri boshlang'ich davrada sanoq boshlangan o'quvchidan boshlanib, sanoq yo'nalishi (soat millari yo'nalishi) bo'yicha aniqlandi. Berilgan  $n$  va  $k$  uchun guruhlardagi o'quvchilar tartib nomerlari aniqlansin.
19. Natural  $n$  soni va  $n$  ta haqiqiy sonlardan iborat ketma-ketlik berilgan. Bu ketma-ketlikni shunday tartiblangki, undagi barcha manfiy qiymatli elementlar o'zaro joylashish tartiblarini saqlagan holda ketma-ketlik boshiga ko'chirilsin va ulardan keyin musbat qiymatli elementlar ham huddi shu shart asosida joylashsin (qo'shimcha massiv ishlatilmasin).
20. Natural  $n$  soni va  $a_1, a_2, \dots, a_n$  butun sonlardan iborat ketma-ketlik berilgan. Ketma-ketlikda bir marta qatnashgan elementlar chop qilinsin.
21. Natural  $n$  soni va  $a_1, a_2, \dots, a_n, b_1, b_2, \dots, b_n$  sonlar ketma-ketliklari berilgan. Bu ketma-ketliklar bir-biridan faqat elementlarining joylashuv tartibi bilangina farq qilishi yoki yo'qligi aniqlansin.
22. Natural  $n$  soni,  $A = \{a_1, a_2, \dots, a_n\}$  va  $B = \{b_1, b_2, \dots, b_n\}$  sonlar ketma-ketliklari berilgan.  $A$  ketma-ketlik  $[1..n]$  oralig'idagi har xil butun son-

- lardan tashkil topgan (indekslar). B ketma-ketlikdagi elementlar A ketma-ketlikda ko'rsatilgan joylarga o'tkazilsin, ya'ni  $b_j$  element  $a_i$  indeks bo'yicha joylashsin.
23. Juft bo'lgan  $n$  natural soni va yarmigacha qiymatlar bilan to'ldirilgan  $A = \{a_1, a_2, \dots, a_n\}$  ketma-ketlik berilgan. Ketma-ketlik elementlarining qiymatlarini takrorlab yozish orqali to'ldirilsin (masalan,  $A = \{3, 8, \dots\}$  uchun  $A = \{3, 3, 8, 8, \dots\}$ ).
24. Natural  $n$  soni va  $A = \{a_1, a_2, \dots, a_n\}$  ketma-ketlik berilgan. Ketma-ketlikdagi elementlar o'sish tartibida bo'lgan eng uzun qism ketma-ketlik topilsin.
25. Fazoda  $n$  material nuqtadan iborat sistema  $x_1, y_1, z_1, x_2, y_2, z_2, \dots, x_n, y_n, z_n$  haqiqiy sonlar ketma-ketligi ko'rinishida berilgan. Bu yerda  $x_i, y_i, z_i$  -  $i$ -chi nuqtaning koordinatalari ( $i = \overline{1, n}$ ). Berilgan  $n$  o'zgarmas uchun sistemaning og'irlik markazi koordinatasi, hamda og'irlik markazidan sistemaning barcha nuqtalarigacha bo'lgan masofalar topilsin.
26. Berilgan  $x, y, z$  vektorlarni element turidagi obyektlar to'plami deb qarab (agarda  $k$  element  $x$  to'plamga tegishli bo'lsa  $x[k]=\text{true}$ , aks holda  $x[k]=\text{false}$  va hokazo), ushbu vektor - to'plamlar ustida quyidagilar amalga oshirilsin:
- agar  $x$  to'plam  $y$  to'plamning qism to'plami bo'lsa, u holda  $t$  o'zgaruvchiga  $\text{true}$  qiymati, aks holda  $\text{false}$  berilsin;
  - $z = x \cap y$  - to'plamlar kesishmasi topilsin;
  - $z = x \cup y$  - to'plamlarning birlashmasi topilsin;
  - $z = x \setminus y$  - to'plamlarning ayirmasi topilsin ( $z$  to'plamga  $x$  to'plamning  $y$  to'plamga kirmagan barcha elementlari kiradi).

## 13. Matritsalar

### *Namunaviy masala*

Elementlari bir-biriga teng bo'lmagan  $n$  - tartibli haqiqiy turdagi kvadrat matritsa berilgan. Eng katta element joylashgan satrni eng kichik element joylashgan ustunga kopaytmasi topilsin.

### *Yechish usuli*

Matritsaning eng katta va eng kichik elementlari topiladi. Buning uchun matritsaning  $(0,0)$  o'rindagi elementi maksimum (minimum) deb faraz qilinadi va  $i$  - satr va  $j$  - ustunlar bo'icha ichma-ich joylashgan takrorlash jarayoni amalga oshiriladi. Maksimum (minimum) element topish bi-

lan birga uning joylashgan o‘rni biror maxi o‘zgaruvchida (minimum uchun minj o‘zgaruvchida) eslab qolinadi. Keyin matritsaning maxi - satri elementlari va minj - ustun elementlariga skalyar kopaytmasi hisoblanadi.

### ***Ilova matni***

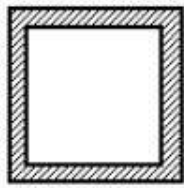
```
#include <iostream.h>
#include <conio.h>
#define n 3
typedef float Fmatritsa[n][n];
int main()
{
    Fmatritsa A; float max,min; int maxi,minj;
    for(int i=0;i<n;i++) for(int j=0;j<n;j++)
        cin>>A[i][j]; //A matritsa elementlarini o‘qish
    // A[0,0]- maksimum hamda minimum element deb faraz qilinadi va matritsaning
    // qolgan elementlari bilan solishtiriladi;
    max=A[0][0]; maxi=0; min=A[0][0]; minj=0;
    for(int i=0;i<n;i++) for(int j=0;j<n;j++)
        if(max<A[i][j]){max=A[i][j]; maxi=i;}
        else if(min>A[i][j]){min=A[i][j]; minj=j;}
    // Maksimum element joylashgan satrni minimum joylashgan ustunga ko‘paytiriladi
    float S=0;
    for(j=0;j<n;j++) S+=A[maxi][j]*A[j][minj];
    cout<<"S="<<S;
    return 0;
}
```

### ***Amaliy masalalar***

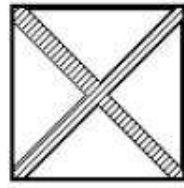
1. const int n = 20;  
float B[n][n];  
Qo‘shimcha matritsadan foydalanmagan holda B matritsaning transponerlangan ko‘rinishi  $B^T$  hosil qilinsin.
2. enum Oy{yan,fev,mar,apr,may,iyn,iyl,avg,sen,okt,noy,dek};  
enum Kun{dush,sesh,chor,paysh,juma,shanba,yaksh,null};  
Kun Kalendar[12][31];  
Kalendar massivi haftaning mos kunlari bilan to‘ldirilsin (mavjud bo‘lmagan sana null deb ko‘rsatilsin). Yil kabisa yili emas va 1-yanvar dushanba deb hisoblansin (Kalendar[yan][0]=dush; Kalendar[yan][1]=sesh;... Kalendar[fev][29]=null;... ).
3. float A[9][9],s;  
A matritsaning quyidagi rasmdagi bo‘yalgan sohalardagi elementlari



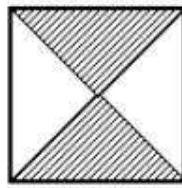
yig'indisi  $s$  topilsin.



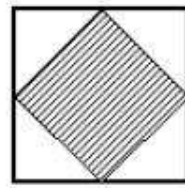
a)



b)



d)



e)

4.  $n$  natural soni va 5-chi tartibli haqiqiy turdagi kvadrat matritsa berilgan. Bu matritsaning  $n$ -darajasi topilsin ( $A^1=A$ ,  $A^2=AA$ ,  $A^3=A^2A$  va hokazo).
5. `const int n=20;`  
`float nuqta[n][2], d;`  
Matritsaning satr elementlarini tekislikdagi nuqtalarning koordinatalari deb qarab, shu nuqtalar orasidagi eng katta masofa topilsin.
6. `int A[10][10], B[9][9];`  
`int n, k; // 0 ≤ n ≤ 10, 0 ≤ k ≤ 10`  
Berilgan  $A$  matritsaning  $n$ -satri va  $k$ -ustunini o'chirish orqali  $B$  matritsa hosil qilinsin.
7. `const int n=8, m=12; int k, S[n][m];`  
 $S$  matritsaning "maxsus" elementlar soni -  $k$  aniqlansin. Element "maxsus" deyiladi, agar u o'zi joylashgan ustundagi boshqa elementlar yig'indisidan katta va o'zi joylashgan satrda chapdagi elementlardan katta, o'ngdagilaridan esa kichik bo'lsa.
8. `int k; char C[10][15];`  
Berilgan  $C$  matritsadagi har xil belgilar soni -  $k$  aniqlansin (takrorlanuvchi belgilar bitta deb hisoblansin).
9. 5 ta satr va 7 ta ustundan iborat haqiqiy turdagi matritsa berilgan. Uning satrlari kamaymaydigan ko'rinishda tartiblansin.
10. O'lchami  $10 \times 5$  bo'lgan haqiqiy turdagi matritsa berilgan. Matritsa satrlarining eng katta elementlari o'sishi bo'yicha tartiblansin.
11. Matritsaning elementi egar nuqta deyiladi, agarda u bir vaqtning o'zida shu element joylashgan satrdagi eng kichigi va ustundagi eng kattasi bo'lsa yoki aksincha.  $10 \times 15$  o'lchamli butun turdagi matritsaning egar nuqtasi indeksi chop etilsin.
12. Elementlari bir-biriga teng bo'lmagan haqiqiy turdagi 7-tartibli kvadrat matritsa berilgan. Uning eng katta element va eng kichik elementlar o'rinlari bilan almashtirilsin.
13. Butun turdagi 10-tartibli kvadrat matritsa ortonormal yoki yo'qligi aniqlansin. Matritsa ortonormal deyiladi, agar turli satrlari skalyar ko'paytmasi 0 ga teng va satrni o'z-o'ziga ko'paytmasi 1 ga teng

bo'lsa.

14. Natural  $n$  soni va  $n \times n$  o'lchamli haqiqiy turdagi  $A$  matritsa berilgan. Quyidagi formula yordamida  $A$  matritsaga teskari matritsa topilsin:

$$A_k^{-1} = A_{k-1}^{-1} (2E - A * A_{k-1}^{-1}),$$

bu yerda  $A$  - berilgan matritsa;  $E$  - birlik matritsa;  $A_k^{-1}$  - teskari matritsaning  $k$  - yaqinlashishi,  $A_0^{-1} = E$ . Teskari matritsa berilgan  $\varepsilon (\varepsilon > 0)$  aniqlikda hisoblansin.

15. Natural  $n$  soni va butun turdagi  $n$ -tartibli kvadrat matritsa berilgan. Matritsa elementlari monoton ketma-ketlik hosil qiluvchi (monoton kamayuvchi yoki o'suvchi) satrlar nomerlari topilsin.
16. Butun turdagi  $n$ -tartibli kvadrat matritsaning absolyut qiymati bo'yicha eng katta elementlari topilsin. Shu elementlar joylashgan ustun va satrlarni o'chirish orqali yangi matritsa qurilsin.
17. Natural  $n$  soni va elementlari faqat 0,1, 2 va 3 sonlaridan tashkil topgan  $n$ -tartibli  $A$  matritsa berilgan. Elementlari har xil sondan iborat barcha  $a_{i,j}, a_{i,j+1}, a_{i+1,j}, a_{i+1,j+1}$  to'rtliklar miqdori topilsin.
18. Haqiqiy turdagi 9-tartibli kvadrat matritsa berilgan. Shunday matritsa hosil qilinsinki, bunda boshlangich matritsada mos element o'z satridagi diagonal elementdan kichik bo'lmasa bir, aks holda 0 qabul qilsin.

19. `const int n=10;`

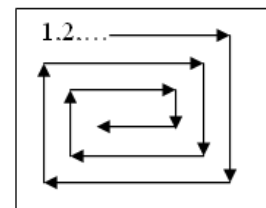
`float A[n][n], b[n], c[n];`

Simmetrik matritsa o'ng uchburchagi  $(n+1)*n/2$  elementlari bilan berilgan. Matritsaning 1-satrdan  $n$ -element, 2-satrdan  $n-1$ -element va oxirida  $n$ -satrdan 1-elementlar  $b$  vektorga yuklanib,  $c=A*b$  hisoblansin.

20. `const int n=7;`

`int A[n][n];`

Butun turdagi  $A$  matritsaga 1, 2, ..., 49 sonlari o'ng yuqori burchakdan boshlab ichkariga spiral bo'yicha joylashtirilsin (rasmga qarang).



21. `const int n=20;`

`int S[n][n];`

$S$  matritsaning elementlarini markaz atrofida  $90^\circ$  ga soat millariga teskari yo'nalishda burish bilan qayta aniqlansin.

22. `const int n=10,m=12;`

`int S[n][m];`

$S$  matritsaning lokal minimumlari chop etilsin. Matritsaning  $s_{ij}$  elementi lokal minimum deyiladi, agar u o'z atrofidagi barcha elementlardan kichik bo'lsa.

23. Butun toq  $n$  ( $n > 1$ ) soni berilgan.  $n$  - tartibli ilohiy kvadrat deb 1 dan  $n^2$  bo'lgan sonlarni kvadrat jadvalda shunday joylashtirishga aytiladiki, ixtiyoriy bir satr va ustundagi sonlar yig'indilari ikkita asosiy diagonallar yig'indisiga teng bo'la-

15	8	1	24	17
16	14	7	5	23
22	20	13	6	4
3	21	19	12	10
9	2	25	18	11

di. Jadvalni to'ldirish quyidagi tartibda amalga oshiriladi: birinchi qatorning o'rtasiga 1 soni joylashtiriladi, keyin yuqori chap tomondagi katakka 2 joylashtiriladi va hakoza. Agar vertikal bo'yicha jadval chegarasidan chiqiladigan bo'lsa, pastdagi (yuqoridagi) kataklar ulangan hisoblanib o'tish amalga oshiriladi. Xuddi shunday holat gorizintal o'tishlarda ham o'rinli hisoblalanadi. Misol tariqasida  $n=5$  uchun jadval to'ldirilishi rasmda kelirilgan.

## 14. Statik massivlar funksiya argumenti sifatida

### *Namunaviy masala*

Berilgan butun sondagi  $N$  pul miqdori 100,50,25,10,5,3,1 qiymatdagi tangalarning eng kam miqdori bilan yig'ilsin.

### *Yechish usuli*

Berilgan  $N$  pul miqdori massivning 0-elementidan kichik bo'lmasa undan massivning 0-elementi ayriladi va u ekranga chop qilinadi. Bu jarayon toki qolgan pul miqdori massivning 0-elementidan kichik bo'lguncha davom etadi. Xuddi shu jarayon qolgan pul miqdori uchun massivning 1-elementi uchun, keyin 2-elementi uchun va shu tariqa massivning keyingi elementlari uchun takrorlanadi. Pul miqdori 0 bo'lganda jarayon to'xtaydi va ekranda zarur tangalarning tartiblangan ro'yxati chop etiladi.

### *Ilova matni*

```
#include <iostream.h>
void Tangalar(unsigned int pul,unsigned int tanga[])
{
for(int i=0;pul>=1;)
if(pul>=tanga[i])
{
pul-=tanga[i];
cout<<tanga[i]<<'t';
}
else i++;
}
int main()
```

```

{
unsigned int Pul,tanga[7]={100,50,25,10,5,3,1};
cin>>Pul;
Tangalar(Pul,tanga);
return 0;
}

```

Ilova ishga tushirib, 239 soni kiritilsa, ekranga quyidagilar chop etiladi:

```
100 100 25 10 3 1
```

### *Amaliy topshiriqlar*

1. char Matn[72];  
char Shifr[52];  
Belgi turidagi Matn massivida lotin harflaridan tashkil topgan matn berilgan. Matn 52 ta lotin harfi uchun aniqlangan Shifr yordamida shifrlansin. Bu yerda Shifr[0] - 'A' ning, Shifr[1] - 'B' ning, ..., Shifr[51]- 'z' ning shifri bo'lgan belgi.
2. enum Hafta\_kuni{yaksh,dush,shesh,chor,paysh,juma,shanba};  
Hafta\_kuni Yil[365];  
Agar 1 yanvar kuni chorshanba bo'lsa (Yil[0]=chor, Yil[1]=paysh va hokazo), Yil massivining har bir elementiga, kabisa bo'lmagan yil hisobi bo'yicha i-kuniga mos keluvchi haftaning nomi (qiymati) berilsin.
3. Raqamlardan iborat va nuqta bilan tugaydigan matn berilgan. Matndagi eng ko'p uchraydigan raqam chop qilinsin (agar bunday raqamlar bir nechta bo'lsa, ulardan ixtiyoriy bittasi chop qilinsin).
4. Har biri 30 ta butun sondan iborat ikkita ketma-ketlik berilgan. Birinchi ketma-ketlikning ikkinchi ketma-ketlikka kirmagan sonlari ichidan eng kichigi topilsin (bunaqa sonlardan kamida bittasi mavjud deb hisoblan-sin).
5. const int n=10; float D[n][n],S;  
Berilgan D matritsa uchun  $S = \sum_{k=1}^n \max_{1 \leq i, j \leq k} D_{ij}$  hisoblansin.
6. Har biri n ta haqiqiy elementlardan tashkil topgan x,y va z vektorlar uchun quyidagi ifoda hisoblansin (n – berilgan o'zgarmas son).  

$$w = \begin{cases} \prod_i (\sin(x_i) + 2), & \text{agar } \prod_i (1 - y_i^2) > 0.5; \\ \prod_i (1 - z_i^2), & \text{aks holda.} \end{cases}$$
7. int A[15][20], b[15];  
Berilgan A matritsadan b vektor hosil qilinsin. Quyidagi ko'rsatilgan

shartlar bajarilsa,  $b[k]$  elementi true, aks holda false qiymat qabul qilsin:

- a) A matritsaning k-satri nollardan iborat;
- b) A matritsaning k-satr elementlari kamayish bo'yicha tartiblangan;
- d) A matritsaning k-satri simmetrik.

8. `const int n=20; char Screen[n][n];`

Berilgan Screen matritsasi elementlarini markaz atrofida  $90^\circ$  ga soat millariga teskari yo'nalishda burish orqali qayta aniqlansin.

9. Elementlari butun sonlardan iborat 10-tartibli kvadrat matritsa ortonormal yoki yo'qligi aniqlansin. Matritsa ortonormal deyiladi, agar turli satrlarni skalyar ko'paytmasi 0 ga teng, satrni o'ziga ko'paytmasi 1 ga teng bo'lsa.

10. Elementlari butun sonlardan iborat 9-tartibli kvadrat matritsa "*sehrli kvadrat*" ekanligi, ya'ni ixtiyoriy bir satr yoki ustunlar bo'yicha elementlar yig'indilari o'zaro tengligi aniqlansin.

11. Kichik lotin harflaridan iborat so'zlar ketma-ketligi berilgan. So'zlar bir-biridan vergul bilan ajratilgan va oxirgi so'zdan keyin nuqta qo'yilgan.  $a_i$  va  $b_i$  juftliklar ichida eng ko'p uchraydigan harflar juftligi aniqlansin (bu yerda  $a_i$  - ketma-ketlikdagi i-so'zning birinchi harfi,  $b_i$  esa oxirgi harfi).

12. Koeffisientlari haqiqiy turda bo'lgan  $n$  - tartibli "*uchburchak*" ko'rinishidagi chiziqli tenglamalar sistemasi yechilsin:

$$\left. \begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1 \\ a_{22}x_2 + \dots + a_{2n}x_n &= b_2 \\ &\dots \\ a_{nn}x_n &= b_n \end{aligned} \right\}$$

13. `char suz1[10], suz2[10];`

Berilgan suz1 massividagi belgilarning o'rinlarini almashtirish orqali suz2 so'zini hosil qilish mumkin yoki yo'qligi aniqlansin.

14. `const n=4,m=5; int A[n][m], B[n][m], C[n,m], N[n][m], D[2*n][3*m];`

Berilgan A,B va C matritsalar orqali D matritsani quradigan

`constr(A,B,C,D)` funksiya tuzilsin. D matritsa  $D = \begin{pmatrix} A & B & C \\ B & N & A \end{pmatrix}$  ko'rinishida

aniqlanadi. Bunda N nol matritsa.

15. Uchta haqiqiy turdagi 4-tartibli kvadrat matritsalar berilgan. Ularning orasida normasi eng kichik bo'lgani chop etilsin (bunday matritsa bitta deb hisoblansin). Matritsaning normasi sifatida uning elementlari absolut qiymatlarining maksimumi olinsin.

16. Natural  $p$  soni va 4-tartibli haqiqiy turdagi A,B va C kvadrat matritsalar

berilgan.  $(ABC)^p$  hosil qilinsin

17. Haqiqiy turdagi  $10 \times 20$  o'lchamli  $A, B$  va  $C$  matritsalar berilgan. Quyidagi kattalik hisoblansin:

$$\frac{\|A\| + \|B\| + \|C\|}{\|A + B + C\|}, \text{ bu yerda } \|D\| = \max_j |D_{1,j}| + \max_j |D_{2,j}| + \dots + \max_j |D_{10,j}|.$$

18. Ikkita 10-tartibli butun turdagi kvadrat matritsa berilgan. Matritsa diagonaliga nisbatan bir marta akslantirish orqali biridan ikkinchisini hosil qilish mumkinmi?
19.  $2^{500}$  va  $1!+2!+3!+\dots+100!$  sonlarining o'nlik yozuvidagi barcha raqamlari chop etilsin. (Ko'rsatma: "uzun" natural sonlarni raqamlardan iborat massiv ko'rinishida ifodalab, kerakli amallar bajarilsin.)
20.  $n$  ( $n=100$ ) ta haqiqiy sonlar berilgan. Fon Neyman usuli bilan ular o'sish tartibida joylashtirilsin: ikkita  $A$  va  $B$  massividan foydalaniladi. Berilgan sonlar  $A$  massiviga yoziladi; keyin yonma-yon sonlar tartiblanib ( $A_1$  va  $A_2$ ,  $A_3$  va  $A_4$  va hokazo)  $B$  ga yoziladi;  $B$  dagi ikkita yonma-yon turgan, tartiblangan juftlik olinib, ular tartiblangan to'rtlikka o'tkaziladi va yana  $A$  ga yoziladi; keyin  $A$  dan har ikkita yonma-yon to'rtlikni tartiblab, sakkizlik sifatida  $B$  ga yoziladi va hokazo.
21. 2 dan 20 gacha bo'lgan butun  $n$  va haqiqiy  $\varepsilon > 0$  sonlari berilgan. Quyidagi formula bilan aniqlanadigan  $n$ -tartibli  $T_n(x)$  Chebishev ko'phadining (12.10 masalaga qarang) barcha ildizlari  $\varepsilon > 0$  aniqlikda hisoblansin.  
*Izoh:* agar  $(-1, 1)$  intervalda  $x_1 < x_2 < \dots < x_k$  -  $T_k(x)$  ko'phadning ildizlari bo'lsa, u holda  $T_{k+1}(x)$  ko'phad quyidagi  $(-1, x_1), (x_1, x_2), \dots, (x_k, 1)$  intervallarda bittadan ildizga ega bo'ladi.
22. Tasodifiy sonlarni hosil qiluvchi funksiyadan foydalangan holda  $n$  ta o'zaro teng bo'lmagan butun sonlar massivini natija sifatida qaytaradigan funksiya tuzilsin.
23. Tasodifiy sonlarni hosil qiluvchi funksiyadan foydalangan holda butun turdagi  $n$ -tartibli kvadrat matritsa hosil qilinsin ( $n=10$ ). Matritsaning eng katta va eng kichik elementlarini almashtiruvchi funksiya tuzilsin.

## 15. Dinamik massivlar funksiya argumenti sifatida

### *Namunaviy masala*

Berilgan  $n \times m$  o'lchamdagi matritsaning  $k$ -ustundagi va  $l$ -satrdagi eng katta elementlari topilsin ( $0 \leq k \leq m, 0 \leq l \leq n$ ).

### ***Yechish usuli***

O'Ichami ilova ishlash jarayonida kiritilishi sababli bosh funksiyada matritsa dinamik ravishda butun sonlar vektorlariga (matritsa satrlariga) ko'rsatgichlar vektori ko'rinishida aniqlanadi. Berilgan ustundagi va satrdagi maxsimal elementlarni topish uchun alohida funksiyalar aniqlanadi va zarur parametrlar ularga argument sifatida uzatiladi. Shuni qayd etish kerakki, 1-satr eng katta elementini topadigan funksiyani chaqirganda argument sifatida matritsaning mos satriga ko'rsatgichni uzatish yetarli, k-ustun maksimumini topishda esa funksiyaga argument sifatida matritsaga ko'rsatgichni uzatish kerak bo'ladi.

### ***Ilova matni***

```
int K_ustun_max(int n,int k,int **a)
{
    int max=a[0][k];
    for(int i=1;i<n;i++) if(max<=a[i][k]) max=a[i][k];
    return max;
}
int L_satr_max(int m,int *satr)
{
    int max=satr[0];
    for(int i=1;i<m;i++) if(max<=satr[i]) max=satr[i];
    return max;
}
int main()
{
    int n,m,k,l;
    int **a;
    cout<<"Matritsaning satrlar soni="; cin>>n;
    cout<<"Matritsaning ustunlar soni="; cin>>m;
    a=new int *[n]; // matritsani hosil qilish
    for(int i=0; i<n; i++) a[i]=new int[m];
    for(int i=0; i<n; i++) // matritsa elementlari qiymatlarini kiritish
        for(int j=0; j<m; j++) cin>>a[i][j];
    // matritsani chop etish
    cout<<"Matritsa ko'rinishi:\n";
    for(int i=0; i<n; i++)
        { for(int j=0; j<m; j++) cout<<a[i][j]<<' '; cout<<endl; }
    do { cout<<"k-ustun nomerini kiriting="; cin>>k; } while(k<0 || k>m);
    do { cout<<"l-ustun nomerini kiriting="; cin>>l; } while(l<0 || l>n);
    cout<<k<<"-ustun maximal elementi ="<<K_ustun_max(n,k,a);
```

```

cout<<'n'<<'\n'<<"-sitr maksimal elementi =";
cout<<L_sitr_max(m,a[l]);
for(int i=0; i<n; i++) delete[]a[i];
delete [] a;
return 0;
}

```

### *Amamliy topshiriqlar*

1. int \*s;  
Berilgan s vektor kattalikning indeksleri quyidagilarga teng bo'lgan elementlari chop qilinsin:  
a) ikkinchi darajalari: (1,2,4,8,16,...);  
b) to'liq kvadratlar: (1,4,9,16,25,...);  
d) Fibonachchi sonlari: (1,2,3,5,8,13,...);
2. float \* x;  
Berilgan x vektor orqali x' vektor quyidagi qoida bo'yicha hosil qilinsin (x'\_k hosil bo'lgan vektorning k-elementi qiymati):  
a)  $x'_k = \max \{x_i\}$  bunda  $1 \leq i \leq k$ ;  
b) vektor elementlari teskari tartibda joylashtirilsin;  
d)  $x'_1 = x_1, x'_n = x_n, x'_k = (x_{k-1} + x_k + x_{k+1})/3, k = 2, 3, \dots, n-1$ ;  
e) vektor elementlari siklik ravishda p taga chapga surilsin.
3. float \* x;  
Berilgan x vektorning barcha manfiy elementlari uning boshiga o'tkazilsin. Hosil bo'lgan vektorda manfiy va qolgan elementlarining boshlang'ich o'zaro joylashuvi saqlansin (qo'shimcha vektordan foydalanilmasin).
4. int \* x,\*y,\*z;  
Berilgan x va y vektorlarning har birida elementlar kamaymaydigan tartibda joylashgan. Bu ikki vektorni birlashtirib, shunday z vektor hosil qilinsinki, uning elementlari ham kamaymaydigan tartibda bo'lsin.
5. char \*suz1, \*suz2; bool teng;  
Massiv ko'rinishida berilgan suz1 va suz2 so'zlarning har birida belgilar takrorlanib kelmaydi deb hisoblagan holda, ularda qatnashayotgan belgilarning joylashuvi bilan farq qilsa, teng o'zgaruvchiga true, aks holda false qiymat berilsin.
6. float \*x;  
Berilgan x vektor almashtirish usuli orqali kamaymaydigan ko'rinishda tartiblansin ("*pufakchalar usuli*"). Bu usulning 1-qadamida qo'shni  $x_k$  va  $x_{k+1}$  ( $k=1,2,3,\dots,n-1$ ) elementlar ketma-ket solishtiriladi va agarda



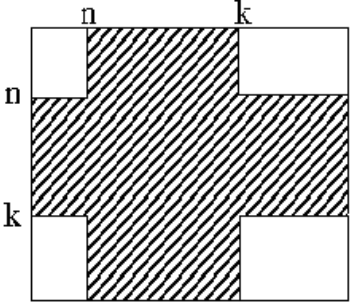
$x_k > x_{k+1}$  bo'lsa, ularning o'rnini almashtiriladi. Shu yo'l bilan eng katta elementni vektorning oxirida joylashuviga erishiladi. Keyingi qadamda bu usul oxirgi elementdan tashqari barcha elementlarga qo'llaniladi va h.k. Jarayonning birorta qadamida elementlarni almashtirish ro'yi bermasa tartiblash to'xtatilsin.

7. `float *x;`  
Berilgan  $x$  vektor orasiga qo'yish usuli orqali kamaymaydigan ko'rinishda tartiblash. Bu usulda: vektorning birinchi  $k$  ta elementi kamaymaydigan ko'rinishda tartiblangan deb hisoblanadi;  $k+1$ -elementi olinadi va u birinchi  $k$  ta element orasiga shunday joylashtiriladiki, hosil bo'lgan  $k+1$  ta element tartiblangan bo'ladi. Bu usul  $k$  o'zgaruvchining  $0$  dan  $n-2$  gacha qiymatlari uchun takrorlanadi.
8. Nuqta bilan tugaydigan kichik lotin harflaridan iborat matn berilgan. Shu matnga faqat bir marta kiruvchi barcha harflar alfavit tartibida chop qilinsin.
9. `float **A, **B, **C; // n*n o'lchamli matritsalar`  
`float *x, *y; // n o'lchamli vektor`  
Berilgan natural  $n$  uchun quyidagilar hisoblansin:  
a)  $C=A+B$ ;    b)  $y=Ax$ ;    d)  $C=AB$ .
10. Berilgan butun  $n$  va  $m$  boyicha  $n \times m$  o'lchamli haqiqiy turdagi dinamik matritsa berilgan. Uning satrlari kamaymaydigan ko'rinishda tartiblansin:  
a) birinchi elementlar bo'yicha;    b) elementlar yig'indisi bo'yicha;  
d) eng katta elementlari bo'yicha.
11. Shaxmat musobaqasida qatnashgan  $n$  ta shaxmatchining natijalari  $T$  jadvalda berilgan ( $n > 2$ ):  
`enum Uyin_Natijasi{Y,D,M,X};`  
Uyin\_Natijasi turidagi  $n \times n$  o'lchamli dinamik massiv (jadval) hosil qilinsin. Uning qiymatlari quyidagicha aniqlasin:  $jadval[i][j]=Y$ , agar  $i$ -ishtirokchi  $j$ -ishtirokchi ustidan g'alaba qozongan bo'lsa, (bunda  $jadval[j][i]=M$ ),  $jadval[i][j]=D$  va  $jadval[j][i]=D$ , agar  $i$ - va  $j$ -ishtirokchilar durang o'ynagan bo'lsa, hamda  $jadval[i][i]=X$ .  
Jadvalning ko'rinishi quyidagicha bo'lishi mumkin (rasmga qarang). Yutuq uchun 3 ochko, durang uchun 1 ochko, mag'lubiyat uchun 0 ochko beriladi. Ishirokchilar nomerlari ularning to'plagan ochkolari bo'yicha o'smaydigan tartibda chop qilinsin.

(	X	Y	M
M	X	D	D
Y	D	X	)
12. Berilgan  $n \times m$  o'lchamli butun turdagi dinamik ravishda yaratilgan matritsaning egar nuqtasi indeksleri chop etilsin. Matritsa elementi "egar nuqta" deyiladi, agarda u bir vaqtning o'zida shu element joy-

lashgan satrdagi eng kichik va ustundagi eng katta element bo'lsa yoki aksincha.

13.  $n$  o'lchovli chiziqli fazoda  $m$  ta vektor koordinatalari bilan berilgan. Bu vektorlar chiziqli erkli bo'ladimi? Vektorlar chiziqli erkli deyiladi, agar har qanday  $i \neq j (1 \leq i, j \leq m)$  uchun  $x_i = \alpha x_j$  sharti o'rinli bo'lmasa, bu yerda  $\alpha > 0$ .
14. Elementlari  $n$  ta haqiqiy sonlardan iborat  $x, y$  va  $z$  vektorlar berilgan.  $(a, a)-(b, c)$  kattalik hisoblansin, bu yerda  $a$  vektor berilgan vektorlar ichidan eng katta minimal elementga ega bo'lgan vektor (bunday vektor yagona deb hisoblansin),  $b$  va  $c$  qolgan ikkita vektorlar,  $(p, q)$ -  $p$  va  $q$  vektorlarning skalyar ko'paytmasini bildiradi.
15.  $9 \times 4$  o'lchamli uchta butun turdagi matritsalar dinamik ravishda aniqlanib, qiymatlari berilgan. Faqat nollardan iborat satrlari eng ko'p bo'lgan matritsa chop etilsin (agar bunday matritsalar bir nechta bo'lsa, barchasi chop etilsin).
16.  $n$  ta butun elementli dinamik aniqlangan  $x$  vektorning elementlari o'sish bo'yicha tartiblangan.  $k$  o'zgaruvchiga berilgan  $p$  soniga teng bo'lgan  $x$  vektori elementining tartib nomeri berilsin, agarda bunday element bo'lmasa,  $k$  o'zgaruvchisiga  $-1$  berilsin.  
Masalani yechish uchun quyidagi ikkilik (binar) qidirish usuli qo'llanilsin:  $p$  sonni vektor o'rtasidagi yoki unga eng yaqin element bilan solishtiriladi; agar bu sonlar teng bo'lsa qidirish to'xtatiladi; agar  $p$  son elementdan kichik bo'lsa, u holda  $p$  sonini vektorning chap yarmidan izlash kerak, aks holda o'ng yarmidan; vektorning tanlangan yarmiga yuqoridagi algoritm qo'llaniladi.
17. Butun turdagi  $N \times N$  o'lchamli dinamik ravishda aniqlangan qiymatlardan tashkil topgan  $A$  matritsaning bo'yalgan sohasidagi katta elementini topuvchi  $\max\_Paint(A, n, k)$  funksiyasi tuzilsin. Bu yerda  $A$ -matritsa,  $n$  va  $k$ -indekslar bo'lib (rasmga qarang), ularning to'g'ri kiritilishi nazorat qilinsin.
 


18.  $n$  ta satr va  $m$  ta ustundan iborat  $A$  va  $B$  matritsalarining maksimal elementlarini almashtiruvchi  $\text{swap}(A, B)$  funksiyasi tuzilsin. Har bir matritsada maksimal element bitta deb hisoblansin.
19.  $p(x) = p_0 x^n + p_1 x^{n-1} + \dots + p_{n-1} x + p_n$  ko'phad koeffitsiyentlari bilan berilgan. Berilgan haqiqiy  $x$  va  $a$  sonlari uchun quyidagilar hosil qilinsin:
  - a)  $(x-a)p(x)$  ko'phadning koeffitsiyentlaridan tashkil topgan  $r$  vektor;

b)  $p(x+a)$  ko'phadning koeffitsiyentlaridan tashkil topgan  $q$  vektor.

## 16. Satrlar

### *Namunaviy masala*

ASCIIZ satr ko'rinishida berilgan gapdagi simmetrik so'zlar chop etilsin. Gap nuqta bilan tugaydi va undagi so'zlar bir-biridan ',' (vergul) yoki '\_' (probel) bilan ajratilgan.

### *Yechish usuli*

Qo'yilgan masalani yechish uchun takrorlanuvchi ikkita amalni bajarish kerak bo'ladi:

- 1) satrdan so'zlarni (satr ostilarini) ajratib olish;
- 2) satr ostini simmetriklikka tekshirish, agar natija ijobiy bo'lsa, satrni chop etish.

Satrdan satr ostilarini ajratib olish uchun `<string.h>` kutubxonasi mavjud `strtok()` funksiyasidan foydalanish mumkin. `strtok()` prototipi

```
char* strtok(char* str, const char * delim);
```

ko'rinishida bo'lib, u `str` satrdagi `delim` satrda (ro'yxatda) berilgan ajratuvchilar oraliqiga olingan satr qismini qiymat sifatida qaytaradi. Funksiya qaytargan satr qismini simmetriklikka tekshirish uchun satrning (vektorning) markaziga nisbatan teng uzoqlikda joylashgan elementlarning o'zaro bir xil ekanligi asosida tekshiradigan mantiqiy funksiyani tuzish orqali yechiladi.

### *Ilova matni*

```
#include <iostream.h>
#include <string.h>
const N=80;
bool Simmetrik(const char * s)
{
    int n=strlen(s);
    for(int i=0;i<=n/2;i++)
        if(s[i]!=s[n-i-1]) return false;
    return true;
}
int main()
{
    char Gap[N];
    gets(Gap);
    char Ajratuvchi[]=" ,.";
}
```

```

char* Gap_Osti;
Gap_Osti=strtok(Gap,Ajratuvchi);
if(Gap_Osti && Simmetrik(Gap_Osti))cout<<Gap_Osti<<'\n';
while(Gap_Osti)
{
    Gap_Osti=strtok(NULL,Ajratuvchi);
    if(Gap_Osti && Simmetrik(Gap_Osti))cout<<Gap_Osti<<'\n';
}
return 0;
}

```

### ***Amaliy topshiriqlar***

1. char t[100];  
Berilgan t satrdagi bir-biridan farqli bo‘lgan kichik lotin harflari alfavit tartibida chop qilinsin.
2. char s[80];  
Berilgan s satr quyidagicha o‘zgartirilsin:
  - a) satrdagi barcha “bir” satrostilari “\_1\_” satrostiga almashtirilsin (bu yerda ‘\_’ - probel);
  - b) satrga kiruvchi oxirgi ‘x’ harfi (agarda u mavjud bo‘lsa) “ks” ga almashtirilsin;
  - d) satrga kiruvchi barcha “th” belgilar juftliklari o‘chirilsin;
  - e) satrga kiruvchi har bir ‘q’ harfidan keyin ‘u’ harfi qo‘shilsin.
3. char s1[10],s2[10],s3[11];  
O‘nlik sanoq sistemasidagi butun s1, s2 sonlarning razryad bo‘yicha yig‘indisi s3 o‘zgaruvchisida hosil qilinsin.
4. char bayt[8];  
Berilgan butun n ( $0 \leq n \leq 127$ ) soniga qarama-qarshi sonning (manfiy sonning) qo‘shimcha koddagi ikkilik ko‘rinishi bayt massivida hosil qilinsin.
5. char gap[80];  
Berilgan gapdagi so‘zlar bir-biridan ‘,’ yoki ‘\_’ (probel) belgisi bilan ajratilgan va gap nuqta bilan tugaydi (gap o‘zgaruvchisining qiymati).  
Quyidagilar chop qilinsin:
  - a) ketma-ketlikda faqat bir marta uchragan so‘zlar;
  - b) ketma-ketlikda bir necha bor uchragan so‘zlar;
  - d) ketma-ketlikdagi barcha so‘zlar alfavit tartibida.
6. char gap[80];  
Berilgan gapdagi so‘zlar bir-biridan ‘,’ yoki ‘\_’ (probel) belgisi bilan ajratilgan va gap nuqta bilan tugaydi (gap o‘zgaruvchisining qiymati).  
Quyidagi shartlarni bajaruvchi so‘zlar chop qilinsin:



harflar chastotalarining kamayishi bo'yicha chop qilinsin.

15. `const int n=20,m=40; char matn[n][m];`  
Har bir satri  $m$  belgigacha bo'lgan  $n$  ta satrdan iborat matn berilgan. Matn  $m$  uzunligidagi o'ng va chap tomondan tekislangan satrlarga o'tkazilsin. Bunda satrda kamida ikkita so'z bor deb hisoblanadi. Qisqa satrlar so'zlar orasiga probellarni qo'yish bilan to'ldiriladi.
16. Berilgan so'zni quyidagi qoida yordamida ikkiga bo'lishni amalga oshiring (keyingi satrga o'tkazish uchun):
  - 1) ketma-ket kelgan ikkita unli harfni ajratish mumkin, agar birinchisidan oldin undosh harf va ikkinchisidan keyin kamida bitta harf kelsa;
  - 2) ketma-ket kelgan ikkita undosh harfni ajratish mumkin, agar birinchisidan oldin unli harf va ikkinchisidan keyin soz bo'lagidagi kamida bitta unli harf kelsa ('' - ajratish belgilari o'zidan oldindagi harf bilan bitta deb qaraladi);
  - 3) agar 1) va 2) qoidalarni qo'llash mumkin bo'lmasa so'zni shunday bo'lish kerakki, unig birinchi qismi kamida bitta unli harfni o'z ichiga olishi va unli bilan tugallanishi kerak, ikkinchi qism kamida bitta unli harfni o'z ichiga olishi kerak.
17. Berilgan  $n$  butun soni ( $0 \leq n \leq 999999$ ) so'zlar orqali yozilsin. Masalan, 15 soni "*o'n besh*" ko'rinishida.
18. Ikkilik sanoq sistemasidagi son satr ko'rinishida berilgan. Shu sonning 8 va 16 lik sanoq sistemasidagi ko'rinishi chop qilinsin.
19. `char x[10];`  
Berilgan  $x$  satri identifikator yoki yo'qligi aniqlansin.
20. `typedef char Misra[60]; typedef Misra Turtlik[4]; Turtlik she_r;`  
Berilgan `she_r` turtlikda qofiya bor yoki yo'qligi aniqlansin.
21. `const int n=60;`  
`char jumla1[n], jumla2[n];`  
Berilgan `jumla1` va `jumla2` satrlar bir-biridan faqat ulardagi so'zlarning gapdagi joylashuv o'rni bilan farq qilishi aniqlansin. So'zlar bir-biridan ' , ' yoki ' \_ ' (probel) belgisi bilan ajratilgan.
22. `char s[60];`  
Berilgan satrda polindrom so'zlar mavjud bo'lsa, ular chop qilinsin, aks holda bunday so'z yo'qligi haqida xabar berilsin (satrdagi so'zlar bir-biridan bitta yoki bir necha probel orqali ajratiladi).
23. `const int n=30; typedef char FISH[40]; FISH Guruh[n]; unsigned int Tug_Yil[n];`  
O'quv guruhi (Guruh) talabalarining ro'yxati quyidagi tartibda: talabaning familiyasi, ismi va otasi ismining bosh harflari hamda tug'ilgan yili nuqta bilan ajratilgan ko'rinishda berilgan. Ro'yxat lotin alifbosidagi harflar joylashuvi bo'yicha tartiblansin. Tatriplash talaba familiyasi va inisiallari

- bo'yicha amalga oshirilsin. Agar bir xil familiya va initsiilli talabalar uchrasa, ular ro'xatda tug'ilgan yilini (Tug\_Yil) kamayishi bo'yicha joylashtirilsin.
24. `const int n=60; char Jumla[n];`  
Berilgan jumladagi (Jumla) har bir so'zning birinchi harfi bosh harfga almashtirilsin. So'zlar bir-biridan ',' yoki '\_' (probel) belgisi bilan ajratilgan.
  25. `const int n=60; char Jumla[n];`  
Lotin harflaridan tashkil topgan Jumla satri berilgan. Satrdagi har bir harf alfavitdagi o'zidan keyingi harf bilan almashtirish orqali shifrlansin va qayta tiklansin. ('A' harfi o'tkazilsin 'B' harfiga, 'a'→'b', 'B'→'C', 'z'→'a' va hokazo). Almashtirishda harflar registri inobatga olinsin. Satrdagi ajratuvchilar (probel, ',' va boshqalar) o'zgartirilmasin.
  26. `const int n=60; char Jumla[n];`  
Lotin harflaridan tashkil topgan Jumla satri berilgan. Satr, undagi juft o'rinda turgan harflarni satr boshiga o'tkazish va toq o'rndagi harflarni teskari tartibda qayta joylashtirish orqali shifrlansin va qayta tiklansin. Masalan, "Programma" satri "rgamamroP" satriga aylanadi.
  27. `char *fam[ ]; char *ism[ ]; int tel[ ];`  
Yuqorida keltirilgan massivlarda mos ravishda familiyalar, ismlar va telefon raqamlar berilgan. So'ralgan ismga (famiyaga) mos telefon raqami yoki so'ralgan telefonga mos ismni (famiyani) topuvchi, agarda bunday ma'lumot mavjud bo'lmasa, uning yo'qligi haqida xabar beruvchi ilova tuzilsin.
  28. Butun musbat n, m sonlari va `char a[n][m]` massiv berilgan. a massivning har bir satridagi raqamlar miqdori aniqlansin.
  29. `char **familiya, **ism, **telefon;`  
Yuqorida keltirilgan massivlarda mos ravishda familiyalar, ismlar va telefon raqamlar berilgan (bu massivlarning har qatorlari o'zaro bir qiymatli bo'g'liq).  
a) familiya massivining alfavit bo'icha tartiblanishiga mos ravishda ism va telefon massivlari ham tartiblansin;  
b) telefon raqamlarining boshlangich bo'lagi berilgan raqamlar bilan ustma-ust tushuvchi mos familiya va ismlar chop qilinsin.
  30. Satr va n (n>0) natural soni berilgan. Satrdagi n-so'z o'chirilsin.
  31. Berilgan matn teskariga o'girilsin, yani undagi har bir so'z teskari yozilsin (masalan: "UzMU 1-kurs talabasi" → "isabalat sruk-1 UMzU").
  32. Berilgan S satrda ikkita bir xil belgilar takrorlanmaydigan eng uzun satrosti chop etilsin.

33. Fibonnachchi harflari ketma-ketligi hosil qilinsin:  $S_1 = "a", S_2 = "b"$  va  $S_n = S_{n-1}S_{n-2}, (n > 2)$ , yani  $S_n$  so'zi  $S_{n-1}$  so'zga  $S_{n-2}$  so'zni ulashdan hosil bo'ladi. Berilgan butun  $k (k > 2)$  uchun  $S_k$  so'zi qurilsin.

## 17. Rekursiv funksialar

### *Namunaviy masala*

Tezkor tartiblash usuli yordamida  $n$  ta ( $n > 0$ ) butun sonlardan iborat  $x$  massiv kamaymaydigan ko'rinishda tartiblansin.

### *Yechish usuli*

Tezkor tartiblash usulining mohiyati shundaki,  $x$  massivning birorta elementi (masalan, o'rtadagi elementi)  $x_i$  olinadi. Massiv elementlari shunday almashtiriladi, natijada massivning  $x_i$  element joylashgan joydan chap tomonidagi qismida faqat undan katta bo'lmagan elementlar, o'ng tomonda undan kichik bo'lmagan elementlar joylashib qoladi, y'ani  $x_i$  tartiblangan ketma-ketlikdagi o'zining yakuniy o'rnini topadi. Keyingi qadamda  $x_i$  element joylashgan joydan chap va o'ng tomondagi massiv qismlariga yuqoridagi usul qo'llaniladi va hokazo.

### *Ilova matni*

```
void Tezkor_Tartiblash(int *a, int i_chap, int i_ung)
{
    int chap, ung, ai;
    if (i_chap >= i_ung) return;
    chap = i_chap;
    ung = i_ung;
    ai = a[(chap+ung)/2];
    while(chap <= ung)
    {
        while(a[chap] < ai) chap++;
        while(a[ung] > ai) ung--;
        if(chap <= ung)
        {
            int vaqtincha = a[chap];
            a[chap] = a[ung];
            a[ung] = vaqtincha;
            chap++; ung--;
        }
    }
}
```



```

Tezkor_Tartiblash(a, i_chap, ung);
Tezkor_Tartiblash(a, chap, i_ung);
}
int main()
{
int *x;
int n;
cin>>n;
x=new int [n];
for (int i=0;i<n;i++) cin>>x[i];
cout<<"Berilgan massiv"<<endl;
for(int i=0;i<n;i++) cout<<x[i]<<"\t";
cout<<endl;
Tezkor_Tartiblash(x,0,n-1);
cout<<"Tartiblangan massiv:"<<endl;
for(int i=0;i<n;i++) cout<<x[i]<<"\t";
return 0;
}

```

### *Amaliy masalalar*

1.  $C_n^0 = C_n^n = 1; C_n^m = C_{n-1}^m + C_{n-1}^{m-1}$  formula bo'yicha  $C_n^m$  binom koeffisientini hisoblaydigan  $C(n,m)$  rekursiv funksiya tuzilsin, bunda  $0 \leq m \leq n$ .
2. Berilgan  $n$  ( $n > 1$ ) ta musbat sonlarning EKUBi topilsin.
3. Berilgan  $n$  ( $n > 1$ ) ta musbat sonlarning eng kichik umumiy karralisi (EKUK) topilsin.
4. `const int n=40; float x[n];`  
Berilgan  $x$  vektorning  $k$ - elementidan keyingi o'rindagi elementlari orasidagi eng kichik elementni topadigan  $\min1(k)$  yordamchi rekursiv funksiya qurilib, uning yordamida  $x$  vektorining eng kichik elementini topadigan  $\min(x)$  funksiyasi tuzilsin.

5. Quyidagi

$$x^n = \begin{cases} 1, & \text{agar } n = 0; \\ 1/x^{|n|}, & \text{agar } n < 0; \\ x \cdot x^{n-1}, & \text{aks holda.} \end{cases}$$

formula yordamida  $x^n$  kattalikni hisoblovchi  $\text{Daraja}(x,n)$  rekursiv funksiya tuzilsin. Bu yerda  $x$  ( $x \neq 0$ ) haqiqiy,  $n$ -butun son.

6. Berilgan ikkita  $a$  va  $b$  natural sonlar eng katta umumiy bo'luvchisi (EKUB) Eyler usulida hisoblansin:

$$\text{EKUB}(a,b) = \begin{cases} b, & \text{agar } b = 0; \\ \text{EKUB}\left(b, \begin{cases} a \\ b \end{cases}\right), & \text{aks holda.} \end{cases}$$

7. char satr[100];  
Berilgan satr satrning i-elementidan boshlanib, j-elementida tugaydigan qismini simmetrikligini tekshiradigan rekursiv mantiqiy simm(s,i,j) funksiya tuzilsin.
8. enum Ism=(Sayyora,...,Erkin,null);  
Berilgan b ismli odam a ismli odamning avlodi (bolasi, nevarasi, chevarasi va hokazo) bo'lishligini tekshiradigan avlod(a,b) mantiqiy funksiyasi tuzilsin. Masalani yechishda oldindan tuzilgan ona(x) va ota(x) funksiyalaridan foydalaning. Bunda ona(x) va ota(x) funksiyalari x ismli odamni mos ravishda onasi va otasining ismini, agar x ga mos keluvchi ota-ona to'g'risida ma'lumot bo'lmasa, yo'q (null) qiymatini qabul qiladi.
9. Agarda BolalarSoni(x) funksiyasi x ismli odamning farzandlar sonini, Bola(x,k) funksiyasi esa, x ismli odamning k-farzandining ismini bildirsa (bu yerda k - x odamning farzandlar sonidan oshmasligi kerak), ushbu funksiyalar yordamida yuqorida keltirilgan 17.8-masala yechilsin.
10. Kesmani teng ikkiga bo'lish usuli yordamida  $f(x)=0$  tenglamani  $[a,b]$  oraliqda eps aniqligidagi ildizini topadigan root(f,a,b,eps) rekursiv funksiyasi tuzilsin. Bu yerda  $\text{eps}>0, a<b, f(a)*f(b)<0, [a,b]$  oralig'ida  $f(x)$  - uzluksiz va monoton funksiya deb hisoblansin.
11. O'qish faylida manfiy son bilan tugaydigan, bo'sh bo'lmagan musbat haqiqiy sonlar ketma-ketligi berilgan. Musbat sonlar yig'indisini hisoblaydigan parametrsiz sum() rekursiv funksiya tuzilsin.
12. Matndagi (oxiri nuqta bilan tugagan) raqamlar sonini hisoblaydigan parametrsiz digits() rekursiv funksiya tuzilsin.
13. O'qish faylida oxiri nuqta bilan tugaydigan matn berilgan. Bu matnni teskari tartibda chop qiladigan teskari() rekursiv funksiya tuzilsin.
14. Nol bilan tugaydigan butun sonlar ketma-ketligi berilgan. Birinchi navbatda ketma-ketlikning barcha manfiy sonlari, so'ngra musbat sonlar chop qilinsin (ixtiyoriy tartibda).
15. O'qish faylida nuqta bilan tugaydigan matn berilgan. Uning formula ekanligi quyidagi qoidalar asosida tekshirilsin:  
 $\langle \text{formula} \rangle ::= \langle \text{raqam} \rangle | (\langle \text{formula} \rangle \langle \text{amal} \rangle \langle \text{formula} \rangle)$   
 $\langle \text{amal} \rangle ::= + | - | *$   
 $\langle \text{raqam} \rangle ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 .$
16. O'qish faylida quyidagi ko'rinishda mantiqiy ifoda xatosiz yozilgan.  
 $\langle \text{mantiqiy ifoda} \rangle ::= \text{true} | \text{false} | \langle \text{amal} \rangle (\langle \text{operandlar} \rangle)$

$\langle \text{amal} \rangle ::= ! \mid \&\& \mid \parallel$

$\langle \text{operandlar} \rangle ::= \langle \text{operand} \rangle \mid \langle \text{operand} \rangle, \langle \text{operandlar} \rangle$

$\langle \text{operand} \rangle ::= \langle \text{ mantiqiy ifoda} \rangle$

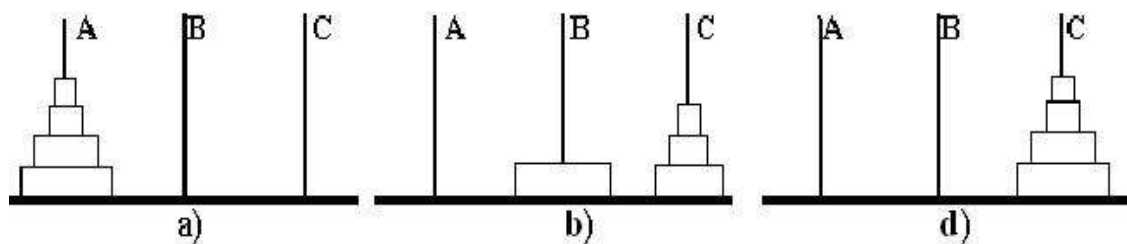
(“&&” va “||” amallarida operandlar soni ixtiyoriy bo‘lishi mumkin, ‘!’ amalida esa faqat bitta). Bu ifodaning qiymati hisoblansin. (Masalan,  $\&\&(\parallel(\text{false},!(\text{false})), \text{true},!(\text{true})) \rightarrow \text{false}$ .)

17. O‘qish faylidan nuqta bilan tugaydigan matn o‘qilsin. Uning tuzilishi quyidagi qoidani qanoatlantirishi tekshirilsin.

$\langle \text{matn} \rangle ::= \langle \text{element} \rangle \mid \langle \text{element} \rangle \langle \text{matn} \rangle$

$\langle \text{element} \rangle ::= a \mid b(\langle \text{matn} \rangle \mid [\langle \text{matn} \rangle ] \{ \langle \text{matn} \rangle \})$

18. “Xanoy minorasi” masalasi. Uchta A, B, C qoziq va n ta har xil o‘lchamli halqalar mavjud. Halqalar o‘lchamlari o‘shish tartibida 1 dan n gacha tartiblangan. Barcha halqalar A qoziqda a) rasmdagidek joylashtirilgan. A qoziqdagi barcha halqalarni C qoziqqa quyidagi qoidalarga amal qilgan holda rasmdagidek o‘tkazish talab etiladi: halqalarni bittadan ko‘chirish kerak va katta o‘lchamli halqani kichik o‘lchamli halqa ustiga qo‘ymaslik kerak.



Amallar ketma-ketligini chop etadigan (“halqa q dan r ga o‘tkazilsin” ko‘rinishida, bunda q va r – ‘A’, ‘B’ yoki ‘C’) masalani n ta halqa uchun yechadigan ilova tuzilsin.

Ko‘rsatma: halqalarni A dan C ga to‘g‘ri o‘tkazishda b), d) rasmlardagidek ko‘rinish uchraydi.

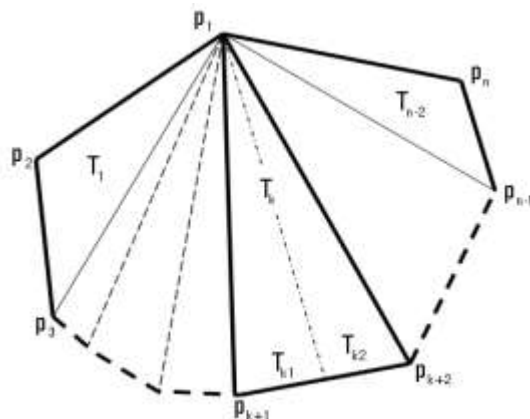
19. 5 ta har xil natural son berilgan. Bu sonlarni barcha o‘rin almashish holatlari chop qilinsin.
20. Shaxmat taxtachasida 8 ta farzin shunday joylashtirilsinki, ular bir-birini “urmasin”. Barcha shunday holatlar (92 ta) chop etilsin.
21. Berilgan nomanfiy n va m butun sonlar uchun  $A(n,m)$  funksiya qiymati hisoblansin.

$$A(n, m) = \begin{cases} m + 1, & n = 0; \\ A(n - 1, 1), & n \neq 0, m = 0; \\ A(n - 1, A(n, m - 1)), & n > 0, m > 0. \end{cases}$$

22. 1 dan n gacha nomerlangan n ta aholi punkti mavjud. Ayrim punktlar o‘zaro yo‘llar bilan tutashgan. Bu yo‘llar yordamida 1-punkt dan n-

punktga borish mumkin yoki yoʻqligini aniqlansin. Yoʻllar haqidagi maʼlumot  $i$  va  $j$  ( $i < j$ ) sonlar juftliklari ketma-ketlik koʻrinishida berilgan. Ketma-ketlik 2 ta nol bilan tugaydi.

23. Butun  $n$  ( $n > 2$ ) soni va soat millari harakati yoʻnalishiga nisbatan teskari ketma-ketlikda joylashgan  $n$  ta  $p_1(x_1, y_1), \dots, p_n(x_n, y_n)$  uchlardan tashkil topgan qavariq koʻpburchak berilgan. Koʻpburchak yuzasi ikkita  $A$  va  $B$  sohalarga shunday ajratilsinki ular uchun  $|A - B| < \varepsilon$  oʻrinli boʻlsin. Bu yerda  $\varepsilon$  ( $\varepsilon > 0$ ) - berilgan aniqlik. Javob tariqasida har bir soha uchlari soni va



ularning koordinatalari chop qilinsin. Masalani yechishning quyi-dagi variantini taklif qilish mumkin. Koʻpburchakni umumiy uchi bitta, masalan  $p_1(x_1, y_1)$  nuqtada boʻlgan  $T_1, T_2, \dots, T_{n-2}$  uchburchaklarga boʻlib chiqiladi. Har bir uchburchak soha-ning yuzasi mos ravishda  $S_1, S_2, \dots, S_{n-2}$

boʻlsin (rasmga qarang).  $A = \sum_{i=1}^k S_i$  va  $B = \sum_{i=k+1}^{n-2} S_i$  hisoblagan

holda,  $|A - B| < \varepsilon$  sharni qanoatlantiruvchi  $k$  soni topiladi va mos koordinatalar chop qilinadi. Aks holda  $A > B$  shartni qanoatlantiruvchi eng kichik  $k$  soni topiladi va  $A = A - S_k$  amali bajarilib quyidagi qadamlarga oʻtiladi:

- 1)  $T_k$ - uchburchak umumiy uchga nisbatan qarshi tomonini teng ikkiga boʻlish orqali ikkita  $T_{k1}$  va  $T_{k2}$  uchburchakka ajratilsin;
- 2)  $A = A + S_{k1}$ ,  $B = B + S_{k2}$ ;
- 3) Agar  $|A - B| > \varepsilon$  va  $A > B$  boʻlsa  $T_k = T_{k1}$ ,  $A = A - S_k$  va 1- punktga oʻtilsin;
- 4) Agar  $|A - B| > \varepsilon$  va  $A < B$  boʻlsa  $T_k = T_{k2}$ ,  $B = B - S_k$  va 1- punktga oʻtilsin;
- 5)  $A$  va  $B$  sohalarning mos koordinatalari chop qilinsin.

24. Ishorasiz butun sonlar massivi  $a$  quyidagi usulda tartiblansin. Faraz qilyalik,  $k$ -berilgan sonning kompyuter xotirasidagi ikkilik koʻrinishidagi razryadlar soni boʻlsin (masalan, short int turi uchun  $k=16$ ). Massiv elementlari ularning  $k$ -razryadi boʻyicha ikkiga ajratiladi:  $k$ -razryadda 0 boʻlganlari  $a_0$  massivga, 1 boʻlganlari  $a_1$  massivga koʻchiriladi. Xuddi shu usulda  $a_0$  va  $a_1$  massivlarning har biri  $k-1$  razryad boʻyicha yana ikkita

massivga ajratiladi va hokazo. Jarayon  $k=0$  bo'lguncha davom etadi va undan keyin teskari tomonga ketiladi: har bir  $k$ - qadamdagi  $a_0$  va  $a_1$  massivlar ulanishi asosida  $a$  massiv yangidan shakllanadi va u funsiyaga  $k+1$  qadamdan amalga oshirilgan murojaatga natija sifatida qaytariladi.

## 18. Strukturalar

### *Namunaviy masala*

Tekislikda berilgan  $n$  ta  $p_i(x_i, y_i)$ ,  $i = \overline{1, n}$  nuqtalarni o'z ichiga oladigan minimal radiusli aylana aniqlansin.

### *Yechish usuli*

Barcha nuqtalarni o'z ichiga oladigan aylana markazi -  $M(x_m, y_m)$  nuqtalarning geometrik markazi bo'ladi, bu yerda

$$x_m = \frac{\sum_{i=1}^n x_i}{n}, y_m = \frac{\sum_{i=1}^n y_i}{n}.$$

Aylana radiusi sifatida aylana markazidan nuqtalargacha bo'lgan masofalarning eng kattasi olinadi -  $r = \max_{1 \leq i \leq n} \left\{ \sqrt{(x_m - x_i)^2 + (y_m - y_i)^2} \right\}$ .

### *Ilova matni*

```
struct Nuqta { float x,y; };
Nuqta Aylana_Markazi(Nuqta*nuqta,int n)
{
    Nuqta N;
    N.x=0; N.y=0;
    for(int i=0;i<n;i++)
    {
        N.x+=nuqta[i].x;
        N.y+=nuqta[i].y;
    }
    N.x/=n; N.y/=n;
    return N;
}
float Aylana_radiusi(Nuqta m,Nuqta*nuqta,int n)
{
    float r=0,d;
    for(int i=0;i<n;i++)
    {
        d=sqrt(pow(nuqta[i].x-m.x,2)+pow(nuqta[i].y-m.y,2));
    }
}
```

```

    if(d>r)r=d; }
return d;
}
int main()
{
    int n;
    Nuqta a_markaz;
    float a_radius;
    cout<<"Nuqtalar sonini kiriting: "; cin>>n;
    Nuqta *nuqta=new Nuqta[n];
    for(int i=0;i<n;i++)
    {
        cout<<"\nx["<<i+1<<"]="; cin>>nuqta[i].x;
        cout<<"y["<<i+1<<"]="; cin>>nuqta[i].y;
    }
    a_markaz=Aylana_Markazi(nuqta,n);
    a_radius=Aylana_radiusi(a_markaz,nuqta,n);
    cout<<"Aylana M("<<a_markaz.x<<','<<a_markaz.y<<") , ";
    cout<<"R="<<a_radius;
    return 0;
}

```

### ***Amaliy masalalar***

- enum Rasm={piki,trefi,bubni,cherwi};  
enum Nom={olti,etti,sakkiz,toqqiz,on,valet,dama,qirol,tuz};  
struct Karta{Rasm r;Nom n;};  
Rasm kz;  
Karta k1,k2;  
Zot rasmi kz berilganda k1 karta k2 kartani urishi yoki yo'qligini aniqlovchi mantiqiy Uradi(k1,k2,kz) funksiyasi tuzilsin.
- struct Domino\_tosh{unsigned char chap,ung;};  
Domino\_tosh domino[28];  
Berilgan domino qatordagi domino toshlari to'g'ri qo'yilganligini (navbatdagi domino toshining o'ng qismidagi son keyingi toshning chap qismidagi songa tengligini) tekshiradigan Tugri\_qator(domino) mantiqiy funksiya tuzilsin.
- struct Dekard{double x,y;};  
struct Qutb{double r,fi;}; //  $r \geq 0, -\pi < fi \leq \pi$   
Dekart d;  
Qutb q;  
Berilgan d nuqta koordinatasini dekart (dekart) koordinatalar siste-

- masidan qutb (Qutb) koordinatalar sistemasidagi nuqtaga o‘tkazadigan  $DQ(d,r)$  va teskari almashtirishni bajaruvchi  $QD(r,d)$  funksiyalar tuzilsin.
4. struct Shaxmat\_maydoni
 

```
{
  char vert;          // 'a'..'h'
  unsigned char goriz; // goriz: 1..8
};
```

 Shaxmat\_maydoni m1,m2;  
 Farzin bir yurishda m1 maydondan m2 maydonga o‘tishi mumkinligini tekshiradigan Farzin\_yurish(m1,m2) mantiqiy funksiya tuzilsin.
  5. struct Vaqt { unsigned char soat,min,sek;};  
 Vaqt t1,t2;  
 Quyidagi masalalar yechilsin:
    - a) t1 vaqt t2 vaqtdan oldin kelishini tekshiradigan oldin(t1,t2) mantiqiy funksiya tuzilsin (bir sutka ichida);
    - b) t vaqtdan 1 sekund ortiq vaqtni t1 parametriga beruvchi secund(t,t1) funksiya tuzilsin (sutka almashinuvi hisobga olingan holda);
    - d) t1 vaqtdan t2 vaqtgacha qancha vaqt o‘tganligini hisoblaydigan interval(d,t2,t1) funksiyasi tuzilsin ( $t2 > t1$ ).
  6. struct Rasional{int surat, maxraj;};  
 Rasional r[20],a,b;  
 Quyidagi masalalar yechilsin:
    - a) a va b ratsional sonlarning tengligini tekshiruvchi teng(a,b) mantiqiy funksiyasi tuzilsin;
    - b) a va b ratsional sonlar yig‘indisini a o‘zgaruvchisiga qaytaruvch summa(a,b) funksiyasi tuzilsin;
    - d) a ratsional sonni qisqartirib bo‘lmaydigan ko‘rinishga keltiradigan qisqartir(a) funksiyasi tuzilsin;
    - e) r ratsional sonlarning eng kattasini m parametriga beruvchi max(r,m) funksiyasi tuzilsin.
  7. struct Kompleks{double re,im;};  
 Kompleks turdagi z va  $\varepsilon > 0$  haqiqiy sonlari berilgan. Quyidagi funksiyalar  $\varepsilon$  aniqlikda hisoblansin:
    - a)  $y = e^z = 1 + z/1! + z^2/2! + \dots + z^n/n! + \dots$ ;
    - b)  $y = \operatorname{sh}z = z + z^3/3! + z^5/5! + \dots + z^{2n+1}/(2n+1)! + \dots$ ;
    - d)  $y = \operatorname{cos}z = 1 - z^2/2! + z^4/4! - \dots + (-1)^n z^{2n}/(2n)! + \dots$ ;
    - e)  $y = \ln(1+z) = z - z^2/2 + z^3/3 - \dots + (-1)^{n-1} z^n/n + \dots, (|z| < 1)$ ;
    - g)  $y = \operatorname{arctg}z = z - z^3/3 + z^5/5 - \dots + (-1)^n z^{2n+1}/(2n+1) + \dots, (|z| < 1)$ .
  8. struct Kompleks{double re,im;};

```
struct Koeff{Kompleks a,b,c;}; // a≠0
```

```
Koeff Kvt;
```

```
Kompleks x1,x2;
```

Koeffitsiyentlari Kompleks turidagi  $ax^2+bx+c=0$  kvadrat tenglama ildizlari topilsin.

9. struct Sana{unsigned char kun,oy;unsigned int yil;};

```
enum Hafta_ kuni{dush,shesh,chor,pay,juma,shan,yak};
```

```
Sana d;
```

Quyidagi masalalar yechilsin:

a) d-sanaga tegishli bo'lgan oydagi kunlar sonini hisoblovchi Oy\_kunlari(d) funksiyasi tuzilsin (kabisa bo'lmagan yil uchun);

b) d-sana to'g'riligini (masalan, 31 iyun bo'lmasligi va h.k.) tekshiradigan mantiqiy Tugri\_sana(d) funksiyasi tuzilsin.

d) 1-yilning 1-yanvaridan to d-sanagacha necha kun o'tganligini hisoblaydigan Kunlarsoni(d) funksiyasi tuzilsin.

e) d - sana haftani qaysi kunini aniqlaydigan Hafta\_kuni(d) funksiyasi tuzilsin (eramizning 1-yilining 1-yanvari dushanba (dush) kuni bo'lganligini hisobga olgan holda).

10. struct Manzil{char Kucha[20]; int uy,xonadon;};

```
struct Yashovchi{char familiya[20], shahar[20]; Manzil manzil;};
```

```
Yashovchi *shaxs;
```

Berilgan shaxs ro'yxatidagi turli shaharda, lekin bir xil manzilda (Manzil) yashovchi ixtiyoriy ikkita shahar yashovchisining familiyasini chop qiladigan Taqdir\_xazili() funksiyasi tuzilsin.

11. O'qish faylida talabalarning sessiya natijalari haqidagi ma'lumot quyidagi ko'rinishga ega:

```
<familiya>,<guruh nomeri>,<reyting1>,<reyting2>,<reyting3>.
```

Bu yerda familiya 15 harfgacha bo'lgan satr, guruh nomeri - butun son, har bir reyting 0 va 100 oraligidagi haqiqiy son: reyting1 - matematik analizdan, reyting2 - algebradan, reyting3 - programmalash predmetlaridan. Quyidagilar aniqlansin:

a) kamida bitta fandan qarzdor bo'lgan talabaning familiyasi;

b) barcha imtihonlarni 4 va 5 bahoga topshirgan talabalar necha foizni tashkil qilishi;

d) talabalar qaysi fandan imtihonlarni eng yaxshi natija bilan topshirganligi;

e) talabalari o'zlashtirishining o'rtacha qiymati bo'yicha o'smaydigan tartibdagi guruhlarining tartib nomerlari.

12. O'qish faylida oliy o'quv yurti talabalari to'g'risidagi quyidagi ma'lumot yozilgan:



<familiya>,<ismi>,<otasining ismi>,<jinsi>,<yoshi>,<kursi>.

Bu yerda familiya, ismi va otasining ismi – har biri uzunligi 12 harfdan ko‘p bo‘lmagan satrlar, jinsi - ‘e’ yoki ‘a’ harflar bilan (erkak, ayol) ko‘rsatilgan, yoshi - 16 dan 35 gacha bo‘lgan butun son, kursi - 1 dan 4 gacha bo‘lgan butun son. Quyidagi masalalar yechilsin:

a) erkaklar soni eng ko‘p bo‘lgan kurs nomeri;

b) eng ko‘p tarqalgan erkak va ayollar ismlari;

d) yoshi va ismi bir xil bo‘lgan talaba qizlar familiyalarining alfavit tartibidagi ro‘yxati.

13. AEROFLOT strukturasi <reysning manzil punkti>,<reys nomeri> va <samolet turi> maydonlarini o‘z ichiga oladi. Berilgan n o‘lchamidagi AEROFLOT turidagi massiv yaratilib, ma’lumotlar kiritilsin va quyidagi amallar bajarilsin:

a) massiv, reys nomerining o‘sichi bo‘yicha tartiblansin;

b) klaviaturadan kiritilgan manzilga uchadigan reys nomeri va samolet turi chop etilsin;

d) klaviaturadan kiritilgan samolet turi uchadigan reys nomerlari va manzillari chop etilsin.

14. ISHCHI strukturasi <familiya va initsiallari>,<lavozimi> va <ishga kirgan yili> maydonlarini o‘z ichiga oladi. Berilgan n o‘lchamidagi ISHCHI turidagi massiv yaratilsin va quyidagi amallar bajarilsin:

a) massiv ishchi familiyasining alfavit bo‘yicha joylashuviga mos tartiblansin;

b) klaviaturadan kiritilgan yildan keyin ishga kirgan ishchilar familiyasi chop etilsin;

d) klaviaturadan kiritilgan lavozimdagi ishchilar familiyasi va ishga kirgan yili chop etilsin.

15. POEZZD strukturasi <poezd nomeri>,<boradigan manzil nomi> va <jo‘nash vaqti> maydonlarini o‘z ichiga oladi. Berilgan n o‘lchamidagi POEZZD turidagi massiv yaratilsin va quyidagi amallar bajarilsin:

a) massiv poyezd nomerlarining o‘sishi bo‘yicha tartiblansin;

b) klaviaturadan kiritilgan nomerga mos poezd nomeri haqidagi ma’lumot chop etilsin;

d) klaviaturadan kiritilgan manzilga va ko‘rsatilgan vaqtdan keyin jo‘naydigan poezdlar haqidagi ma’lumot chop etilsin.

16. MARSHRUT strukturasi <marshrut boshi>,<marshrut oxiri> va <marshrut nomeri> maydonlarini o‘z ichiga oladi. Berilgan n o‘lchamidagi MARSHRUT turidagi massiv yaratilsin va quyidagi amallar bajarilsin:

a) massiv marshrut nomerlarining o‘sishi bo‘yicha tartiblansin;

b) klaviaturadan nomeri kiritilgan marshrut haqidagi ma’lumot chop

etilsin;

d) klaviaturadan kiritilgan nom bo'yicha, boshlanashi yoki oxiri shu nomdagi punkt bo'lgan marshrutlar haqidagi ma'lumot chop etilsin.

17. BLOKNOT strukturasi shaxs haqidagi - <familiya va ismi>, <telefon nomeri> va <tug'ilgan sanasi> maydonlarini o'z ichiga oladi. Berilgan n o'lchamidagi BLOKNOT turidagi massiv yaratilsin va quyidagi amallar bajarilsin:

a) massiv odamning tug'ilgan sanasining o'sishi bo'yicha tartiblansin;

b) klaviaturadan telefon nomeri kiritilgan odam haqidagi ma'lumot chop etilsin;

d) tug'ilgan oyi klaviaturadan kiritilgan songa mos keluvchi odamlar haqidagi ma'lumot chop etilsin.

18. NARX strukturasi mahsulot haqidagi <mahsulot nomi>, <mahsulot sotiladigan magazin nomi> va <mahsulotning so'mdagi narxi> maydonlarini o'z ichiga oladi. Berilgan n o'lchamidagi NARX turidagi massiv yaratilsin va quyidagi amallar bajarilsin:

a) massiv mahsulot nomini alfavit bo'yicha joylashuviga mos tartiblansin;

b) nomi klaviaturadan kiritilgan mahsulot haqidagi ma'lumot chop etilsin;

d) klaviaturadan kiritilgan nomdagi magazinda sotiladigan mahsulotlar ro'xati va ularning jami narxi chop etilsin.

19. ORDER strukturasi bank mijozi haqidagi <to'lovchining hisob raqami>, <oluvchining hisob raqami> va <o'tkaziladigan pul miqdori> maydonlarini o'z ichiga oladi. Berilgan n o'lchamidagi ORDER turidagi massiv yaratilsin va quyidagi amallar bajarilsin:

a) massiv to'lovchining hisob raqami bo'yicha tartiblansin;

b) hisob raqami klaviaturadan kiritilgan to'lovchining hisob raqamidan qancha pul olinganligi haqidagi ma'lumotlar chop etilsin;

d) hisob raqami klaviaturadan kiritilgan oluvchining hisob raqamiga qaysi hisobdan qancha pul kelib tushganligi haqidagi ma'lumot chop etilsin.

20. KUTUBXONA strukturasi <kitob registratsiya nomeri>, <muallif>, <kitob nomi>, <nashr yili> va <nashriyot> maydonlaridan tashkil topgan. Berilgan butun n uchun elementlari KUTUBXONA strukturalaridan tashkil topgan massiv hosil qilinsin va quyidagi amallar bajarilsin:

a) mualliflar familiyalari alfavit bo'yicha tartiblangan ko'rinishdagi kitoblar ro'yxati chop etilsin;

b) ko'rsatilgan nashriyot tomonidan chiqarilgan kitoblar ro'yxati chop etilsin;

d) registratsiya nomeri ko'rsatilgan kitob massivdan o'chirilsin.

## 19. Matn fayllari

### *Namunaviy masala*

Butun sonlar kamaymaydigan tartibda kiritilgan  $f_1$  va  $f_2$  matn fayllari berilgan. Ushbu fayllar elementlaridan, sonlarning tartiblanishi saqlangan holda,  $f_3$  matn fayli hosil qilinsin. Fayllarning diskdagi nomlari buyruq satri orqali kiritilsin.

### *Yechish usuli*

Fayllar bilan ishlash uchun <iostream.h> kutubxonasida aniqlangan funksiyalardan foydalanamiz:

fopen() – fayl o'zgaruvchisini diskdagi fizik fayl bilan bog'lash uchun xizmat qiladi. Agar funksiya NULL(0) qiymatini qaytarsa, bog'lanish muvaffaqiyatsiz bo'lganligini bildiradi;

fscanf() funksiyasidan fayldagi sonni formatli o'qish va fayl tugaganligini aniqlash uchun foydalaniladi. Funksiyaning -1(EOF) qiymatni qaytarishi, fayl tugaganligini anglatadi.

Masalani yechish algoritmi quyidagicha:

1.  $f_1$  va  $f_2$  fayldan mos ravishda  $a, b$  sonlari o'qiladi;
2. Toki  $a \leq b$  va  $f_1$  tugamaganlik sharti bajarilar ekan  $a$  qiymati  $f_3$  fayliga yozilsin va  $f_1$  fayldan qiymat  $a$  o'zgaruvchiga o'qilsin;
3. Toki  $a > b$  va  $f_2$  tugamaganlik sharti bajarilar ekan  $b$  qiymati  $f_3$  fayliga yozilsin va  $f_2$  fayldan qiymat  $b$  o'zgaruvchiga o'qilsin;
4. O'qilayotgan fayllardan kamida bittasi tugamagan bo'lsa 2-qadamga o'tilsin.
5. Tamom.

Fayllar nomlari buyruq satrida ilova nomidan ("exe" fayldan) keyin probel ajratuvchisi bilan yoziladi va ular main() funksiyasining argv – satrlar massiviga joylashadi, argc - buyruq satrdagi satrostilari soni.

Masalan, "d:\f1\_f2\_f3.exe file1.txt file2.txt file3.txt" buyruq satrida 4 ta satrostilari mavjud.

C++Builder muhitidagi Run → Parametrs opsiyasidagi Parametrs maydoniga "file1.txt file2.txt file3.txt" satrini yozish orqali buyruq parametrlarini berish va ilovani ishga tushirish mumkin.

### *Ilova manti*

```
const int n=80;  
typedef FILE * Fmatn;  
typedef char Satr[n];
```

```

int Ulash(Satr f1_nomi,Satr f2_nomi,Satr f3_nomi)
{
int a,b;
bool tamom_f1=false,tamom_f2=false;
Fmatn f1,f2,f3;
if((f1=fopen(f1_nomi,"rt"))==NULL)
{
cout<<" "<<f1_nomi<<" fayli ochilmadi!";
return 1;
}
if((f2=fopen(f2_nomi,"rt"))==NULL)
{
cout<<" "<<f2_nomi<<" fayli ochilmadi!";
return 1;
}
if((f3=fopen(f3_nomi,"wt"))==NULL)
{
cout<<" "<<f3_nomi<<" fayli hosil qilinmadi!";
return 1;
}
if(fscanf(f1,"%i",&a)==EOF) tamom_f1=true;
if(fscanf(f2,"%i",&b)==EOF) tamom_f2=true;
do
{
while(!tamom_f1&&(tamom_f2||a<=b))
{
fprintf(f3,"%i ",a);
if(fscanf(f1,"%i",&a)==EOF) tamom_f1=true;
}
while(!tamom_f2&&(tamom_f1||b<a))
{ fprintf(f3,"%i ",b);
if(fscanf(f2,"%i",&b)==EOF) tamom_f2=true;
}
}
while (!tamom_f1||!tamom_f2);
fclose(f1); fclose(f2); fclose(f3);
return 0; }
int main(int argc,char* argv[])
{
Satr f1_nomi,f2_nomi,f3_nomi;

```

```

if(argc!=4) { cout<<"Buyruq satri noto'g'ri kiritilgan!"; return 0;}
strcpy(f1_nomi,argv[1]); strcpy(f2_nomi,argv[2]); strcpy(f3_nomi,argv[3]);
if(!Ulash(f1_nomi,f2_nomi,f3_nomi)) cout<<"\nFayl hosil qilindi: f1+f2=>f3";
else cout<<"\nMasalani yechish imkoniyati yo'q.";
return 0;
}

```

### *Amaliy topshiriqlar*

1. Berilgan s matn faylning manfiy elementlari yig'indisini hisoblaydigan funksiya tuzilsin.
2. Bo'sh bo'lmagan, butun sonlardan tashkil topgan r matn fayldagi sonlar o'sish yoki kamayish tartibida joylashganligini tekshiradigan mantiqiy funksiya tuzilsin.
3. Berilgan t<sub>1</sub> va t<sub>2</sub> matn fayllar biri ikkinchisining nusxasi yoki yo'qligini tekshiradigan mantiqiy funksiya tuzilsin.
4. Kamida ikkita satrdan iborat bo'lgan f matn faylining oxiridan bitta oldingi satrini natija sifatida qaytaradigan funksiya tuzilsin.
5. Berilgan s satrda uchragan raqamlarni t matnga o'tkazadigan funksiya tuzilsin.
6. Berilgan musbat butun n sonidan oshmaydigan Fibonachchi sonlarini f matn fayliga yozadigan funksiya tuzilsin.
7. Butun sonlar yozilgan, bo'sh bo'lmagan f matn fayl berilgan. Fayldagi elementlarning o'rta arifmetigidan kichik bo'lgan fayl elementlar miqdorini aniqlaydigan funksiya tuzilsin.
8. Berilgan f matn fayli bo'sh bo'lmagan n satrlardan iborat. Satrdagi so'zlar bir-biridan vergul bilan ajratilgan, oxirgi so'z nuqta bilan tugagan. Har bir satrning oxirgi so'zi "kesib" olingan so'zlardan tashkil topgan satr g matn faylga yozilsin.
9. Berilgan t matn fayl uchun quyidagi funksiyalar tuzilsin:
  - a) add1(t,s), t-matn boshiga s satrni qo'shadigan;
  - b) addlast(t,s), t-matn oxiriga s satrini qo'shadigan;
  - d) double(t), t-matndagi har bir raqamni ikkilantiradigan;
  - e) replace(t,s) - bo'sh bo'lmagan t matnning oxirgi satrini s satri bilan almashtiradigan;
  - f) next(t) - t matnda uchragan har bir raqamni, shu raqamdan keyin keluvchi raqam bilan almashtiradigan ('9' raqami '0' bilan almashtiriladi);
  - j) del(t)- t matndagi oxiridan bitta oldingi satrini (agar u mavjud bo'lsa) o'chiradigan;
  - i) first(t) - t matndagi har bir satrlarning faqat birinchi uchraganini qoldiradigan.

10. Haqiqiy sonlar yozilgan f matn faylidagi eng uzun o‘svuchi ketma-ketlik elementlari miqdorini aniqlovchi funksiya tuzilsin.
11. Berilgan f va g matn fayllarida sonlar kamaymaydigan ko‘rinishda tartiblangan bo‘lsin. Bu fayllarni yagona kamaymaydigan ko‘rinishda h faylga birlashtirish talab qilinadi.
12. Mantiqiy relation(f) funksiyasi tuzilsin. Bu funksiya f fayldagi berilganlar “munosabat” sintaksisidagi to‘g‘ri yozuv ekanligi aniqlasin (qoidalar quyida keltirilgan).
- <munosabat>::=<son><munosabat belgisi><son>  
 <munosabat belgisi>::=<| = |> | <= | <> | >=  
 <son>::=<raqam> | <raqamlar>  
 <raqamlar>::=<nol emas><raqam> | <raqamlar><raqam>  
 <raqam>::=0 | <nol emas>  
 <nol emas>::=1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9
13. To‘qqizta satrlardan iborat t matn faylini hosil qiluvchi funksiya tuzilsin. Bunda 1-satrdan bitta ‘1’ belgisi, 2-satrdan ikkita ‘2’ belgisi va shu tariqa 9-satrdan to‘qqizta ‘9’ belgisi bo‘lsin.
14. Klaviaturadan belgilarni bittalab, birinchi nuqttagacha o‘qiydigan va ularni t faylga 40 ta belgidan iborat satr ko‘rinishida yozadigan funksiya tuzilsin (satrga nuqta kirmaydi va oxirgi satrdagi belgilar 40 dan kam bo‘lishi mumkin).
15. Quyidagilarni amalga oshiradigan funksiya tuzilsin:  
 a) t matn fayldagi bo‘sh satrlar sonini hisoblaydigan;  
 b) t matn faylidagi satrlar ichida eng uzun satr uzunligini hisoblaydigan.
16. Berilgan t matn fayli bo‘sh bo‘lmagan satrlardan iborat bo‘lsin. Quyidagi shartlarni bajaruvchi satrlar sonini hisoblovchi funksiya tuzilsin:  
 a) ‘d’ harfidan boshlanadigan;  
 b) ‘z’ harfi bilan tugaydigan;  
 d) bir xil belgi bilan boshlanadigan va tugaydigan;  
 e) bir xil belgilardan tashkil topgan.
17. Berilgan t<sub>1</sub> matn faylidan, satrlarga bo‘linishini saqlagan holda, t<sub>2</sub> faylga o‘tkazadigan funksiya tuzilsin.
18. Berilgan t matn fayli har birining uzunligi 80 belgidan oshmaydigan satrlarga bo‘lingan deb hisoblab funksiya tuzilsin. Funksiya t fayldagi har bir satrni 80 belgigacha o‘ng tomondan probel (‘␣’) bilan to‘ldirib, barcha satrlari 80 belgidan iborat t<sub>2</sub> faylga o‘tkazadi.
19. char suz[ ][20];  
 Belgilar soni 20 tagacha bo‘lgan sozlar ro‘yxati berilgan (suz). Har bir so‘zni t matn fayliga alohida satr ko‘rinishida o‘tkazilsin.

20. Berilgan  $t$  matn faylida bo‘sh bo‘lmagan va probel bilan ajratilgan haqiqiy sonlar ketma-ketligi yozilgan. Bu sonlar ichida eng kattasini topadigan funksiya tuzilsin.
21. Berilgan  $t_1$  matn faylida bo‘sh bo‘lmagan va probel bilan ajratilgan butun sonlar ketma-ketligi yozilgan. Bu sonlar ichida musbatlarini  $t_2$  faylga o‘tkazadigan funksiya tuzilsin.
22. Bo‘sh bo‘lmagan  $t$  matn faylidagi har satrni, uning boshiga satrning tartib nomerini qo‘yib chop qiladigan funksiya tuzilsin. Satr tartib nomeri 4 o‘rin egallaydi va undan keyin probel qo‘yiladi.
23. Berilgan Kitob matn faylidagi satrlarning boshlang‘ich bo‘linishlarini inkor etgan holda shunday satrlarga bo‘linsinki, natijada satr nuqta bilan tugasin yoki 60 belgidan iborat bo‘lsin, agar bu belgilar orasida nuqta uchramasa.
24. Berilgan matn faylidagi eng qisqa satrlarning birinchisi chop qilinsin.
25. Name1 va Name2 nomli matn fayllari berilgan. Ulardagi satrlarni ketma-ket birlashuvidan yangi Name3 fayli hosil qilinsin. Fayllarni birlashish tartibi (ketma-ketligi) foydalanuvchi tomonidan beriladi.
26. Matn fayli va  $k$  butun soni berilgan. Matn fayldagi  $k$ -satr o‘chirilsin. Agar faylda bu satr bo‘lmasa, u o‘zgarmasdan qoldirilsin.
27. Matn fayli va  $k$  butun soni berilgan. Matn fayldagi  $k$ -satr oldiga (keyinga) bo‘sh satr qo‘yilsin. Agar faylda bu satr bo‘lmasa u o‘zgarmasdan qoldirilsin.
28. Matn fayli va  $s$  satri berilgan. Fayldagi barcha bo‘sh satrlar  $s$  satr bilan almashtirilsin.
29. Matn fayli berilgan. Undagi ketma-ket keluvchi probellar bitta probel bilan almashtirilsin.
30. Ichidagi satrlari 60 ta belgidan oshmagan va chap tomonga tekislangan matn fayli berilgan. Har bir bo‘sh bo‘lmagan satrlar oldiga etarli sondagi probelni qo‘yish orqali satrlar o‘ng tomonga (markazga) tekislansin.
31. Berilgan matn faylidagi abzats boshlanishini 5 ta probeldan boshlanuvchi satr aniqlaydi. Matn fayli abzats oldidagi probellar olib tashlash va satr oldiga bo‘sh satrni qo‘yish orqali yangi ko‘rinishga keltirilsin.
32. Berilgan matn faylidagi satrlarni teskari tartibda joylashtirish orqali yangi matn fayli hosil qilinsin.
33. Matn fayli va butun  $k$  soni berilgan. Matn faylidagi  $k$ - xatboshi (abzats) o‘chirilsin. Xatboshi 5 ta probel bilan boshlanadigan satr bilan aniqlanadi. Agar bu tartib nomerli xatboshi bo‘lmasa fayl o‘zgarishsiz qoldirilsin.
34. Har biri chap va o‘ng tomondan probellar bilan to‘ldirilgan haqiqiy sonni ifodalovchi satrlardan tashkil topgan matn fayli berilgan. Bu son-

larning yig'indisi va miqdori aniqlansin. Masalani yechishda faqat fayldan belgili o'qish amalga oshirilsin.

35. Name1 va Name2 nomli matn fayllari berilgan. Ulardagi mos tartib nomerdagi satrlarni ketma-ket yozishdan yangi Name3 fayli hosil qilinsin (Name1 faylining 1-satri, Name2 faylining 1-satri, Name1 faylining 2-satri, Name2 faylining 2-satri va hokazo). Agar birorta fayl ikkinchisidan qisqa bo'lsa, u tugashi bilan ikkinchi faylning qolgan satrlari Name3 fayliga ko'chiriladi.
36. Ikkita matn fayli berilgan. Ularning faqat bittasiga kiruvchi satrlar chop etilsin.
37. Name1 nomli bo'sh bo'lmagan matn fayli va  $k$  natural soni berilgan. Ikkita matn fayllari hosil qilinsin: Name2 faylga Name1 har bir satrning birinchi (oxirgi)  $k$  ta belgisini o'zida saqlovchi, (agarda satr uzunligi  $k$  sonidan kichik bo'lsa satrning o'zini saqlasin), Name3 fayli esa har bir satrning  $k$  - belgisidan tashkil topsin (agarda satr uzunligi  $k$  dan kichik bo'lsa Name3 fayliga probel yozilsin).
38. Ikkita  $f_1$  va  $f_2$  matn fayllarini satrma-satr solishtiradigan funksiya tuzilsin. Funksiya  $f_2$  fayldagi  $f_1$  faylga nisbatan o'chirilgan yoki qo'shilgan satrlarni chop qilsin.
39. C++ tilida yozilgan ko'p qatorli programma matnidagi har bir identifikatorlar, kalit so'zlar va ular uchragan satrlar nomerlarini eslab qolish asosida lug'at yaratilsin.
40. Tekislikda to'g'ri chiziq  $ax+by+c=0$  tenglama bilan beriladi. Bunda  $a, b$  koeffisientlari bir vaqtda 0 emas va  $a, b, c$  - butun sonlar. Agar  $f$  faylda to'g'ri chiziqlar koeffisientlari saqlangan bo'lsa (uchdan kam bo'lmagan),  $f$  fayldan  $g$  faylga quidagi shartlarni bajaruvchi to'g'ri chiziqlar koiffisientlarini yozing:
  - a) birinchi to'g'ri chiziqqa parallel;
  - b) birinchi to'g'ri chiziqqa parallel va bir-biridan farqli chiziqlar;
  - d) berilgan to'g'ri chiziq bilan kesishadigan chiziqlar.
41. Butun sonlardan iborat mant fayl berilgan, shu fayl elementlarining teskari tartibda joylashuvidan hosil bo'lgan yangi fayl hosil qilinsin.
42. Elementlari haqiqiy sonlardan iborat kvadrat matritsani satrlar bo'yicha o'zida saqlovchi matn fayl berilgan. Matritsaning  $i$ -satr va  $j$ -ustundagi elementlari chop qilinsin. Agar matritsaning  $i$ -satri yoki  $j$ -ustuni bo'lmasa, bu holda xabar berilsin.
43. Lotin harflaridan tashkil topgan satr berilgan. Undagi harflarning necha marta uchraganligini ifodalovchi gistogramma matn faylda (son orqali) hosil qilinsin.



## 20. Binar fayllar

### *Namunaviy masala*

Talaba haqidagi ma'lumot uning familiyasi va inisiallari, o'quv bosqichi, fanlar boyicha reyting bali bilan berilgan. Talabalar haqidagi ma'lumot binar faylga saqlansin. Fayldan ko'rsatilgan baho olgan talabalar ro'yxati chop etilsin.

### *Yechish usuli*

Har bir talaba haqidagi ma'lumot matn faylidan o'qilib <familiyasi va ismu sharifi>, <o'quv bosqichi > va <fan boyicha reyting bali> maydonlaridan tashkil topgan struktura turidagi o'zgaruvchiga o'zlashtiriladi va binar faylga yoziladi. Berilganlar tugagandan keyin binar fayl yopiladi. Klaviaturadan 2 va 5 oraliq'dagi baho soni kiritiladi va binar fayl o'qish rejimida qayta ochiladi va talaba haqidagi berilganlar strukturaga o'qiladi. Agar strukturaning <fan boyicha reyting bali> maydoni qiymati baho qiymatiga mos oraliqqa tegishli bo'lsa, struktura maydonlarining qiymatlari ekranga chop qilinadi. Ushbu jarayon binar fayl tugaguncha davom etadi.

O'qishni yaxlit satr sifatida qabul qilish uchun talaba familiyasi va ismu sharifi '.' belgisi bilan ajratilib kiritiladi.

### *Ilova matni*

```
const int N=80;
typedef char Fayl_nomi[N];
struct Talaba
{ char FISh[30]; unsigned char kurs; float ball; };
int Talaba_Matn_Binar(FILE *t, FILE *b)
{
    int n=0;
    Talaba talaba;
    do
    {
        if(fscanf(t,"%s %i %f", &talaba.FISh, &talaba.kurs, &talaba.ball)==EOF) break;
        else
        { n++;
          if(fwrite(&talaba,sizeof(Talaba),1,b)!=1)
          {
              printf("Binar faylga yozishda xatolik bo'ldi!");
              return -1;
          }
        }
    }
}
```

```

return n;
}
int Bahoga_mos_Talabalar(FILE * bfile, int baho)
{
int n=0;
Talaba talaba;
bool talaba_bor, sarlavha=true;
while(1)
{
if(!fread(&talaba,sizeof(Talaba),1,bfile)) break;
talaba_bor=false;
switch(baho)
{
case 5: talaba_bor=talaba.ball>85; break;
case 4: talaba_bor=(talaba.ball>71 && talaba.ball<86); break;
case 3: talaba_bor=(talaba.ball>55 && talaba.ball<=71); break;
default: talaba_bor=(talaba.ball<56);
}
if(talaba_bor)
{
n++;
if(sarlavha)
{
printf(" Fandan bahosi %i bo'lgan talabalar ro'yxati:\n",baho); sarlavha=false; }
printf("FISH:%s Bosqich:%i Reyting Ball:%6.2f\n",talaba.FISH,talaba.kurs, tala-
ba.ball);
}
}
return n;
}
int main(int argc, char* argv[])
{
int n;
unsigned char baho;
FILE * f_matn,*f_binar;
Fayl_nomi mf_nomi,bf_nomi;
printf("Matn fayl nomini kiriting:");
scanf("%s",&mf_nomi);
printf("Binar fayl nomini kiriting:");
scanf("%s",&bf_nomi);

```

```

if((f_matn=fopen(mf_nomi,"rt"))==NULL)
{
printf("%s fayli ochilmadi!",mf_nomi); return -1;
}
if((f_binar=fopen(bf_nomi,"wb"))==NULL)
{
printf("%s faylni yaratishda xatolik ro'y berdi!",bf_nomi); return -1;
}
do
{
printf("Bahoni kiritng(2..5):"); scanf("%i",&baho);
}
while(baho<2 || baho>5);
n=Talaba_Matn_Binar(f_matn, f_binar);
if(n)
{
fclose(f_binar); f_binar=fopen(bf_nomi,"rb");
n=Bahoga_mos_Talabalar(f_binar,baho);
if(n)printf("Jami talabalar soni:%i",n);
else printf("%i baho olgan talabalar yo'q",baho);
}
fclose(f_matn);
fclose(f_binar);
return 0;
}

```

### ***Amaliy topshiriqlar***

1. SEMESTR faylida 1-bosqich o'quv semestrining natijalari haqida ma'lumot mavjud. Har bir talaba to'g'risidagi ma'lumot quyidagi ko'rinishda berilgan:

<familiya>,<guruh tartib nomeri>,<baho1>,<baho2>,<baho3>.

Bu yerda familiya - 12 harfgacha, guruh tartib nomeri - 101 dan 116 gacha butun son, har bir baho – 2,3,4 yoki 5, baho1 – matematik analizdan, baho2 – algebradan, baho3 – programmalashdan. Quyidagi natijalarni chop qiluvchi funksiyalar tuzilsin:

- a) kamida bitta fandan qazdor bo'lgan talabaning familiyasi;
- b) barcha imtihonlarni 4 va 5 ga topshirgan talabalar necha foizni tashkil qilishini;
- d) talabalar qaysi fandan imtihonlarni eng yaxshi natija bilan topshirganlar;

e) talabalar o'zlashtirishining o'rtacha qiymati bo'yicha o'smaydigan tartibdagi guruhlarining tartib nomerlari.

2. ANKETA faylida oliy o'quv yurti talabalarining har biri to'g'risida quyidagi ma'lumot yozilgan:

<familiya>,<ismi>,<sharifi>,<jinsi>,<yoshi>,<o'quv bosqichi>,

bu yerda talabaning familiyasi, ismi va otasining ismi -satrlar, jinsi 'E' va 'A' harflar bilan ko'rsatilgan, yoshi - 16 dan 35 gacha bo'lgan butun son, o'quv bosqich - 1 dan 4 gacha bo'lgan butun son. Quyidagi shartlar uchun natija beruvchi funksiyalar tuzilsin:

a) erkaklar soni eng ko'p bo'lgan o'quv bosqichi;

b) eng ko'p tarqalgan erkak va ayollar ismlari;

d) yoshi va shariflari bir vaqtda eng ko'p tarqalgan talaba qizlar familiyalarining alfavit tartibidagi ro'yxati.

3. AEROFLOT strukturasi, <reysning manzil punkti>,<reys nomeri>,<samolet turi> va <bo'sh o'rinlar soni> maydonlarini o'z ichiga oladi. Ushbu strukturadagi berilganlar f binar faylga yozilgan. f binar fayl ustida quyidagi amallar bajarilsin:

a) klaviaturadan kiritilgan manzilga uchadigan reys nomeri va samolet turi chop etilsin;

b) klaviaturadan kiritilgan samolet turi uchadigan reys nomerlari va manzillar chop etilsin.

d) ko'rsatilgan manzil uchun bo'sh o'rinlar soni ko'rsatilsin.

4. POEzd strukturasi, <poezd nomeri>,<manzil nomi>,<jo'nash vaqti> va <bo'sh o'rinlar> maydonlarini o'z ichiga oladi. Ushbu strukturadagi berilganlar f binar faylga yozilgan. f binar fayl ustida quyidagi amallar bajarilsin:

a) klaviaturadan kiritilgan poezd nomeriga mos poezd haqidagi ma'lumot chop etilsin;

b) klaviaturadan kiritilgan manzilga va ko'rsatilgan vaqtdan keyin jo'naydigan poezd haqidagi ma'lumot chop etilsin.

d) ko'rsatilgan poezd nomerida bo'sh o'rinlar bo'lsa uning qiymati berilgan k soniga kamaytirilsin.

5. BLOKNOT strukturasi tanish odamning <familiya va ismi>,<telefon raqami> va <tug'ilgan sanasi> maydonlarini o'z ichiga oladi. Ushbu strukturadagi berilganlar f binar faylga yozilgan. f binar fayl ustida quyidagi amallar bajarilsin:

a) klaviaturadan telefon raqami kiritilgan odam haqidagi ma'lumot chop etilsin;

b) tug'ilgan oyi klaviaturadan kiritilgan songa mos keluvchi odamlar haqidagi ma'lumot chop etilsin;

- d) klaviaturadan kiritilgan familiya va ism bo'yicha f fayldan yozuv o'chirilsin.
6. MUCHAL strukturasi shaxsning <famiyiasi va ismi>,<muchal nomi> va <tug'ilgan sanasi> maydonlarini o'z ichiga oladi. Ushbu strukturadagi berilganlar f binar faylga yozilgan. f binar fayl ustida quyidagi amallar bajarilsin:
- a) massiv shaxsning tug'ilgan sanasini o'sishi bo'yicha zanjir ko'rinishidagi dinamik struktura yordamida tartiblansin va g binar faylga yozilsin;
- b) familiyasi klaviaturadan kiritilgan familiya bilan ustma-ust tushadigan shaxslar haqidagi ma'lumot chop etilsin;
- d) klaviaturadan kiritilgan muchal yili tug'lgan shaxslar haqidagi ma'lumot chop etilsin.
7. BANK strukturasi bank mijozlari haqidagi <mijoz familiya,ismi va shari-fi>,<hisob raqami>,<mablag' miqdori>,<foyiz>,<mablag qo'yilgan yil>,<mijoz manzili> maydonlarni o'z ichiga oladi. Ushbu strukturadagi berilganlar f binar faylga yozilgan. f binar fayl ustida quyidagi amallar bajarilsin:
- a) dinamik zanjir strukturasiidan foydalangan holda mijoz familiyasining alfavit bo'yicha tartiblangan ro'yxati chop etilsin;
- b) klaviaturadan hisob raqami ko'rsatilgan mijoz hisobiga, uning foyiziga mos ravishda qo'shiladigan pul miqdori hisoblansin;
- d) klaviaturadan kiritilgan yilda bankka mablag' qo'ygan mijoz haqidagi ma'lumot chop etilsin.
8. KIYIM fayli <mahsulot shifri>,<o'lcham>,<bo'yi>,<reja bo'yicha mahsulot soni>, <amalda tayyor mahsulotlar soni> maydonlaridan iborat. MAHSULOT fayli ushbu mahsulotlar haqidagi <mahsulot shifri>,<mahsulot nomi> maydonlaridan iborat strukturadagi berilganlardan tashkil topgan. Ekranda <mahsulot shifri>,<mahsulot nomi>,<o'lcham>,<bo'yi>,<reja bo'yicha mahsulot soni>,<amalda tayyor mahsulotlar soni>ustunlaridan iborat jadval chop etilsin.
9. TALABA faylida <fakultet shifri>,<o'quv kursi>,<guruh tartib raqami>,<talaba familiyasi va sharifi>,<predmet shifri>,<predmet bo'yicha baho> maydonlaridan iborat. FAKULTET fayli fakultetlar haqidagi <fakultet shifri>,<fakultet nomi> maydonlardan tashkil topgan. PREDMET faylida esa <fakultet shifri>,<o'quv kursi>,<predmet shifri>,<predmet nomi> berilganlari joylashgan. Ekranga har bir o'quv predmeti bo'yicha o'zlashtirish jadvali chop etilsin. Unda <fakultet nomi>,<o'quv kursi>, <guruh tartib raqami>,<talaba familiyasi va ismu sharifi>,<predmet bo'yicha baho> ustunlari bo'lsin.
10. SHAXS faylida shaxs haqidagi <familiya va sharifi>, <tugilgan yili>,

<ma'lumoti>, <lavozimi> maydonlaridan, ISH\_HAQI fayli ish haqi bo'yicha <lavozim>,<ish haqi> maydonlaridan iborat. Ekranga korxonaxodimlari haqidagi ma'lumot beruvchi va <familiya va sharifi>, <tugilgan yili>, <ma'lumoti>, <lavozimi> va <ish haqi> ustunlaridan tashkil topgan jadval chop etilsin.

11. DARSXONA fayli <dars turi>,<auditoriya tartib raqami>,<auditoriya sig'imi> maydonlaridan tashkil topgan. DARS fayli esa <predmet>,<dars turi>,<talabalar soni> maydonlardan iborat. DARS faylidagi har bir predmet uchun unga mos keluvchi auditoriya tartib raqamlari chop etilsin.
12. AVTO fayli <tabel tartib raqami>,<avtomobil egasi familiyasi>,<avtomobil nomeri> va <avtomobil turi> maydonlari bilan, avtomobillar texnik holatlari TA\_MIR faylida <tabel tartib raqami>, <avtomobilning texnik holati> maydonlari bilan berilgan. Avtomobilning texnik holati uchta holatda bo'lishi mumkin: 1-yaxshi; 2-qoniqarli; 3-qoniqarsiz. Quyidagi masalalar yechilsin:
  - a) texnik holati yaxshi bo'lgan barcha avtomobillar haqidagi ma'lumot chop etilsin;
  - b) ko'rsatilgan avtomobil nomeri bo'yicha uning texnik holati aniqlansin;
  - d) texnik holati yaxshi bo'lgan barcha avtomobillar haqidagi ma'lumot fayldan o'chirilsin.
13. TELEFON fayli <telefon nomeri>, <abonent familiyasi>, <manzil> maydonlaridan iborat. Abonentning telefonda gaplashish vaqti GAPLASHUV faylida <telefon nomeri>, <gaplashuv vaqti> maydonlari bilan berilgan. Ayni vaqtdagi bir minutda gaplashish pul miqdori berilgan holat uchun quyidagilar amalga oshirilsin:
  - a) ko'rsatilgan abonentga telefon uchun to'lov qog'ozi chop etilsin.
  - b) gaplashuv vaqti 100 minutdan oshgan telefon nomerlari va ularning egalarining familiyalari chop etilsin.
14. TAOM faylida taom haqidagi ma'lumot <taom nomi>, <kaloriyasi> va <narxi> maydonlari bilan berilgan. KOMPLEKS faylida esa shu taomlardan tashkil qilingan taomlar majmualari <majmua tartib raqami>,<taom nomi> maydonlari orqali ko'rsatilgan. Har bir taom majmuasining tarkibi, umumiy kaloriyasi va narxi chop etilsin.
15. TEATR faylida <teatr kodi>, <teatr nomi> va <teatrdagi o'rinlar soni> maydonlari, BILET faylida <teatr kodi>, <spektakl nomi> va <sotilgan biletlar soni> maydonlari berilgan. Quyidagilar aniqlansin:
  - a) bo'sh o'rinlari bor bo'lgan teatr va spektakl nomi;
  - b) ko'rsatilgan spektaklga bo'sh o'rin bor yoki yo'qligi;

- d) ko‘rastilgan teatrda qanday spektakl bo‘laytganligi.
16. Yo‘lovchi umumiy yukini tavsiflovchi kattalik – umumiy yuk tarkibiga kiruvchi yuklar soni va umumiy og‘irligi hisoblanadi.  
Berilgan YO\_LOVCHI faylda yo‘lovchilarning yuklari haqidagi ma’lumot saqlanadi.  
Quyidagi masalalar yechilsin:
- a) shunday yuklar topilsinki, ularning har birining og‘irligi barcha yuklar o‘rtacha og‘irligidan farqi 0.3 kg.dan oshmasin;
  - b) yuklar soni bir xil, yuklar og‘irligi bir biridan 0.5 kilogrammdan ko‘p farq qilmaydigan ikkita yo‘lovchi bor yoki yoqligi aniqlansin;
  - d) yuklar soni va yuklar og‘irligi boyicha boshqa yo‘lovchilardan ortiq bolgan yo‘lovchi bor yoki yoqligi aniqlansin;
  - e) bitta yuki 30 kilogrammdan kam bo‘lmagan yo‘lovchi bor yoki yoqligi aniqlansin.
17. Eksport qilinuvchi tovarlar haqidagi ma’lumot – <tovar nomi>,<tovar export qilayotgan davlat nomi>,<tovar bahosi> va <tovar hajmi>,<tovar soni> EXPORT faylida berilgan. Ma’lum bir tovarni eksport qiluvchi davlat va shu tovarning umumiy eksportdagi hajmi aniqlansin.
18. Butun sonlardan iborat binar SERIYA fayli berilgan. Undagi seriyalar chop etilsin (ya‘ni, bir xil sonlardan iborat ketma-ketlik ostilari).
19. Haqiqiy sonlardan iborat binar LOKAL\_MM fayli berilgan. Fayldagi lokal minimumlar va maximumlar miqdori aniqlansin. Lokal maksimum (minimum) deb sonlar ketma-ketligidagi  $a_{i-1} < a_i > a_{i+1}$  ( $a_{i-1} > a_i < a_{i+1}$ ) shartni qanoatlantiruvchi  $a_i$  soniga aytiladi.
20. Haqiqiy sonlardan iborat SONLAR binar fayli berilgan. SONLAR faylining juft o‘rindagi (0,2,4,...) elementlaridan JUFT\_SONLAR fayli, toq o‘rindagi (1,3,5,...) elementlari TOQ\_SONLAR fayli hosil qilinsin.
21. Haqiqiy sonlardan iborat binar SON\_KVADRATI fayli berilgan. Undagi barcha elementlar (sonlar) kvadratlari bilan almashtirilsin.
22. Haqiqiy sonlardan iborat YON\_ARIFM binar fayli berilgan. Fayl boshidagi va oxiridagi elementlaridan boshqa har bir elementlar, o‘zi va ikkita yon qo‘shnilarining o‘rta arifmetigi bilan almashtirilsin.
23. Butun  $a_i, i = \overline{0, N-1}$  sonlardan iborat binary MASSIV\_A fayli berilgan ( $N > 0$ ). Fayldagi sonlarning boshlang‘ich joylashuvi quyidagicha o‘zgartirilsin:  $a_0, a_{n-1}, a_1, a_{n-2}, a_2, a_{n-3}, \dots$
24. Binar MATRITSA faylida butun  $n$  va  $k$  soni, hamda  $n$  kvadrat haqiqiy matritsa yuqori uchburchagi bilan ( $k=1$ ) yoki pastki uchburchagi bilan ( $k=2$ ) yoki uchta diagonal elementlari (bosh diagonal elementi va uning yonidagi 2 ta elementlar) bilan ( $k=3$ ) berilgan. Klaviaturadan

kiritilgan natural  $i$  va  $j$  sonlari uchun matritsaning  $i$  - satr va  $j$  - ustunda turgan elementi mavjud bo'lsa, u chop qilinsin. Bunday element faylda bo'lmasa va indexlar matritsa chegarasidan chiqmasa 0 qiymati, aks holda null chop qilinsin.

25. Bosh diagonalga nisbatan simmetrik bo'lgan  $A$  va  $B$  matritsalarini yuqori yoki pastki uchburchak elementlarini o'zida saqlagan  $MATR\_A$ ,  $MATR\_B$  fayllari berilgan. Yangi  $MATR\_C$  fayli  $A*B$  elementlaridan tashkil topsin.
26. Butun  $n$ ,  $m$  va  $S$  sonlari berilgan ( $0 < n, m < 10, -108 < s < 108$ ).  $n \times m$  o'lchamdagi  $S = \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} a_{ij}$  shartni qanoatlantiruvchi  $A$  matritsa hosil qilinsin.  $f$  binar faylga  $n$ ,  $m$  va  $A$  matritsa yozilsin.

## 21. Ko'rsatkich turi

### *Namunaviy masala*

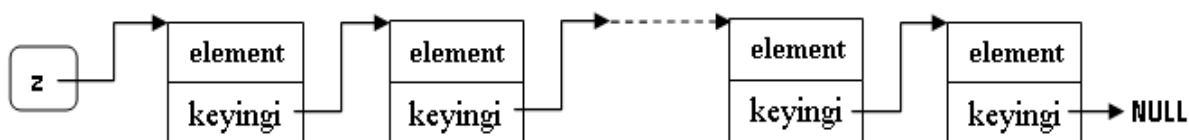
Noldan farqli butun sonlardan iborat chiziqli ro'yxat (zanjir) yaratilsin va undan ko'rsatilgan songa teng element o'chirilsin.

### *Yechish usuli*

Butun sonlarning chiziqli ro'yxatning (zanjirning) xalqasi butun turdagi element va xuddi shunday zanjirga ko'rsatkich maydonlaridan tashkil topgan dinamik struktura ko'rinishida bo'ladi:

```
struct Zanjir{int element; Zanjir * keyingi};
```

Zanjirning ( $Zanjir * z$ ) ko'rinishi quyidagi rasmda keltirilgan.



Ilova ning bosh funksiyasida chiziqli ro'yxat hosil qilish uchun  $Zanjir$  turidagi zanjir o'zgaruvchisi aniqlanadi va unga bo'sh ko'rsatkich qiymati 0 beriladi (uning ekvivalenti - NULL). Keyin takrorlash operatori tanasida klaviaturadan butun son kiritiladi va zanjirga element joylshatirish funksiyasini chaqirish orqali bu son ro'yxatning oxiriga qo'shiladi. Funksiya yangi hosil bo'lgan ro'yxat boshining adresini yana zanjir o'zgaruvchisiga qaytaradi. Agar klaviaturadan 0 soni kiritilsa ro'yxatni hosil qilish jarayoni tugaydi. Ro'yxat ustida amal sifatida berilgan son bilan ustma-ust tushadigan elementlarni o'chirish uchun o'chiriladigan son o'zgaruvchiga o'qiladi va u zanjir elementini o'chirish funksiyasi chaqirishida argument sifatida uzatiladi. Funksiya bu son bilan ustma-ust tushadigan ro'yxat elementlarini o'chiradi (agar bunday element mavjud



bo'lsa) va o'zgartirilgan ro'yxat boshining adresini zanjir o'zgaruvchisiga qaytarib beradi.

***Hlova matni:***

```
#include <iostream.h>
struct Zanjir { int element; Zanjir * keyingi; };
Zanjir * Element_Joylash(Zanjir * z, int yangi_elem)
{
    Zanjir * yangi=new Zanjir;
    yangi->element=yangi_elem;
    yangi->keyingi=0;
    if(z) // ro'yxat bo'sh emas
    {
        Zanjir * temp=z;
        while(temp->keyingi) // ro'yxat oxirgi elementini olish
        temp=temp->keyingi; // yangi elementni ro'yxat oxiriga qo'shish
    }
    else z=yangi; // ro'yxat bo'sh
    return z; // ro'yxat bo'shi adresini qaytarish
}
Zanjir * Element_Uchirish(Zanjir * z, int del_elem)
{
    if(z) { Zanjir* temp=z; Zanjir* oldingi=0; // joriy elementdan olingisiga ko'rsatgich
    while (temp)
    {
        if (temp->element==del_elem)
        {
            if(oldingi) // o'chiriladigan element birinchi emas
            { // o'chiriladigan elementdan oldingisini keyngisi bilan ulash
                oldingi->keyingi = temp->keyingi;
                delete temp; // elementni o'chirish
                temp=oldingi->keyingi;
            }
            else // o'chiriladigan element ro'yxat boshida
            { z=z->keyingi; delete temp; temp=z; }
        }
        else // element o'chiriladigan sondan farqli
        { oldingi=temp; temp=temp->keyingi; }
    }
}
```

```

return z;
}
void Zanjir_Ekranga(Zanjir * z)
{
cout<<"Zanjir elementlari:"<<endl;
Zanjir * temp=z;
while(temp)
{
cout<<temp->element<<' '; temp=temp->keyingi;
}
cout<<endl;
}
Zanjir * Zanjirni_Uchirish(Zanjir * z)
{
Zanjir * temp;
while(z)
{
temp=z; z=z->keyingi; delete temp;
}
return z;
}
int main()
{
Zanjir * zanjir=0;
int son,del_element;
do
{
cout<<"\nSonni kiriting (0-jaryon tugatish): ";
cin>>son;
if (son) zanjir=Element_Joylash(zanjir,son);
}
while(son);
Zanjir_Ekranga(zanjir);
cout<<"\nO'chiriladigan elementni kiriting: ";
cin>>del_element;
zanjir= Element_Uchirish(zanjir,del_element);
Zanjir_Ekranga(zanjir);
Zanjir = Zanjirni_Uchirish(zanjir);
return 0;
}

```

### *Amaliy topshiriqlar*

*1-23 - masalalarda Z o'zgaruvchi zanjirga ko'rsatkich: "Zanjir \* Z;"*

1. Z zanjir butun turdagi elementga (element maydoni) ega. Z zanjir elementlarining o'rta arifmetigi topilsin.
2. Z zanjir elementlari satr turida. Z zanjirga kiruvchi barcha  $Suz_1$  qiymatiga teng elementlar  $Suz_2$  bilan almashtirilsin.
3. Z zanjir elementi satr turida. Z zanjirdagi bir xil belgi bilan boshlanuvchi va tugaydigan elementlar soni aniqlansin.
4. Z zanjir elementi satr turida. Z zanjirning oxirgi elementi bilan ustma-ust tushadigan elementlar soni aniqlansin.
5. Z zanjir elementlari butun turda. Z zanjirdan ikkita  $Z_1$  va  $Z_2$  zanjirlar hosil qilinsin. Bunda  $Z_1$  - zanjirning musbat elementlari va  $Z_2$  - qolgan elementlari.
6. Oraga qo'yishni bajaruvchi funksiyalar tuzilsin:
  - a) Z zanjir boshiga yangi Yangi elementni qo'yadigan;
  - b) Z zanjir oxiriga yangi Yangi elementni qo'yadigan.
7. Oraga qo'yishni bajaruvchi funksiyalar tuzilsin:
  - a) bo'sh bo'lmagan Z zanjirining birinchi elementidan keyin yangi y elementni;
  - b) Z zanjiriga kiruvchi har bir y elementdan keyin yangi x elementni joylashtiradigan.
8. Bo'sh bo'lmagan va tartiblangan Z zanjirga yangi y elementni shunday joylashtiringki, unda Z zanjirdagi tartib buzilmasin.
9. Zanjirning elementlarini o'chiradigan funksiya tuzilsin:
  - a) bo'sh bo'lmagan Z zanjirning birinchi elementini;
  - b) Z zanjirning ikkinchi elementini, agar u mavjud bo'lsa.
10. Butun turdagi elementlardan iborat Z zanjir elementini o'chiradigan funksiya tuzilsin:
  - a) bo'sh bo'lmagan Z zanjirning oxirgi elementini;
  - b) Z zanjirdan birinchi manfiy element, agar u mavjud bo'lsa;
  - d) Z zanjirdagi barcha manfiy elementlarni.
11. Klaviaturadan matni (satrni) o'qib, teskari tartibda chop qilinsin. Bunda matndagi har bir so'z zanjir elementi sifatida qaralsin.
12. Butun  $n$  ( $n > 1$ ) va  $n$  ta haqiqiy sonlar berilgan. Bu sonlar kamaymaydigan tartibda chop qilinsin. Bunda har bir son zanjir elementi sifatida qaralsin.
13. Quyidagi amallarni bajaruvchi funksiya tuzilsin:
  - a)  $Z_1$  va  $Z_2$  zanjirlar tengligini tekshiruvchi;

- b)  $Z_1$  zanjir  $Z_2$  zanjirga kirishini aniqlovchi.
14. Quyidagi amallarni bajaruvchi funksiya tuzilsin:
- $Z$  zanjirda kamida ikkita bir xil element bor-yo'qligini aniqlovchi;
  - bo'sh bo'lmagan  $Z$  zanjir oxiriga uning birinchi elementini olib o'tuvchi;
  - bo'sh bo'lmagan  $Z$  zanjir boshiga uning oxirgi elementini olib o'tuvchi.
15. Quyidagi amallarni bajaruvchi funksiya tuzilsin:
- $Z_1$  zanjir oxiriga  $Z_2$  zanjirining barcha elementlarini qo'shuvchi;
  - agar  $Z_1$  zanjirda  $x$  element mavjud bo'lsa, uning davomiga  $Z_2$  zanjirning barcha elementlarini qo'shuvchi.
16. Quyidagi amallarni bajaruvchi funksiya tuzilsin:
- $Z$  zanjirda ketma-ket keluvchi teng qiymatli elementlar guruhidan bittasini qoldiruvchi;
  - $Z$  zanjirda bir xil qiymatli elementlardan faqat bittasini qoldiruvchi.
17. Berilgan shartlarni bajaruvchi rekursiv funksiya tuzilsin:
- $e$  element  $Z$  zanjirga kirishi yoki kirmasligini aniqlovchi;
  - $e$  element  $Z$  zanjirga necha marta kirishini hisoblovchi.
18. Quyidagi shartlarni bajaruvchi rekursiv funksiya tuzilsin:
- bo'sh bo'lmagan, haqiqiy turdagi elementli  $Z$  zanjirga kiruvchi maksimal qiymatli elementni topuvchi;
  - $Z$  zanjir elementlarini teskari tartibda chop qiluvchi.
19. Quyidagi shartlarni bajaruvchi rekursiv funksiya tuzilsin:
- $Z$  zanjirdagi barcha  $e_1$  elementni  $e_2$  elementga almashtiruvchi;
  - $Z$  zanjirdan qiymati  $e$  ga teng bo'lgan elementlarning birinchisini o'chiruvchi (agar u mavjud bo'lsa).
20. Quyidagi shartlarni bajaruvchi rekursiv funksiya tuzilsin:
- $Z$  zanjirdan berilgan  $e$  elementning barchasini o'chiruvchi;
  - $Z$  zanjirining nusxasi –  $Z_1$  zanjirni hosil qiluvchi.
21. Berilgan  $Z_1$  va  $Z_2$  zanjirlar bo'yicha quyidagi shartlarni qanoatlantiruvchi elementlardan tashkil topgan  $Z$  zanjir hosil qilinsin:
- $Z_1$  va  $Z_2$  zanjirlarining kamida bittasiga kiruvchi;
  - bir vaqtda  $Z_1$  va  $Z_2$  zanjirlariga kiruvchi;
  - $Z_1$  zanjirga kiruvchi, lekin  $Z_2$  zanjirga kirmaydigan;
  - $Z_1$  va  $Z_2$  zanjirlardan biriga kiruvchi, lekin ikkinchisiga kirmaydigan.
22. Kamayuvchi bo'lmagan  $Z_1$  va  $Z_2$  zanjirlarni kamayuvchi bo'lmagan zanjirga birlashtiruvchi funksiya tuzilsin:
- yangi  $Z$  zanjir qurish orqali;
  - $Z_1$  va  $Z_2$  zanjirdagi ko'rsatgichlarni mos ravishda o'zgartirish va

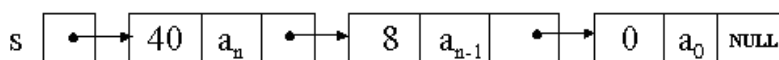
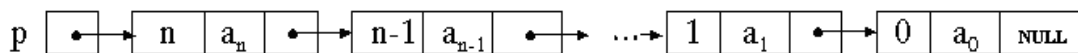
natijani  $Z_1$  zanjirida hosil qilish orqali.

23.  $Z$  zanjiridagi  $Z_1$  zanjir ostining birinchi kirishini  $Z_2$  zanjir bilan almashtiruvchi funksiya tuzilsin.

24. Butun koeffitsiyentli

$$P(x) = a_n x^n + a_{n-1} x^{n-1} + a_1 x + a_0$$

ko'phadni zanjir ko'rinishida tasvirlash mumkin (quyidagi rasmdagi  $p$  zanjir), agar  $a_i = 0$  bo'lsa, mos xalqa zanjirga kiritilmaydi. Quyidagi rasmda  $s$  zanjir  $S(x) = 52x^{40} - 3x^8 + x$  ko'phadga mos zanjir keltirilgan.



Ko'phadni tavsiflashning zanjir ko'rinishiga mos turlar e'lon qilinsin va bu zanjir ustida quyidagi amallarni bajaruvchi funksiyalar tuzilsin:

a)  $p$  va  $q$  ko'phadlar tengligini tekshiruvchi  $Teng(p,q)$  mantiqiy funksiyasi;

b) berilgan  $x$  uchun  $p$  ko'phadning qiymatini hisoblovchi  $Qiymat(p,x)$  funksiyasi;

d)  $p$  ko'phadning hosilasi bo'lgan  $q$  ko'phadni quruvchi  $Hosila(p,q)$  funksiyasi;

e)  $q$  va  $r$  ko'phadlar yig'indisi bo'lgan  $p$  ko'phadni quruvchi  $Yig'indi(p,q,r)$  funksiyasi;

f)  $p$  ko'phadni  $v$  o'zgaruvchining qiymati bo'lgan (bitta harfli) o'zgaruvchi nomida chop qiluvchi  $Chop\_Qilish(p,v)$  funksiyasi tuzilsin. Misol uchun,  $s$ -ko'phad uchun  $Chop\_Qilish(S,'y')$  funksiyasi " $52y^{40} - 3y^8 + y$ " ifodasini chop qiladi.

25. ("*Sanagich*")  $n$  ta bola aylana bo'ylab turibdi. Sanoq birinchisidan boshlanib,  $k$ -bola davradan chiqariladi va har bir chiqarishdan keyin davra qisqaradi. Keyingi sanoq  $k+1$ -boladan boshlanadi. Bolalarni davradan chiqib ketish tartibi aniqlansin. Ilova uchun boshlang'ich berilganlar  $n$  va  $k$  natural sonlar bo'lib, uning natijasi - davradan chiqib ketuvchi bolalarning boshlang'ich tartib nomerlari ketma-ketligi.

26. Berilgan matn (satr) simmetrikligi aniqlansin. So'zlar zanjir elementi deb qaralsin.

27. Kamida ikkita har xil va 0 bilan tugaydigan natural sonlar ketma-ketligi berilgan. Eng katta va eng kichik sonlar o'rtasidagi sonlar teskari tartibda chop qilinsin. Sonlar zanjir elementi deb qaralsin.

28. Natural  $N$  va  $M$  sonlari berilgan bo‘lib,  $\frac{N}{M}$  kasrning o‘nlik ko‘rinishi chop etilsin. Agar kasr davriy qismlardan iborat bo‘lsa, ularning birinchisini qavsga olish bilan cheklanilsin (masalan,  $\frac{7}{15} = 0,4(6)$ ). Sonning kasr qismidagi raqamlar dinamik zanjir elementi deb qaralsin.

## 22. Sinflar. Inkapsulyatsiya

Quyidagi masalalarni yechishda sinf yaratilishi va unda qo‘yilgan masalani to‘liq qamrab oluvchi berilganlar-a‘zolar va funksiya-a‘zolar aniqlanishi zarur.

### *Namunaviy masala*

Berilgan matn Shenon usuli bilan shifrlansin va qayta tiklansin.

### *Yechish usuli*

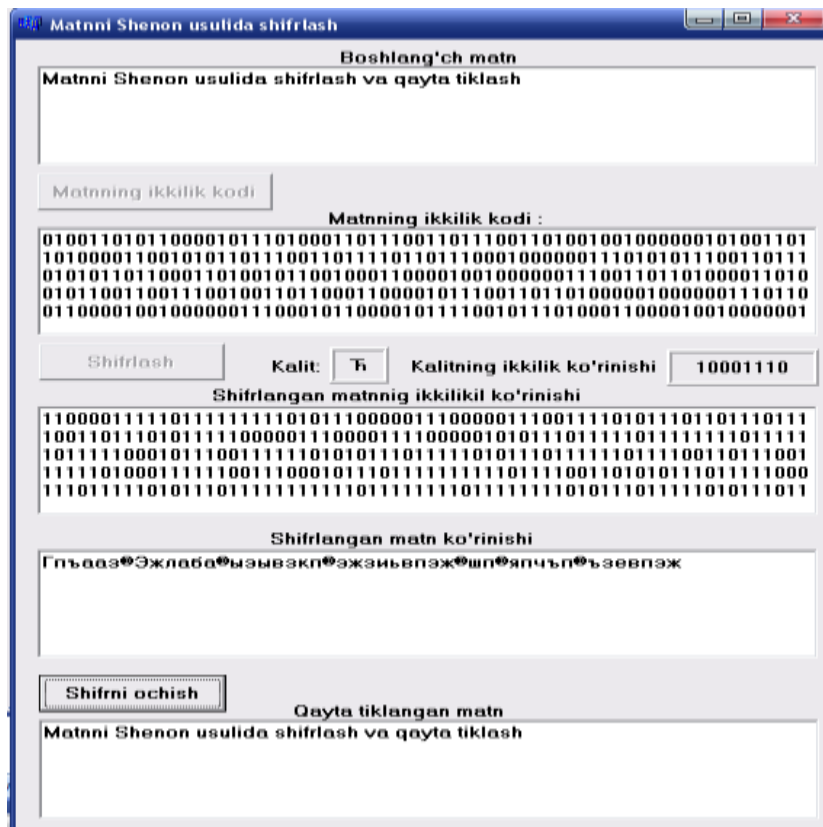
Shenon usulining mohiyatida  $a$  baytni  $b$  baytga “*inkor qiluvchi yoki*” (“ $\wedge$ ”) amalini ketma-ket ravishda ikki marta qo‘llash natijasida yana  $a$  baytning o‘zi tiklanishi yotadi:  $a \wedge b \wedge b = a$ .

Shifrlanuvchi matn ( $Matn_0$ ) satrlar ko‘rinishida, shifrlash kaliti ( $Kalit$ ) esa bayt ko‘rinishida beriladi.  $Matn_0$  matnning satrlaridagi har bir bayt bilan  $Kalit$  o‘rtasida razryadli ‘ $\sim$ ’ amalini bajarish orqali shifrlangan  $Matn_1$  matni hosil qilinadi. Xuddi shu amalni  $Matn_1$  matniga qo‘llash orqali  $Matn_0$  matni qayta tiklanadi.

Masalani yechish algoritmi quyidagi qadamlardan iborat bo‘ladi:

1.  $Matn_0$  satrlari kiritiladi;
2.  $Kalit$  qiymati tasoddiy ravishda hosil qilinadi;
3.  $Matn_0$  matnning har bir satrini  $Kalit$  bilan shifrlash asosida  $Matn_1$  satrlari hosil qilinadi.
4.  $Matn_1$  matnning har bir satrini  $Kalit$  bilan “*qayta shifrlash*” asosida  $Matn_2$  matni hosil qilinadi.

Ilovaning bajarilishi quyidagi rasmda ko‘rsatilgan.



### *Ilova matni*

```

TForm *Form1;
class Shifr_Shenon
{
    String matn0,matn1,matn2, kalit;
    unsigned char kalit_son;
    String _10dan_2ga(unsigned char c);
    String matn_bin_matn_char(String mtm);
public:
    void Matnni_kiritish(String _S);
    void Shifrlash();
    void Shifni_ochish();
    String Matndan_01ga(String);
    String _01dan_Matnga();
    char _01dan_Belgiga(String);
    String Kalitni_olish(){return kalit;};
    char Kalit_belgini_olish(){return kalit_son;};
    String Matn0_olish(){return matn0;};
    String Matn1_olish(){return matn1;};
    String Matn_01dan_Matn_char(int);
};
String Shifr_Shenon::_10dan_2ga(unsigned char c)

```

```

{
String Dicemal="";
for (int k=1; k<=8;k++)
{
if(c&0x80) Dicemal=Dicemal+'1';
else Dicemal=Dicemal+'0'; c=c<<1;
}
return Dicemal;
};
void Shifr_Shenon::Matnni_kiritish(String _S)
{
matn0=Matndan_01ga(_S);
unsigned int kod;
randomize();
kod=rand();
kalit_son=kod%256;
kalit=_10dan_2ga(kalit_son);
while(kalit[1]!='0') kalit.Delete(1,1);
};
void Shifr_Shenon::Shifrlash()
{
matn1.operator =("");
int len_kalit=kalit.Length();
for(int i=0;i<matn0.Length();i++)
{
if(matn0[i+1]=='0'&&kalit[i%len_kalit+1]=='0'
|| matn0[i+1]=='1'&&kalit[i%len_kalit+1]=='1') matn1+="0";
else matn1+="1";
}
};
void Shifr_Shenon::Shifrni_ochish()
{
matn2.operator =("");
int len_kalit=kalit.Length();
for(int i=0; i<matn1.Length();i++)
{
if(matn1[i+1]=='0'&&kalit[i%len_kalit+1]=='0'
|| matn1[i+1]=='1'&&kalit[i%len_kalit+1]=='1') matn2+="0";
else matn2+="1";
}
}

```



```

};
String Shifr_Shenon::Matndan_01ga(String matn)
{
    String sbin="";
    for(int i=1; i<=matn.Length(); i++)
    {
        unsigned char c=(unsigned char)matn[i]; sbin.operator+=(_10dan_2ga(c)); }
    return sbin;
};
char Shifr_Shenon::_01dan_Belgiga(String sbin)
{
    unsigned int koda=0, pow2=1;
    for(int i=8;i>=1;i--)
    {
        koda+=((unsigned char)(sbin[i])-48)*pow2;
        pow2*=2;
    }
    return koda;
}
String Shifr_Shenon::matn_bin_matn_char(String mtn)
{
    String matn="";
    while (mtn.Length()) {matn+=_01dan_Belgiga(mtn.SubString(1,8));
mtn.Delete(1,8);}
    return matn;
};
String Shifr_Shenon::Matn_01dan_Matn_char(int imatn)
{
    if(imatn==1) return matn_bin_matn_char(matn1);
    return matn_bin_matn_char(matn2);
};
Shifr_Shenon shenon;
void __fastcall TForm::Button_shifrlashClick(TObject *Sender)
{
    Button_shifrlash->Enabled=0;
    shenon.Shifrlash();
    Memo_matn1_01->Lines->Text=shenon.Matn1_olish();
    Memo_matn1->Lines->Text=shenon.Matn_01dan_Matn_char(1);
    Button_shifrn_ochish->Enabled=1;
}

```

```

void __fastcall TForm::Button_shifrni_ochishClick(TObject *Sender)
{
    shenon.Shifrni_ochish();
    Memo_matn2->Lines->Text=shenon.Matn_01dan_Matn_char(2);
}
void __fastcall TForm::Edit_kalitChange(TObject *Sender)
{
    if (Memo_matn0_01->Lines->Text!="" &&Panel_Kalit->Caption!="")
        Button_shifrlash->Enabled =1;
    else Button_shifrlash->Enabled =0;
}
void __fastcall TForm::Memo_matn0Change(TObject *Sender)
{
    if (Memo_matn0->Lines->Text!="") Button_01_kod->Enabled =1;
    else Button_01_kod->Enabled =0;
    Memo_matn0_01->Lines->Clear();
    Memo_matn1_01->Lines->Clear();
    Memo_matn2->Lines->Clear();
}
void __fastcall TForm::Button_01_kodClick(TObject *Sender)
{
    shenon.Matnni_kiritish(Memo_matn0->Lines->Text);
    Memo_matn0_01->Lines->Text=shenon.Matn0_olish();
    Panel_Kalit_Bin->Caption=shenon.Kalitni_olish();
    Panel_Kalit->Caption=shenon.Kalit_belgini_olish();
    Button_shifrlash->Enabled=1;
    Button_01_kod->Enabled=0;
}
void __fastcall TForm::FormCreate(TObject *Sender)
{
    Button_01_kod->Enabled=0;
    Button_shifrlash->Enabled=0;
    Button_shifrni_ochish->Enabled=0;
    Memo_matn0_01->ReadOnly=1;
    Memo_matn1_01->ReadOnly=1;
    Memo_matn2->ReadOnly=1;
}

```

### ***Amaliy topshiriqlar***

1. 10 lik sanoq sistemasida berilgan sonni 2, 8 va 16 lik sanoq sistemasidagi ko‘rinishini chop qiluvchi SANOQ\_SISTEMA sinfi yaratilsin.

2. Kompleks sonlar ustida arifmetik amallarni bajaruvchi KOMPLEKS sinfi yaratilsin.
3. Berilgan natural  $n$  soni uchun  $n \times n$  o'lchamidagi  $A$  matritsani maksimal va minimal elementini topadigan, uning bosh diagonalga nisbatan simmetrik ekanligini aniqlaydigan, transponerlangan ko'rinishini chop etuvchi funksiya-a'zolarini o'z ichiga oluvchi MATRITSA sinfi yaratilsin.
4. Uch o'lchamli fazoda koordinatalari bilan berilgan ikkita vektorni tavsiflovchi VEKTOR2\_3D sinfi aniqlansin. Sinfda vektorlarni qo'shish va ayirish orqali yangi vektorlar hosil qiluvchi, ikkita vektorning skalyar ko'paytmasini, vektor uzunligini va ikkita vektorlar orasidagi burchak kosinusini hisoblovchi funksiya-a'zolar aniqlansin.
5. Ko'phad darajasi va koeffitsientlari bilan berilgan bitta o'zgavuchili ko'phadni tavsiflovchi KO\_PHAD sinfi yaratilsin. Sinfda ko'phad berilgan argumentdagi qiymatini hisoblovchi, ko'rsatilgan tartibdagi hosilasini topadigan funksiya-a'zolar aniqlansin.
6. Uy kutubxonasini tavsiflovchi UY\_KUTUBXONASI sinfi aniqlansin. Unda ixtiyoriy sondagi kitoblar bilan ishlash, qandaydir alomati bo'yicha kitobni izlash (muallif yoki yil bo'yicha), yangi kitobni qo'shish va o'chirish imkoniyatlari bo'lsin.
7. Yon daftarni o'zida aks ettiruvchi YON\_DAFTAR sinfi yaratilsin. Unda ixtiyoriy sondagi yozuvlar bilan ishlash, qandaydir alomati bo'yicha yozuvni izlash (familiya, tug'ilgan yili yoki telefon nomeri bo'yicha), yangi yozuvni qo'shish va o'chirish imkoniyatlari bo'lsin.
8. Talabalar guruhini tavsiflovchi TALABA\_GURUHI sinfi yaratilsin. Unda ixtiyoriy sondagi talabalar bilan ishlash, qandaydir alomati bo'yicha talabani izlash (familiya, tug'ilgan yili yoki telefon nomeri bo'yicha), yangi yozuvni qo'shish, o'chirish va tartiblash imkoniyatlari bo'lsin.
9. Stek ustidagi amallarni bajaruvchi STEK sinfi aniqlansin. Ushbu sinfdan labirintdan chiqish masalasini yechishda foydalanilsin. Labirint kvadratlardan tashkil topgan matritsa ko'rinishida beriladi. Har bir kvadrat ochiq yoki yopiq bo'ladi. Yopiq kvadratga kirish mumkin emas. Agar kvadrat ochiq bo'lsa uning yon tomonidan kirish mumkin (burchagidan kirish mumkin emas). Har bir kvadrat uning matritsadagi koordinatalari bilan beriladi. Labirintdan chiqish amalga oshirilganda topilgan yo'l chop qilinadi (kvadratlar koordinatalari juftliklarining ketma-ketligi).
10. YUGURUVCHI sinfi yugurish musobaqasi natijalari haqidagi <yuguruvchi familiyasi va inisiallari>, <jamoa nomi> va <masofani bosib o'tgan vaqti (sekundlarda)> berilgan-a'zolarini o'z ichiga oladi. Berilgan  $n$  o'lchamidagi YUGURUVCHI sinf ob'ektlari massivi yaratilsin va

quyidagi amallar bajarilsin:

a) massiv yuguruvchilarni masofani bosib o'tgan vaqtining kamayishi bo'yicha tartiblansin;

b) jamoa a'zolarining o'rtacha yugurish vaqti bo'yicha yuqori natija ko'rsatgan uchta jamoa nomlari chop etilsin.

11. FUTBOL sinfi jamoasining o'yin natijalari haqidagi <jamoa nomi>, <g'alabalar soni>, <duranglar soni>, <mag'lubiyatlar soni>, <kiritgan to'plar soni> va <o'tkazgan to'plar soni> berilgan-a'zolari o'z ichiga oladi.

Berilgan n uchun FUTBOL sinfi obyektlari massivi hosil qilinsin va to'plagan ochkolari bo'yicha jamolar jadvali chop etilsin. Bunda quyidagilarga e'tibor berilsin: agar ikkita jamoaning ochkolari teng bo'lsa, kiritilgan va o'tkazib yuborilgan to'plar farqi qaraladi. Farqi katta bo'lgan jamoa uyqori qatorga o'tadi, aks holda qur'a tashlanadi va shunga qarab jamoa o'rni aniqlanadi.

12. AVTOMOBIL sinfida avtomobil va uning egasi haqidagi <avtomobil egasining familiyasi va sharifi>,<avtomobil rusumi>,<avtomobil raqamii> berilgan-a'zolari aniqlangan. Berilgan n o'lchamidagi AVTOMOBIL sinfi obyektlari massivi yaratilsin va quyidagi amallar bajarilsin:

a) massiv avtomobil egalari familiyalarini alfavit bo'yicha joylashuviga mos tartiblansin;

b) kiritilgan avtomobil rusumidagi avtomobil egalari haqidagi ma'lumot chop etilsin;

d) kiritilgan avtomobil rusumi va nomeri bo'yicha avtomobil egasining familiyasi chop etilsin.

13. Nuqta bilan tugaydigan matn berilgan. Uning formula ekanligi quyidagi qoidalar asosida tekshirilsin. Natija ijobiy bo'lgan holda formula qiymati hisoblansin:

<formula>::=<raqam>|(<formula><amal><formula>)

<amal>::= + | - | \*

<raqam>::=0|1|2|3|4|5|6|7|8|9

Masalan, "5" formula qiymati 5, "((2-4)\*6)" formula qiymati 12.

14. SMS\_XABAR sinfi yaratilsin. Unda uyali telefondagi tugmalar orqali xabar terish jarayoni amalga oshirilsin. Tugma ma'lum bir vaqt intervalida ketma-ket bosilganda unga bog'langan harf va raqamlarning matnda paydo bo'lishi, hamda tugma bosishda vaqt intervalidan oshganda kursorni bir o'rin o'ngga siljishi amalga oshirilsin. Sinfda matni o'chirish, kursorni harakatlantirish va xabarni ko'rsatilgan adressga (nomerga) jo'natilganligi haqida xabar beruvchi funksiy-a'zolar aniqlansin.

15. TO\_RTBURCHAK sinfi yaratilsin. Uning tarkibida tekislikdagi to'rtburchak uchlari - A,B,C va D nuqtalar kiritilganda:

- a) to'rtburchak mavjudligini;  
 b) agar to'rburchak mavjud bo'lsa, uning qavariq yoki yo'qligi;  
 d) to'rtburchakning yuzasi va perimetrini hisoblaydigan funksiya-a'zolar aniqlansin.
16. Butun sonlar juftligi bilan berilgan ratsional sonlar ustida amal bajaruvchi RATSIONAL sinfi aniqlansin. Sinf obyektlari massivini yaratilsin va sinfning do'st funksiyalari yordamida quyidagi masalalar yechilsin:  
 a) berilgan  $a$  va  $b$  ratsional sonlarning tengligi tekshirilsin;  
 b) berilgan  $a$  va  $b$  ratsional sonlar yig'indisi  $r$  ratsionalga berilsin;  
 d) berilgan  $r$  ratsional son qisqarmaydigan ko'rinishga keltirilsin;  
 e) massiv ko'rinishida berilgan ratsional sonlarning eng kattasi topilsin.
17. O'zaro ekvivalent formullardan foydalangan holda berilgan mantiqiy ifodani soddalashtirish amalini bajaradigan MANTIQUIY\_IFODA sinfi yaratilsin. Quyida o'zaro ekvivalent formulalar ro'yxati keltirilgan.
- 1)  $\neg\neg x \equiv x$ ;                      2)  $x \vee \neg x \equiv 1$ ;                      3)  $x \wedge 1 \equiv x$ ;  
 4)  $x \vee 0 \equiv x$ ;                      5)  $x \vee 1 \equiv 1$ ;                      6)  $x \wedge 0 \equiv 0$ ;  
 7)  $x \vee (x \wedge y) \equiv x$ ;                      8)  $x \wedge (x \vee y) \equiv x$ ;                      9)  $x \wedge x \equiv x$ ;  
 10)  $x \vee x \equiv x$ ;                      11)  $x \wedge \neg x \equiv 0$ ;                      12)  $\neg(x \vee y) \equiv \neg x \wedge \neg y$ ;  
 13)  $\neg(x \wedge y) \equiv \neg x \vee \neg y$ ;                      14)  $x \wedge y \equiv y \wedge x$ ;                      15)  $x \vee y \equiv y \vee x$ ;  
 16)  $(x \wedge y) \wedge z \equiv x \wedge (y \wedge z)$ ;                      17)  $(x \vee y) \vee z \equiv x \vee (y \vee z)$ ;                      18)  $(\neg\neg x) \equiv x$ ;  
 19)  $x \wedge (y \vee z) \equiv (x \wedge y) \vee (x \wedge z)$ ;                      20)  $x \vee (y \wedge z) \equiv (x \vee y) \wedge (x \vee z)$ ;
18. AXBOROT\_ZICHLASH\_8\_7 sinfi yaratilsin va unda matni zichlashning "sakkizdan yettiga" usuli amalga oshirilsin. Bu usul ASCII turidagi 0..127 kodlari oralig'dagi belgilarda yozilgan matni zichlashtirishga mo'ljallangan. Algoritm asosida belgi kodidagi ishlatilmaydigan (barcha belgi uchun 0 bo'lgan) 7-razryaddan foydalanish yotadi. Matn 8 baytdan iborat bloklarga bo'linadi va 1-baytning 7 ta razryadidagi qiymatlar qolgan 7 ta baytlarning 7-razryadlariga ustun bo'yicha yoziladi.
- Misol uchun quyidagi 8-baytli blokdan 7 baytlik blok hosil bo'ladi:
- |                  |                          |                  |
|------------------|--------------------------|------------------|
| 1-bayt 0111 0101 |                          | 2-bayt 1111 1101 |
| 2-bayt 0111 1101 |                          | 3-bayt 1010 0011 |
| 3-bayt 0010 0011 |                          | 4-bayt 1101 0110 |
| 4-bayt 0101 0110 | hosil bo'ladigan 7 bayt: | 5-bayt 0001 0000 |
| 5-bayt 0001 0000 | (1-bayt 7-ustunda)       | 6-bayt 1110 1101 |
| 6-bayt 0110 1101 |                          | 7-bayt 0010 1010 |
| 7-bayt 0010 1010 |                          | 8-bayt 1111 1001 |
| 8-bayt 0111 1001 |                          |                  |
- Berilgan matn zichlashtirilsin va qayta tiklansin. Matn bloki 8 baytga

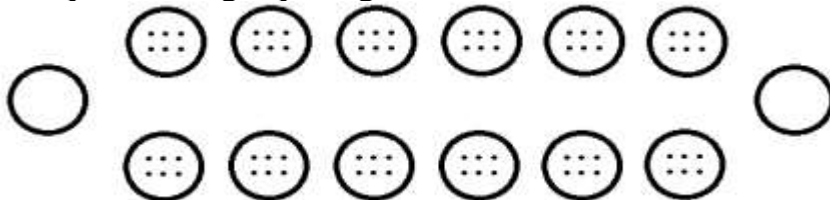
to'lmaganda u probel belgisi bilan to'ldirilsin.

19. SUDOKU sinfi yaratilsin. Uning tarkibiga sudoku o'yinini amalga oshiruvchi berilgan a'zolar va funksiya-a'zolar aniqlansin. O'yinning boshlang'ich holatida  $9 \times 9$  o'lchamidagi kataklar berilgan bo'lib, ular  $3 \times 3$  o'lchamdagi 9 ta kvadratlarga bo'lingan. Kataklarning ayrimlari 1 dan 9 gacha bo'lgan sonlar bilan to'ldirilgan. Quyidagi shart ostida qolgan bo'sh kataklar 1 dan 9 gacha bo'lgan

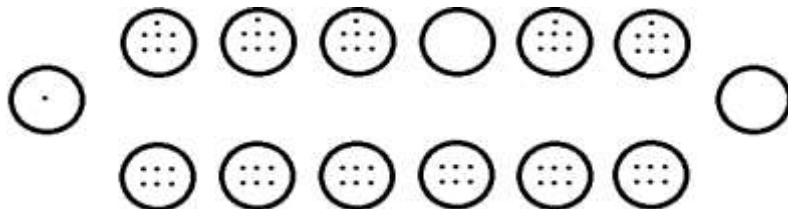
1		8		3		7		4
	6				5			
2			1					2
6					3	1		9
	3		4	1				
5		2			6			3
		4	7		9	8		5
7	9			6				1

sonlar bilan to'ldirilsin: katakka yoziladigan son katak joylashgan gorizontal va vertikal yo'nalishdagi kataklarda, hamda katak joylashgan kvadrat ichida takrorlanmasligi kerak. Rasmda Sudoku o'yinining boshlang'ich ko'rinishiga misol keltirilgan.

20. Kalax o'yinini amalga oshiruvchi KALAX sinfi aniqlansin. Kalax o'yinida ikkita ishtorokchi qatnashadi - odam va kompyuter. O'yinda ikki qatorda bir-biriga qarama-qarshi joylashgan, toshlar bilan to'ldirilgan oltitadan chuqurchalar, hamda qo'shimcha ravishda har bir o'yinchiga bittadan bo'sh chuqurcha (kalaxlar) bo'ladi. O'yin boshida har bir o'yinchi tomondagi chuqurchalar oltita toshcha bilan to'ldiriladi, kalaxdan tashqari (rasmga qarang).



O'yinning har bir qadamida ishtirokchilardan biri o'z tomonidagi qaysidir chuqurchadan barcha toshchalarni oladi va soat millari harakatiga teskari yo'nalishda sikllik ravishda o'z chuqurchalariga bittadan toshchani joylab chiqadi, shu jumladan, o'z kalaxiga ham (rasmga qarang).



Navbatdagi qadam quyidagi holatlar bo'yicha amalga oshiraladi:

- agar o'yinchi joylagan toshchalarning oxirigisi bo'sh chuqurchaga tushsa, shu chuqurchadagi toshcha va shu chuqurchaga mos (qarshisidagi) raqibning chuqurchasidagi barcha toshlar olinib o'z kalaxiga joylaydi, hamda o'yinni o'zi davom ettiradi;

- agar o‘yinchi joylagan toshchalarning oxirgisi kalaxga tushsa, o‘yinni o‘zi davom ettiradi;
  - boshqa holatlarda toshchalarni joylash navbati raqibga o‘tadi.
- Agar birorta o‘yinchi chuqurchalarining barchasi bo‘sh bo‘lib qolsa (raqib yurishi sababli bo‘lsa ham), raqib chuqurchalarida qolgan barcha toshlar olinib, o‘yinchi kalaxiga o‘tkaziladi va o‘yin nihoyasiga yetadi. O‘z kalaxida umumiy toshchalar sonining yarmidan ko‘pini yig‘gan o‘yinchi g‘olib hisoblalanadi.

## 23. Vorislik

Ushbu bo‘limda qo‘yilgan masala mazmunidan kelib chiqqan masalaning umumiy xususiytlarini o‘z ichiga olgan tayanch sinf yaratilishi va undan voris sifatida hosil bo‘lgan sinf esa bevosita qo‘yilgan masalani yechilishi kerak. Masala voris sinf ob‘yektini e‘lon qilib yechiladi.

### *Namunaviy masala*

Berilgan butun  $n$ , va haqiqiy  $x$  va  $y$  qiymatlari uchun  $10^y - x^y + x^n$  ifoda hisoblansin.

### *Yechish usuli*

Berilgan ifoda qiymati uchta ifoda ostilari qiymatlarini alohida hisoblash va mos arifmetik amallar bajarish ko‘rinishida bajariladi. Har bir ifoda o‘zaro vorislik munosabatida bo‘lgan uchta sinf obyektlari sifatida qaraladi.

Tayanch sinf sifatida  $x^n$  ifodani hisoblash uchun yaratilgan `lfoda_x_n` sinfi bo‘lib, uning tarkibiga butun  $n$ , haqiqiy  $x$  berilgan-a’zolari, berilgan-a’zolariga qiymat beruvchi konstruktor, nusxalash konstruktori va bevosita  $x^n$  qiymatni hisoblaydigan `Hisobla()` funksiya-a’zosi kiradi.

Tayanch `lfoda_x_n` vorisi sifatida  $x^y$  ifoda osti uchun `lfoda_x_y` sinfi yaratilgan. Unda haqiqiy  $y$  berilgan-a’zo aniqlangan bo‘lib,  $x$  qiymati tayanch sinfdan voris sifatida olinadi. Ifoda ostilari o‘zaro bog‘liqligini inobatga oladigan bo‘lsak, `lfoda_x_y` sinfi obyekti `lfoda_x_n` obyekt mavjud bo‘lgandagina yuzaga kelishi mumkin va shu sababli unda nusxalash konstruktorlari qo‘llanilgan. `Hisobla()` funksiyasi  $x^y$  ifoda ostisi qiymatini hisoblaydi.

O‘z navbatida  $10^y$  ifoda ostisi uchun `lfoda_10_y` sinfi `lfoda_x_y` sinfi vorisi qilib aniqlangan. Unda berilgan-a’zolar yo‘q,  $y$  qiymati ajdod sinflardan vorislik bo‘yicha olinadi. `Hisobla()` funksiya-a’zosi  $10^y$  qiymatini hisoblaydi.

Qo'yilgan masala mazmuniga ko'ra sinflar obyektlarini aniqlash qat'iy ketma-ketlikda amalga oshiriladi. Oldin lfoda\_x\_n, keyin unga bog'liq lfoda\_x\_y obyekt va nihoyat lfoda\_10\_y sinf obyektlari aniqlanadi va umumiy ifoda qiymati hisoblanadi.

### *Ilova matni*

#### *Unit>Ifoda.h fayli:*

```
class lfoda_x_n
{
protected:
double x;
int n;
public:
lfoda_x_n(int _n, double _x){ n=_n; x=_x;}
lfoda_x_n(lfoda_x_n & if1){ n=if1.n; x=if1.x; }
double Hisobla(){ return pow(x,n);}
};
class lfoda_x_y:public lfoda_x_n
{
protected:
double y;
public:
lfoda_x_y(double _y,lfoda_x_n & if1):lfoda_x_n(if1) { y=_y; }
lfoda_x_y(lfoda_x_y & if12): lfoda_x_n(if12.n,if12.x) { y=if12.y;}
double Hisobla(){ return pow(x,y);}
};
class lfoda_10_y:public lfoda_x_y
{
public:
lfoda_10_y(lfoda_x_y& if_1_2):lfoda_x_y(if_1_2){}
double Hisobla(){return pow(10,y);}
};
lfoda_x_n * if_xn;
lfoda_x_y* if_xy;
lfoda_10_y * if_10y;
```

#### *Unit>Ifoda.cpp fayli:*

```
void __fastcall TForm1::Btn_HisoblaClick(TObject *Sender)
{
if_xn=new lfoda_x_n(StrToInt(Edit_n->Text),StrToFloat(Edit_x->Text));
if_xy=new lfoda_x_y(StrToFloat(Edit_y->Text),*if_xn);
```

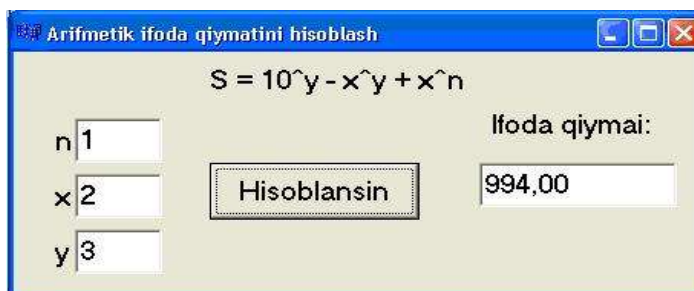


```

if_10y=new lfoda_10_y(*if_xy);
Edit_S->Text=FloatToStrF(if_xn->Hisobla()-if_xy->Hisobla()+
if_10y->Hisobla(),2,6,2);
delete if_xn; delete if_xy; delete if_10y;
}

```

Ilovaning ishlash jarayonidagi ko‘rinishi



### *Amaliy topshiriqlar*

1. 10 lik sanoq sistemasida berilgan ishorasiz haqiqiy sonni ko‘rsatilgan “p.m” formatda chop qiladigan FLOAT\_PRINT sinf yaratilsin. Bu yerda p-son ko‘rinishidagi jami joylar (pozitsiya), m-sonning kasr qismidagi raqamlar soni. Natijani 2,8 va 16 lik sanoq sistemasidagi ko‘rinishini chop qiluvchi FLOAT\_PRINT2, FLOAT\_PRINT8, FLOAT\_PRINT16 hosilaviy sinflar yaratilsin.
2. Kompleks sonlar ustida arifmetik amallar bajaradigan KOMPLEKS tayanch sinfi yaratilsin. Undan voris sinf sifatida kompleks koeffisientli kvadrat tenglama ildizini topadigan KOMP\_KV\_TENGLAMA sinfi yaratilsin.
3. Berilgan natural n o‘lchamidagi  $a_{ij} (1 \leq i, j \leq n)$  haqiqiy elementli kvadrat matritsa uchun xotiradan joy ajratich, qiymatlarini o‘qish va chop qilish amallarini bajaradigan MATRITSA tayanch sinfi yaratilsin. Uning vorisi qilib matritsaning determinantini minorlar usulida hisoblovchi funksiya-a’zosi bo‘lgan DTR\_MATRITSA sinfi yaratilsin. Matritsa determentanti quyidagicha hisoblanadi:

$$\Delta_n = \sum_{k=1}^n a_{1k} A_{1k}.$$

Bu yerda  $A_{1k} = (-1)^{1+k} M_{1k}$ ,  $M_{1k}$  - minor bo‘lib, u  $a_{ij}$  matritsaning 1-satri va k-ustunini o‘chirishdan hosil bo‘ladigan (n-1) - tartibli matritsaning determenanti.

4. n o‘lchamli fazoda koordinatalari bilan berilgan ikkita vektorni tavsiflovchi, ya’ni ularni xotirada saqlash, qiymatlarini o‘qish va chop qilish amalini bajaruvchi VEKTOR\_XY tayanch sinfi aniqlansin. Uning vorisi

bo'lgan VEKTOR\_XY\_AMAL sinfida vektorlarni qo'shish va ayirish orqali yangi vektorlar hosil qiluvchi, ikkita vektorning skalyar ko'paytmasini, vektor uzunligini va ikkita vektorlar orasidagi burchak kosinusi hisoblovchi funksiya-a'zolar aniqlansin.

5. Ko'phad darajasi va koeffisientlari bilan berilgan bitta o'zga'uvchili ko'phadni xotirada saqlash, qiymat o'qish va chop qilish amalini bajaruvchi KOP\_HAD tayanch sinfi yaratilsin. Ushbu sinf vorisi sifatida berilgan butun k soni uchun k-tartibli Chebishev ko'phadi koeffisientlarini hisoblaydigan va berilgan haqiqiy turdagi argumenti uchun ko'phad qiymatini hisoblovchi SHEBISHEV sinfi yaratilsin (12.10-masalaga qaral-sin).
6. Kitobning nomi, muallifi, nashriyoti nomi va chop qilingan yili bo'yicha berilganlarni xotirada saqlash, qiymat o'qish va chop qilish amalini bajaruvchi KITOB tayanch sinfi yaratilsin. Uning vorisi bo'lgan UY\_KUTUBXONASI sinfida - uy manzili, kutubxona egasi familiyasi, ismi haqida ma'lumotlar bo'lsin.  
UY\_KUTUBXONASI sinfi obyektlarining chiziqli zanjirini yaratgan holda unda ixtiyoriy sondagi kitoblar bilan ishlash, qandaydir alomati bo'yicha kitobni izlash (muallif, yil yoki uy manzil bo'yicha va hokazo), yangi kitobni qo'shish va o'chirish amalga oshirilsin.
7. Berilgan satrni saqlash va chop qilish amallarini o'z ichiga olgan SATR sinfi aniqlansin. SATR sinfining vorisi sifatida ARIFM\_AMAL sinfi yaratilsinki, unda satr ko'rinishida berilgan ikkita son o'rtasida arifmetik amallar ('+', '-', '\*', '/') bajarilsin.
8. Shaxsning familiyasi va ismi, tug'ilgan yili, jinsi, yashash manzili va telefon nomeri bo'yicha ma'lumotni xotirada saqlash, qiymat o'qish va chop qilish amalini bajaruvchi SHAXS sinfi yaratilsin. Uning vorisi qilib talabalar guruhini tavsivlovchi TALABA sinfi yaratilsin. Unda qo'shimcha ravishda talabaning o'qiydigan guruh nomi, kursi haqida ma'lumot bo'lishi kerak. TALABA sinfi obektlari chiziqli ro'yxatini yaratgan holda, qandaydir alomat bo'yicha talabani izlash (familiya, tug'ilgan yili yoki telefon nomeri bo'yicha), ro'yxatga yangi yozuvni qo'shish, o'chirish va tartiblash amallari bajarilsin.
9. Natural sonlarning ikkita to'plamini yaratish, chop qilish amallarini o'z ichiga olgan TUPLAM\_AB sinfi yaratilsin. To'plam ustida amallarni – to'plamga yangi element qo'shish va o'chirish, tuplamlar keshishmasini, birlashmasini, hamda ayirmasini bajaradigan funksiya-a'zolari bo'lgan TUPLAM\_AMALLARI sinfi TUPLAM sinfi vorisi qilib aniqlansin.
10. Berilgan satrni oqimdan o'qish, saqlash, chop qilish amallarini bajara-digan MATN sinfi yaratilsin. Uning vorisi sifatida faqat lotin harfida

yoziqlan matnni shifrlaydigan va qayta tiklaydigan SHIFRLASH sinfi aniqlansin. Shifrlash uchun lotin harflar alfaviti olinadi. Jarayon chapdan o'ng tomonga ketma-ket ravishda matndagi har bir harf bo'yicha amalga oshiriladi. Har qadamda alfavitni ko'rsatilgan songa siklik chapga suriladi va matndagi ayni harfni uning hosil bo'lgan alfavitdagi o'rnidagi (indexidagi) harf bilan almashtiriladi. Har bir qadam uchun alfavitni surish soni beriladi. Masalan, 5,3,2,4 sonlari berilgan bo'lsin. Birinchi qadamda, y'ani matnning birinchi harfini (faraz qilaylik 'A' harfi) kodlashda alfavit 5 marta chapga siklik suriladi va hosil bo'lgan alfavitdagi qaralayotgan harf o'rnida paydo bo'lgan harf bilan ('F') almashtiriladi. Ikkinchi qadamda alfavit yana 3 marta chapga suriladi va hokazo. To'rtinchi qadamdan keyin, ya'ni alfavit 4 marta chapga surilgandan keyin, surilshlar ro'xati takrorlanadi.

11. Stekni amalga oshiruvchi STEK sinfi aniqlansin. Unda stekni tozalash, unga qiymat joylashtirish, o'chirish amallari bajarilsin. Ushbu sinfnin vorisi bo'lgan LABIRINT sinfidan labirintdan chiqish masalasini yechishda foydalanilsin. Labirint kvadratlardan tashkil topgan matritsa ko'rinishida beriladi. Har bir kvadrat ochiq yoki yopiq bo'ladi. Yopiq kvadratga kirish mumkin emas. Agar kvadrat ochiq bo'lsa uning yon tomonidan kirish mumkin (burchagidan kirish mumkin emas). Har bir kvadrat uning matritsada koordinatalari bilan beriladi. Labirintdan chiqish amalga oshirilganda topilgan yo'l chop qilinadi (kvadratlar koordinatalari juftliklarining ketma-ketligi ko'rinishida).
12. TAXTA sinfi shaxmat taxtasini tavsivlaydi. Shaxmat katagi ikkita belgidan tashkil topgan katak ko'rinishida berilgan: lotin harfi (a dan h gacha) va raqam (1 dan 8 gacha), masalan a2 yoki g5. Ularni farzin joylashgan shaxmat taxtasidagi katak koordinatalari sifatida qarab, farzin "uradigan" kataklarni 'X', boshqa kataklarni '0' bilan belgilab, shaxmat taxtasining ko'rinishi chop qilish imkonini beruvchi voris FARZIN sinfi aniqlansin.
13. Ko'rsatkich asosida yaratilgan butun sonlardan iborat navbatni tavsiflovchi NAVBAT sinfida navbat bilan ishlash, ya'ni elementlar oxiriga qo'shish, boshidan o'chirish ("*birinchi kelgan–birinchi ketadi*") bilan bog'liq quyidagi funksiyalar aniqlanishi zarur bo'ladi:
  - Tozalash() - bo'sh navbatni yaratuvchi (navbatni tozalovchi);
  - BushNavbat() - navbatni bo'shligini tekshiruvchi;
  - Navbatga() - navbat oxiriga yangi element qo'shuvchi;
  - Navbatdan() - navbatdagi birinchi elementni qaytaruvchi va uni navbatdan o'chiruvchi.

NAVBAT sinfinin vorisi sifatida sonlarning umumiy navbatidan sonlarni

musbat sonlar navbatiga va musbat bo‘lmagan sonlar navbatiga ajratuvchi MUSBAT\_MANFIY\_NAVBATLAR sinfi aniqlansin.

14. Tasodifiy son hosil qiluvchisidan shakllarning yuzasini va hajmini hisoblashda qo‘llash mumkin. Shunday usullardan biriga Monte-Karlo usuli deyiladi (MONTE\_KARLO sinfi) va uning mohiyati quyidagicha: faraz qilaylik, M shakl birlik kvadrat ichida to‘laligicha yotibdi. Tasodifiy son hosil qiluvchisi yordamida birlik kvadrat ichida n ta nuqta tanlandi, agar  $v(n)$  orqali bu nuqtalarning M shakl ichiga tushganlari miqdori bo‘lsin. U holda M shakl yuzasi taqriban  $\frac{v(n)}{n}$  qiymatiga teng bo‘ladi va n qanchalik ko‘p bo‘lsa yuzaning haqiqiy qiymatiga yaqinlashiladi. Tasodiffiy tanlangan nuqta sifatida  $(r_1, r_2), (r_3, r_4), \dots$  koordinatalari bilan berilgan nuqtalarni olish mumkin, bu yerda  $r_1, r_2, \dots$  tassodifiy son hosil qiluvchisi tomonidan olingan sonlar. Xuddi shunday, uch o‘lchamli fazodagi nuqtalarni  $(r_1, r_2, r_3)$  koordinatalari bilan tanlash orqali birlik kub ichidagi shakl hajmini hisoblash mumkin.

MONTE\_KARLO sinfining vorislari sifatida analitik ko‘rinishi bilan berilgan tekislik figurasi yuzasini hisoblash uchun MONTE\_KARLO\_2D, uch o‘lchamli fazoda figura hajmini hisoblash uchun MONTE\_KARLO\_3D sinflari aniqlansin.

15. Bitta qurilmadan ikkinchisiga kanal orqali 0 va 1 raqamlaridan iborat xabar jo‘natayotganda halal beruvchi shovqinlar ta’sirida xabar xato qabul qilinishi mumkin (0 o‘rniga 1 yoki 1 o‘rniga 0). Bunday xatolikni bartaraf qilish yo‘llaridan biri - har bir uzatiladigan raqmlarni uch marta takrorlashdir. Masalan, 1,0,1 xabari 1,1,1,0,0,0,1,1,1 ko‘rinishida uzatiladi. Qabul qilishda esa har bir uchta raqamlar guruhi unda eng ko‘p uchragan raqam bilan almashtirish orqali xabar tiklanadi.

Yuqorida keltirilgan usul bilan berilgan matnni (satrni) “*junatadigan*“ va “*qabul*“ qiladigan amallarni bajaruvchi XABAR tayanch sinfi yaratilsin. Uning vorisi qilib berilgan matnni yuqorida keltirilgan usulda shifrlash orqali jo‘natidigan va qabul qiladigan XABARNI\_SHIFRLASH voris sinfi yaratilsin. Bu yerda matn belgilarining ASCII kodida asosida jo‘natiladi, qabul qilinadi va tiklanadi.

16. Haqiqiy son kompyuter xotirasida ko‘rinishidagi formatda saqlanadi.

S	P	M
---	---	---

Bu yerda S-son ishorasini aniqlaydi. Agar son musbat bo‘lsa  $S=0$ , aks holda  $S=1$  bo‘ladi. P-son tartibi (Q-sanoq sistemasi asosining darajasi). M-mantissa ( $0 < M < 1$ ). Har qanday son  $(-1)^S M * Q^P$  ko‘rinishiga keltirilib saqlanadi.

Oldindan berilgan format o'ldamlari - M, P qiymatlariga ko'ra berilgan N sonining ichki formatini tavsiflovchi FLOAT\_FORMAT sinfi aniqlansin. Sonning ichki formatiga mos baytlardagi sonlarni mos ravishda 2,10, va 16 lik sanoq sistemasida chop qiluvchi FFORMAT2, FFORMAT10 va FFORMAT16 hosilaviy sinflar yartilsin.

17. Tayanch UCHBURCHAK sinfi aniqlangan, bunda uchburchak uchlarining koordinatalari bilan aniqlangan. Tayanch sinfda Turi()- uchburchak turini aniqlovchi, Yuza() - uchburchak yuzasini hisoblovchi, Perimetr() - uchburchak perimetrni hisoblovchi funksiya-a'zolari bo'lsin.

Quyidagilar amalga oshirilsin:

- a) T\_UCHBURCHAK nomli to'g'ri burchakli uchburchak xususiyatlarini o'z ichiga olgan voris sinf qurilsin va uning funksiya-azolari aniqlansin;
- b) TT\_UCHBURCHAK nomli teng tomonli uchburchak xususiyatlarini o'z ichiga olgan voris sinf qurilsin va uning funksiya-a'zolari aniqlansin;
- d) TY\_UCHBURCHAK nomli teng yonli uchburchak xususiyatlarini o'z ichiga olgan voris sinf qurilsin va uning funksiya-a'zolari aniqlansin.

18. Tayanch TO\_RTBURCHAK sinfida to'rtburchak uchlari  $A(x_1, y_1)$ ,  $B(x_2, y_2)$ ,  $C(x_3, y_3)$  va  $D(x_4, y_4)$  koordinatalari bilan aniqlangan. Tayanch sinfda Mavjud() – to'rtburchak mavjudligini aniqlovchi, Yuza() – to'rtburchak yuzani hisoblovchi, Perimetr() – to'rtburchak perimetrini hisoblovchi funksiya-a'zolari aniqlansin. Quyidagi masallar yechilsin:

- a) ROMB nomli romb xususiyatlarini o'z ichiga olgan voris sinf qurilsin va uning funksiya-a'zolari aniqlansin;
- b) PARALLELOGRAM nomli parallelogram xususiyatlarini o'z ichiga olgan voris sinf qurilsin va uning funksiya-a'zolari aniqlansin;
- d) KVADRAT nomli kvadrat xususiyatlarini o'z ichiga olgan voris sinf qurilsin va uning funksiya-a'zolari aniqlansin.

19. Matnni o'qish, saqlash va chop qilish amallairini o'z ichiga olgan MATN sinfi aniqlansin. Uning vorisi qilib berilgan matnni formula ekanligini aniqlaydigan FORMULA sinfi yaratilsin. Matnni “formula” ekanligini quyidagi grammatik qoidalar aniqlaydi:

$\langle \text{formula} \rangle ::= \langle \text{term} \rangle | (\langle \text{formula} \rangle \langle \text{amal} \rangle \langle \text{formula} \rangle)$

$\langle \text{amal} \rangle ::= + | - | *$

$\langle \text{term} \rangle ::= \langle \text{nom} \rangle | \langle \text{butun} \rangle$

$\langle \text{nom} \rangle ::= \langle \text{harf} \rangle | \langle \text{nom} \rangle \langle \text{harf} \rangle | \langle \text{nom} \rangle \langle \text{raqam} \rangle$

$\langle \text{butun} \rangle ::= \langle \text{raqam} \rangle | \langle \text{butun} \rangle \langle \text{raqam} \rangle$

$\langle \text{harf} \rangle ::= a | b | c | d | e | f$

$\langle \text{raqam} \rangle ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9$

Nuqta bilan tugaydigan matnning “formula” yoki yo'qligi aniqlansin.

20. Berilgan sonlar ustida arifmetik amallarni ('+', '-', '\*', '/') bajaruvchi

KALK\_ARIFMETIKA sinfi aniqlansin. Uning vorisi qilib qo‘shimcha ravishda  $\ln()$ ,  $x^y$ ,  $\sin()$ ,  $\cos()$ ,  $\text{tg}()$ ,  $\text{ctg}()$  va  $\text{sqrt}()$  funksiyalarini o‘z ichiga olgan KALK\_MUHANDIS sinfi aniqlansin.

## 24. Operatorlarni qayta yuklash

### *Namunaviy masala*

Koeffisientlari va ozod hadi bilan berilgan  $AX = B$  chiziqli tenglamalar sistemasi Gauss usulida yechilsin.

### *Yechish usuli*

Chiziqli tenglamalar sistemasini yechishning Gauss usuli haqidagi ma’lumot bilan hisoblash matematikasi bo‘yicha adabiyotda tanisish mumkin. Masalani yechishda Gauss usulining ustun bo‘yicha maksimal elementni tanlash bilan bog‘liq varianti ishlatiladi.

Qo‘yilgan masalani yechish uchun quyidagi sinflar yaratildi:

Vektor sinfi - bu berilgan  $n$  o‘lchamdagi chiziqli tenglamalar sistemasi  $X$  va  $B$  vektorlarini saqlash, yaratish (oddiy va nusxalash konstruktorlari orqali), vektor elementiga murojaat qilish ( $[]$  amalini qayta yuklash orqali), vektorni chop qilish ( $\gg$  amalini qayta yuklash orqali) va dinamik massivni o‘chirish funksiyalarini o‘z ichiga olgan sinf;

Matritsa sinfi - berilgan  $n*n$  o‘lchamdagi kvadrat matritsani saqlash (konstruktor vositasida) va matritsa o‘lchamini qaytaruvchi funksiyalariga ega sinf;

Gauss sinfi - bu Matritsa sinfi vorisi bo‘lib, unga Vektor sinfi agregatsiya qilingan, y’ani sinf tarkibiga Vektor turidagi ob’ektlar ( $X$  va  $B$  massivlar) kiritilgan. Bunda tashqari sinf tarkibiga konstruktor orqali chiziqli tenglamalar sistemasining o‘lchamini, matritsa qiymatlarini ( $A$  matritsa) va ozod hadlar ( $B$  vektorni) kiritish, hamda Gauss usulini amalga oshiruvchi  $\text{Metod}()$  funksiya-a’zo, tenglamalar sistemasi yechimini chop qiluvchi ( $\gg$  amalini qayta yuklash orqali) va olingan yechimni tenglamalar sistemasini qanoatlantirishini tekshirish uchun matritsaning boshlangich holatini tiklash uchun ( $\ll$  amalini qayta yuklash orqali) operator-funksiyalar kiradi.

Matritsa va Vektor sinfida qiymatlarni o‘qish va chop qilishda funksiyalar parametri sifatida uzatiladigan vizual komponentalardan foydalanilgan.

Sinflar e’lonlari  $\langle \text{gauss.h} \rangle$  sarlavha fayliga yoziladi.  $\text{Gauss.cpp}$  modulida Gauss turidagi ko‘rsatkich bo‘yicha obyekt yaratiladi va unga  $\text{TStringGrid}$  obyektlaridan qiymatlar uzatiladi.

Ilova bilan ishlash qadamlari:

1. “Matritsa o‘lchami N:” nishoni bilan belgilangan bir qatorli tahrir maydoniga sistema o‘lchami – n soni kiritiladi va “cin>>N” tugmasi bosiladi.

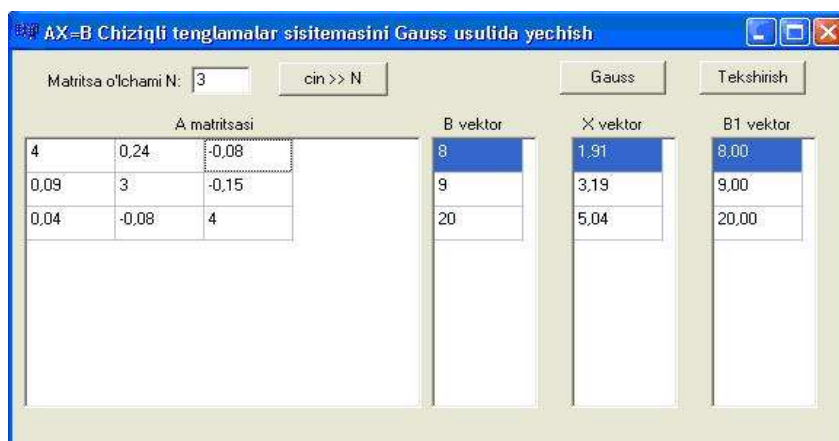
2. Kiritilgan n soniga mos ravishda TStringGrid turidagi “A matritsa” hamda “B vektor” o‘lchamlari shakllanadi va ularning kataklariga qiymatlar kiritiladi.

3. Sistema koeffisientlari kiritilgandan keyin “Gauss” tugmasi bosiladi. Agar sistema yechimga ega bo‘lsa u “X vektor” bilan belgilangan kataklarga chop etiladi, aks holda yechim yo‘qligi haqida xabar beriladi.

4. Olingan X yechim tenglamalar sistemasini qanoatlantirishini tekshirish uchun “Tekshirish” tugmasi bosiladi va ozod hadlar - B vektor qiymatlari “B1 vektor” kataklariga chop etiladi.

Shuni qayd etish kerakki, ilova da qiymatlar kiritishlarini nazorat qilish va tenglamalar sistemasi qiymatlarini qaytadan kiritish holatlari inobatga olinmagan.

Ilova shaklining (formaning) ko‘rinishi quyidagi rasmda keltirilgan.



### *Ilova matni*

#### *Unit\_Gauss.h fayli:*

```
enum TStringdoubleFormat{sffGeneral, sffExponent, sffFixed, sffNumber, sffCur-  
rency };
```

```
class Matritsa
```

```
{
```

```
int na;
```

```
protected:
```

```
double **A;
```

```
public:
```

```
Matritsa(int _na, TStringGrid * SG);
```

```

~Matritsa();
int n_matritsa(){return na;}
};
Matritsa::Matritsa(int _na, TStringGrid * SG)
{
na=_na;
A=new double*[na];
for(int i=0;i<na;i++) A[i]=new double[na];
for(int i=0;i<na;i++)
for(int j=0;j<na;j++)
A[i][j]=StrToFloat(SG->Cells[j][i]);
}
Matritsa::~Matritsa()
{
for(int i=0;i<na;i++) delete [] A[i];
delete A;
}
class Vektor
{
int nv;
double *v;
public:
Vektor(int,TStringGrid*);
Vektor(Vektor*);
~Vektor(){delete [] v;}
void operator>>(TStringGrid*);
int n_vektor(){return nv;}
double & operator[](int i){return v[i];}
};
Vektor::Vektor(int _nv,TStringGrid * SG)
{
nv=_nv;
v=new double[nv];
for(int i=0;i<nv;i++) v[i]=StrToFloat(SG->Cells[0][i]);
}
Vektor::Vektor(Vektor * V)
{
if(V)
{
nv=V->n_vektor(); v=new double[nv];
}
}

```



```

    for(int i=0;i<nv;i++)v[i]=V->operator [](i);
}
}
void Vektor::operator>>(TStringGrid *SG_v)
{
for(int i=0;i<nv;i++) SG_v->Cells[0][i]=FloatToStrF(v[i],sffFixed,6,2);
}
class Gauss:public Matritsa
{
Vektor *B,*X;
public:
Gauss(int n,TStringGrid* SG_A,TStringGrid* SG_B): Matritsa(n,SG_A)
{
B=new Vektor(n,SG_B);}
~Gauss(){ delete [] B; delete [] X;}
bool Metod();
Vektor * AX_B();
void operator >>(TStringGrid*);
void operator <<(TStringGrid*);
Vektor * Vektor_B(){return B;}
};
bool Gauss::Metod()
{
double r;
int k;
int N=n_matritsa();
X=new Vektor(B);
for(int i=0;i<N;i++)
{
k=i;
r=fabs(A[i][i]);
for(int j=i+1;j<N;j++)
if(fabs(A[j][i])>r)
{
k=j;
r=fabs(A[j][i]);
}
}
if(r==0.0) return 0;
if(k!=i)
{

```

```

r=X->operator [](k);
X->operator [](k)=X->operator [](i);
X->operator [](i)=r;
For (int j=i;j<N;j++)
{
    r=A[k][j];
    A[k][j]=A[i][j];
    A[i][j]=r;
}
}
r=A[i][i];
(*X)[i]=(*X)[i]/r;
for(int j=i;j<N;j++)
A[i][j]=A[i][j]/r;
for(int k=i+1;k<N;k++)
{
    r=A[k][i];
    (*X)[k]=(*X)[k]-r*(X)[i];
for(int j=i;j<N;j++) A[k][j]=A[k][j]-r*A[i][j];
}
}
for(int i=N-2;i>=0;i--) for(int j=i+1;j<N;j++) (*X)[i]=(*X)[i]-A[i][j]*(X)[j];
return 1;
}
Vektor* Gauss::AX_B()
{
for(int i=0;i<n_matritsa();i++)
{
(*B)[i]=0;
for(int j=0;j<n_matritsa();j++) (*B)[i]+=A[i][j]*(X)[j];
}
return B;
}
void Gauss::operator >>(TStringGrid * Sg_X)
{
for(int i=0;i<X->n_vektor();i++)
Sg_X->Cells[0][i]=FloatToStrF((*X)[i],sffFixed,6,2);
}
void Gauss::operator <<(TStringGrid * SG_A)
{

```

```

for(int i=0;i<n_matritsa());i++)
  for(int j=0;j<n_matritsa();j++) A[i][j]=StrToFloat(SG_A->Cells[j][i]);
}
Gauss *gauss;
Unit_Gauss.cpp fayli:
...
int n;
void __fastcall TForm1::Btn_UlchamClick(TObject *Sender)
{
  n=StrToInt(Edit_n->Text);
  StringGrid_A->RowCount=n; StringGrid_A->ColCount=n;
  StringGrid_B->RowCount=n; StringGrid_B->ColCount=1;
  StringGrid_X->RowCount=n; StringGrid_X->ColCount=1;
  StringGrid_BT->RowCount=n; StringGrid_BT->ColCount=1;
}
void __fastcall TForm1::Btn_GaussClick(TObject *Sender)
{
  gauss=new Gauss(n,StringGrid_A,StringGrid_B);
  if(gauss->Metod())(*gauss)>>StringGrid_X;
  else ShowMessage("Tenglamalar sistemasi echimga ega emas.");
}
void __fastcall TForm1::Btn_TestClick(TObject *Sender)
{
  *gauss<<StringGrid_A;
  *gauss->AX_B()>>StringGrid_BTest;
}

```

### ***Amaliy topshiriqlar***

1. Kompleks sonlar ustida arifmetik amallar operator-funksiya qilib aniqlangan KOMPLEKS sinfi yaratilib, uning obyektlari ustida amallar bajarilsin.
2. Berilgan  $n$  o'lchamli vektor ustida vektorlarni qo'shish, ayirish, skalyar ko'paytirish, hamda vektorni songa ko'pytirish amallari qayta yuklangan VEKTOR sinfi aniqlansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
3. Berilgan natural  $n$  va  $m$  o'lchamdagi haqiqiy elementli matritsa uchun xotiradan joy ajratich, qiymatlarini o'qish va chop qilish amallarini bajaradigan MATRITSA tayanch sinfi yaratilsin. Berilgan  $A$  va  $B$  matritsalar ustida  $A+B$ ,  $A-B$ ,  $A*B$ , amallarini bajaradigan ARIFM\_MATRITSA sinfi MATRITSA sinfidan voris sifatida yaratilsin va unda ko'rsatilgan amallar qayta yuklansin.

4. Vektor yordamida natural sonlar to'plamini hosil qilish amalini bajaruvchi TUPLAM sinfi yaratilsin. To'plam ustida asosiy amallarni - to'plamga yangi element qo'shish va o'chirish, to'plamlar keshishmasi, birlashmasi, hamda ayirmasi amallari qayta yuklansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
5. Vaqtning sekunt, minut, soat qiymatlari ustida bajariladigan qo'shish, ayirish va taqqoslash amallari qayta yaklaydigan VAQT sinfi aniqlansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
6. Sananing kun, oy, yil qiymatlari ustida bajariladigan qo'shish, ayirish va taqqoslash amallari qayta yaklaydigan SANA sinfi aniqlansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
7. Rasional sonlar ustida, yani surat va mahraj juftligi bilan berilgan sonlar ustida qo'shish, ayirish, kopaytirish taqqoslash amallarini qayta yaklaydigan RATSIONAL sinfi aniqlansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
8. Tekislikda dekart koordinatasida berilgan nuqta koordinatasini qutb koordinatasiga va aksincha, qutb koordinatasidan dekart koordinatasiga otkazuvchi amallarni o'z ichiga olgan QUTB va DEKART sinflari aniqlansin. Sinflar obyektlari ustida ko'rsatilgan amallar bajarilsin.
9. AKSLANTIRISH\_01 sinfi aniqlansin. Unda haqiqiy sonlar massivini [0,1] segmentga akslantirish operator - funksiya ko'rinishida aniqlansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
10. SATR sinfi aniqlansin va unda nol terminatorli satrlar ustida satrga satr qo'shish, satrdagi bir satr ostini ikkinchi satr bilan almashtirish amallari operator-funksiya ko'rinishida aniqlansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
11. STEK sinfi aniqlansin. Unda stekga element joylash va o'qish, stek bo'shligini tekshirish amallari operator-funksiya sifatida aniqlansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
12. Haqiqiy sonlarning  $[a,b]$  va  $[c,d]$  yopiq intervallari ustida quyidagi amallar aniqlangan:  
 $[a,b]+[c,d]=[a+c,b+d];$   
 $[a,b]-[c,d]=[a-c,b-d];$   
 $[a,b]*[c,d]=[min(a*c,a*d,b*c,b*d), max(a*c,a*d,b*c,b*d)];$   
 $[a,b]/[c,d]=[a,b]*[1/c,1/d]$   $c>0$  yoki  $d<0$  shartida;  
 $[a,b]=[c,d]$ , agar  $a=c$  va  $b=d$ ;  
 $[a,b]<=[c,d]$  agar  $a<=c$  va  $b<=d$ .  
 INTERVAL sinfi aniqlanib, unda yuqoridagi amallari qayta yuklansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
13. Chekli  $n$  o'lchamli mantiqda qiymatlar  $0$  dan  $n-1$  gacha sonlar bo'lib,

undagi amallar quyidagicha aniqlanadi:

$$a * b = \min(a, b); \quad a + b = \max(a, b); \quad a - b = a - 1 - b.$$

N\_MANTIQ sinfi aniqlansin. Unda yuqoridagi amallar qayta yuklansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.

14. Ishoralar mantig'ida amallar quyidagicha aniqlanadi:  
 $a * b = \text{sign}(a) * \text{sign}(b); \quad a + b = \text{sign}(a) + \text{sign}(b); \quad a - b = \text{sign}(a) - \text{sign}(b).$   
ISHORA\_MANTIQ sinfi aniqlanib, unda yuqoridagi amallar qayta yuklansin. Sinf obyektlari ustida ko'rsatilgan amallar bajarilsin.
15. Berilgan  $n$ -darajali  $P(x)$  ko'phadni tavsiflovchi KUPHAD sinfi yaratilsin. Unda ikkita ko'phadni qo'shish, ko'pytirish, ayirish amallari, hamda berilgan  $a$  va  $m$  uchun  $P(a)$  qiymatini  $m$  soni bilan taqqoslash amallari operator-funksiyalar ko'rinishida aniqlansin.
16. Katta sonlar ustida arifmetik amallarni bajarish. O'nlik sanoq sistemasidagi ikkita  $a$  va  $b$  butun sonlar satr ko'rinishida berilgan.  $a + b$ ,  $a - b$ ,  $a * b$  va  $a / b$  amallari qayta yuklanuvchi operator ko'rinishida aniqlangan UZUN\_SON sinfi aniqlansin.
17. Uzunligi oldindan noma'lum bo'lgan binar  $a$  va  $b$  sonlar ustida arifmetik amallar bajarilsin. Sonlar satr ko'rinishida berilgan.  $a + b$ ,  $a - b$ ,  $a >> n$  ( $a$  razryadlarini o'ngga  $n$  pozitsiyaga surish),  $a << n$  ( $a$  razryadlarini o'ngga  $n$  pozitsiyaga surish) va  $a^b$  (istisnoli yoki) amallari qayta yuklanuvchi operator ko'rinishida aniqlangan BINAR\_SON sinfi aniqlansin.
18. Uzunlik kilometr, metr, santimetr va millimetr birliklar majmuasi ko'rinishida beriladi. UZUNLIK sinfi aniqlansin va unda uzunliklarni qo'shish va ayirishni amalga oshiradigan, hamda ikkita uzunlikni o'zaro taqqoslaydigan operator-funksiyalar aniqlansin.

## 25. Polimorfizm

Ushbu bo'limda qo'yilgan masala mazmunidan kelib chiqqan holda, uning umumiy xususiyatlarini o'z ichiga olgan tayanch sinf yaratiladi, undan voris sifatida hosil bo'lgan sinf orqali bevosita qo'yilgan masala yechiladi.

### *Namunaviy masala*

Shaxsning o'rta ma'lumot yoki oliy ma'lumot darajasiga mos ravishda anketa ma'lumoti to'ldirilsin.

### *Yechish usuli*

Shaxsning o'rta yoki oliy ma'lumotga ega ekanligiga mos ravishda uning anketa punktlari turlicha bo'ladi. Masalan, o'rta ma'lumotli shaxsning anketasiga quyidagi maydonlar kirishi mumkin: ismi, familiyasi,

maktab (litsey/kollej) tartib raqami va maktabni (litsey/kollejni) tugatgan yili. O‘z navbatida oliy ma‘lumotli shaxs uchun esa bu maydonlar quyidagicha bo‘lishi mumkin: ismi, familiyasi, OTM nomi, fakultet nomi, OTMni tugatgan yili va mutaxassisligi.

Qo‘yilgan masala ikkita o‘zaro vorislik munosabatida bo‘lgan sinflarni yaratish bilan yechiladi:

– tayanch UMUMIY\_TA\_LIM sinfida o‘rta ma‘lumotga ega shaxs uchun anketani shakllantirish, kiritilgan ma‘lumotni qabul qilish va chop qilish amallari bajariladi. Ma‘lumotni qabul qilish va chop qilish formaning TStringGrid va TMemo turidagi komponentalari vositasida amalga oshiriladi (mos funksiya-a‘zolarining parametrida ko‘rsatilgan).

– voris OLIY\_TA\_LIM sinfida oliy ma‘lumotli shaxsning anketasi bilan bog‘liq o‘ziga xos amallar aniqlanadi.

Anketaning ko‘rinishini mos ravishda shakllantirish va chop qilish uchun polimorfizmdan foydalaniladi. Bu vazifani bajaruvchi funksiyalar virtual aniqlanadi:

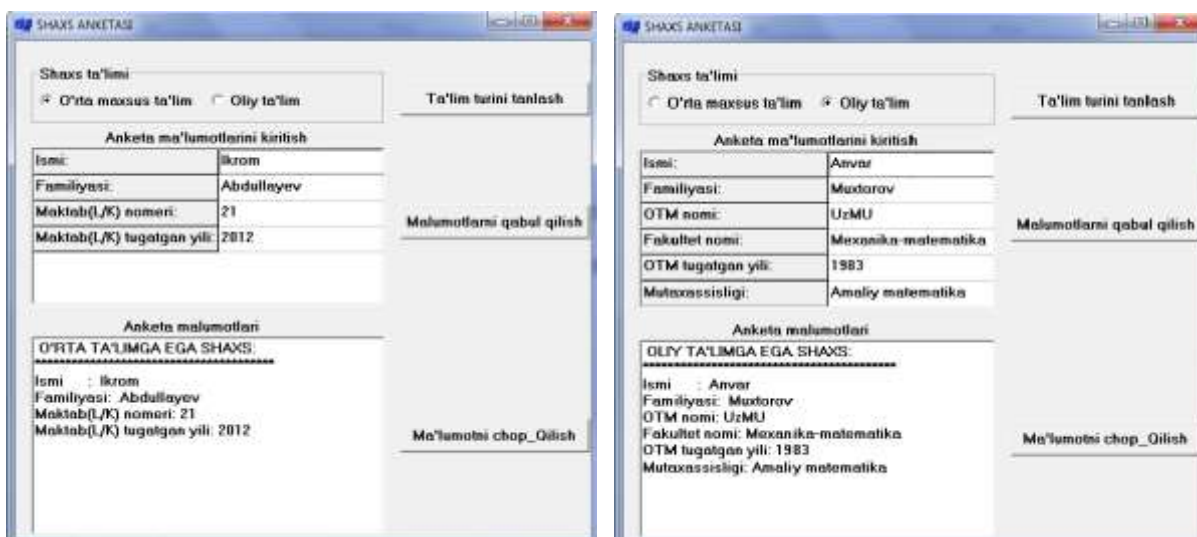
virtual void Anketa\_shakli(TStringGrid\*);

virtual void Chop\_etish(TMemo\*);

Ilova ishlashida quyidagi ketma-ketlikka amal qilinadi:

1. Shaxsning o‘rta yoki oliy ma‘lumotga ega ekanligi aniqlanadi;
2. Ta‘lim turiga mos anketa shakllantiriladi;
3. Anketa ma‘lumotlari qabul qilinadi va chop etiladi.

Ilovaning ish holatlari quyidagi rasmda keltirilgan



## *Hova matni*

### *Unit\_Anketa.h fayli:*

```
...
class UMUMIY_TA_LIM
{
public:
    UMUMIY_TA_LIM();
    ~UMUMIY_TA_LIM();
    void Anketa_qabul(TStringGrid*);
    virtual void Anketa_shakli(TStringGrid*);
    virtual void Chop_etish(TMemo*);
protected:
    int n;
    String * shaxs;
};
UMUMIY_TA_LIM::UMUMIY_TA_LIM()
{ n=4; shaxs=new String[n]; }
UMUMIY_TA_LIM::~~UMUMIY_TA_LIM()
{ delete []shaxs; }
void UMUMIY_TA_LIM::Anketa_shakli(TStringGrid * SG)
{
    SG->RowCount=n;
    SG->Cells[0][0]="Ismi:";
    SG->Cells[0][1]="Familiyasi:";
    SG->Cells[0][2]="Maktab(L/K) nomeri:";
    SG->Cells[0][3]="Maktab(L/K) tugatgan yili:";
}
void UMUMIY_TA_LIM::Anketa_qabul(TStringGrid * SG)
{ for(int i=0; i<n;i++) shaxs[i]=SG->Cells[1][i]; }
void UMUMIY_TA_LIM::Chop_etish(TMemo *memo)
{
    memo->Lines->Clear();
    memo->Lines->Add(" O'RTA TA'LIMGA EGA SHAXS: ");
    memo->Lines->Add("*****");
    memo->Lines->Add("Ismi : "+shaxs[0]);
    memo->Lines->Add("Familiyasi: "+shaxs[1]);
    memo->Lines->Add("Maktab(L/K) nomeri: "+shaxs[2]);
    memo->Lines->Add("Maktab(L/K) tugatgan yili: "+shaxs[3]);
}
class OLIY_TA_LIM :public UMUMIY_TA_LIM
```

```

{
public:
OLiy_TA_LIM();
~OLiy_TA_LIM();
void Anketa_shakli(TStringGrid*);
void Chop_etish(TMemo*);
};
OLiy_TA_LIM::OLiy_TA_LIM() { n=6; shaxs=new String[n]; }
OLiy_TA_LIM::~~OLiy_TA_LIM() {delete [] shaxs; }
void OLiy_TA_LIM::Anketa_shakli(TStringGrid * SG)
{
SG->RowCount=n;
SG->Cells[0][0]="Ismi:";
SG->Cells[0][1]="Familiyasi:";
SG->Cells[0][2]="OTM nomi:";
SG->Cells[0][3]="Fakultet nomi:";
SG->Cells[0][4]="OTM tugatgan yili:";
SG->Cells[0][5]="Mutaxassisligi:";
}
void OLiy_TA_LIM::Chop_etish(TMemo *memo)
{
memo->Lines->Clear();
memo->Lines->Add(" OLIY TA'LIMGA EGA SHAXS: ");
memo->Lines->Add("*****");
memo->Lines->Add("Ismi : "+shaxs[0]);
memo->Lines->Add("Familiyasi: "+shaxs[1]);
memo->Lines->Add("OTM nomi: "+shaxs[2]);
memo->Lines->Add("Fakultet nomi: "+shaxs[3]);
memo->Lines->Add("OTM tugatgan yili: "+shaxs[4]);
memo->Lines->Add("Mutaxassisligi: "+shaxs[5]);
}
UMUMIY_TA_LIM * Shaxs;

```

### ***Unit\_Anketa.cpp fayli:***

```

...
void __fastcall TForm1::Btn_Ta_limClick(TObject *Sender)
{
if(RadioGroup_talim_turi->ItemIndex==0)
Shaxs=new UMUMIY_TA_LIM;
else Shaxs=new OLIY_TA_LIM;
Shaxs->Anketa_shakli(StringGrid_Anketa);
}

```



```

}
void __fastcall TForm1::Btn_Anketa_QabulClick(TObject *Sender)
{
  Shaxs->Anketa_qabul(StringGrid_Anketa);
}
void __fastcall TForm1::Btn_Anketa_ChopClick(TObject *Sender)
{
  Shaxs->Chop_etish(Memo_Anketa);
}

```

### ***Amaliy topshiriqlar***

1. Sonlar progressiyasini hosil qiluvchi PROGRESSIYA abstrakt sinfi aniqlansin. Uning vorisi sifatida arifmetik va geometrik progressiyalar ustida amal bajaruvchi ARIFM\_PROGRESS va GEOM\_PROGRESS sinflari aniqlansin. Undagi progressiya hadini topish, berilgan n sondagi progressiya hadlarini yig'indisini hisoblash funksiya-a'zolari virtual qilib aniqlansin.
2. 10 lik sanoq sistemasida berilgan sonni o'qish, saqlash va chop qilish amallarini o'z ichiga olgan tayanch SANOQ\_SISTEMA\_10 sinfi va uning vorisi sifatida berilgan sonni 2, 8 va 16 lik sanoq sistemasidagi ko'rinishini chop qiluvchi Sonni\_chop\_qilqish() virtual funksiya-a'zosi bo'lgan SANOQ\_SIS\_2, SANOQ\_SIS\_8 va SANOQ\_SIS\_16 sinflar aniqlansin.
3. Kvadrat tenglama ildizlarini hisoblaydigan tayanch KVADRAT, uning vorisi BIKVADRAT sinflar yaratilsin va ularda tenglama ildizini topadigan Kv\_ildiz() polimorf funksiya aniqlansin.
4. Matritsa uchun MATRITSA tayanch sinfi yaratilsin. Uning vorisi sifatida to'g'ri burchakli (TB\_MATRITSA) va kvadrat (KB\_MATRITSA) matritsalar ustida qo'shish, ayirish va ko'paytirish amallarni bajaradigan voris sinflar yaratilsin. Kvadrat matritsa elementlari bosh diagonalga nisbatan simmetrik va u uchburchak ko'rinishda berilgan. Yuqoridagi amallarni bajaruvchi funksiyalar polimorf qilib aniqlansin.
5. "Hayot" o'yini tirik kataklar gipotetik koloniyasi hayotining modelidir. Unda kataklar quyida keltirilgan qoida asosida tirik qoladi, ko'payadi yoki o'ladi (mos rasmlarga qaralsin):
  - a) katak tirik qoladi, agar u o'z atrofida faqat ikki yoki uchta qo'shnilarga ega bo'lsa (mumkin bo'lgan sakkiztadan);
  - b) katak o'ladi, agar uning atrofida faqat qo'shnilardan bittasi yoki umuman bo'lmasa;
  - d) agar katak atrofida to'rtta yoki undan ko'p qo'shnilar bo'lsa, katak o'ladi;

e) bo‘sh joy atrofida uchta katak bo‘lsa, bu joyda katak paydo bo‘ladi. Boshlang‘ich holatga ko‘ra kataklar hayoti tasvirlansin.

X		
	X	X

a)

	X	

b)

X	X	
	X	X
X		

d)

X		X
	X	

e)

6.  $n$  o‘lchovli fazoda koordinatalari bilan berilgan ikki nuqta orasidagi masofani hisoblaydigan MASOFA tayanch sinfi aniqlansin. Uning vorisi sifatida Dekart, Chebishev va Hemming fazosida nuqtalar orasidagi masofani hisoblaydigan DEKART, CHEBISHEV va HEMMING sinflari yaratilsin. Nuqtalar orasidagi masofani hisoblovchi  $\rho()$  funksiyasi polimorf qilib aniqlansin. Metrikalar quyidagi ko‘rinishga ega:

- Evklid:  $\rho(x, y) = \sqrt{(x_1 - y_1)^2 + \dots + (x_n - y_n)^2}$  ;

- Chebishev:  $\rho(x, y) = \max_{1 \leq i \leq n} \{ |x_i - y_i| \}$  ;

- Hemming:  $\rho(x, y) = \sum_{i=1}^n |x_i - y_i|$  .

7. Butun sonlarning chiziqli ro‘yxatini qayta ishlash uchun RUYXAT tayanch sinfi yaratilsin. Uning vorisi sifatida stek va navbat tuzulmalari uchun STEK va NAVBAT sinflari hosil qilinsin va elementlarni ro‘yxatga joylash, undan olish amallarini bajaruvchi funksiyalar polimorf tarzda aniqlansin.
8. Yulduzcha va shina topologiyalarida to‘r hosil qilish uchun umumiy xarajat hisoblansin. Buning uchun tayanch TARMOQ sinfi yaratilsin. Sinfda qurilmalar soni ( $n$ ), qurilmalargacha bo‘lgan masofalar ( $a[n]$ ), sim narxi ( $q$ ) konnektor narxi ( $p$ ) berilgan-a’zolari va ularni qayta ishlash funksiya-a’zolari aniqlansin.

TARMOQ sinfidan hosilaviy YULDUZ va SHINA sinflar yaratilsin.

YULDUZ sinfida xarajatlar  $S = \sum_{k=1}^n (a_k q + 2p)$  formula bilan, SHINA sinfida

esa  $S = \max_{1 \leq k \leq n} \{ a_k \} q + np$  formula bilan hisoblanadi.

Berilgan topologiya va o‘lchamlar bo‘yicha  $S$  xarajat hisoblansin.

9. Telefon muloqoti xarajatini hisoblovchi TARIF sinfi tuzilsin. TARIF sinfida kiruvchi va chiquvchi qo‘ng‘iroqlar daqiqalari soni saqlansin. TARIF sinfidan UNIVERSAL va PROGRESS sinflari voris qilib tuzilsin. Universal (UNIVERSAL) tarifida xarajat  $S = nA + mB$  formula yordamida hisoblanadi. Bu yerda  $n$  - kiruvchi daqiqalar soni,  $m$  - chiquvchi daqiqalar soni va  $A = 0$ ,  $B = 0.03\$$  mos ravishda bir daqiqa uchun to‘lov narxi.

Progress (PROGRESS) tarifida xarajat  $S = nA + m_1B_1 + m_2B_2 + m_3B_3$  ko‘rinishida hisoblanadi. Bu yerda  $n$  - kiruvchi daqiqalar soni,  $m$  - chiquvchi daqiqalar soni,  $A=0.01\$$ ,  $B_1=0.02\$$ ,  $B_2=0.01\$$ ,  $B_3=0.005\$$ .

Hisoblash shartlari:

- agar  $m \leq 50$  bo‘lsa,  $m_1 = m, m_2 = m_3 = 0$ ;
- agar  $50 < m \leq 100$  bo‘lsa,  $m_1 = 50, m_2 = m-50, m_3 = 0$ ;
- agar  $m > 100$  bo‘lsa,  $m_1 = 50, m_2 = 50, m_3 = m-100$ .

Berilgan tarif, kirish-chiqish qo‘ng‘iroqlariga ko‘ra oylik xarajatlar polimorf funksiyalar orqali hisoblansin.

0. Jismoniy shaxsdan daromad solig‘ini olish masalasini yechish uchun quyidagi belgilashlarni kiritilgan (pul birligida):  $Min\_IH$  - minimal ish haqi;  $DM$  - daromad miqdori;  $DS$  - daromad solig‘i.

Daromad solig‘ini olish qoidalari:

a) imtiyozga ega bo‘lmagan shaxslar uchun  $DM$  shkalalari:

- 1)  $[1*Min\_IH..5*Min\_IH]$  pul miqdori uchun daromad solig‘i - 9%;
- 2)  $[5*Min\_IH+1..10*Min\_IH]$  pul miqdori uchun daromad solig‘i - 16%;
- 3)  $10*Min\_IH$  dan katta pul miqdori uchun daromad solig‘i - 22%.

b) nogiron shaxslar uchun  $DM$  shkalalari:

- 1)  $[1*Min\_IH..4*Min\_IH]$  pul miqdori uchun daromad solig‘i - 0%;
- 2)  $[4*Min\_IH+1..5*Min\_IH]$  pul miqdori uchun daromad solig‘i - 9%;
- 3)  $[5*Min\_IH+1..10*Min\_IH]$  pul miqdori uchun daromad solig‘i - 16%;
- 4)  $10*Min\_IH$  dan katta pul miqdori uchun daromad solig‘i - 22%.

Jismoniy shaxsdan olinadigan daromad solig‘ini hisoblaydigan SOLIQ sinfi aniqlansin. Uning vorisi sifatida mos ravishda nogironlardan olinadigan daromad solig‘ini hisoblovchi SOLIQ\_NOGIRON sinfi aniqlansin. Ikkita sinfdan Daromad\_solig() funksiyasi polimorf qilib aniqlansin.

11. To‘rtburchak yuzasini hisoblash uchun TURTBURCHAK tayanch sinfi va romb (ROMB), kvadrat (KVADRAT), to‘g‘ri to‘rtburchak (TTURTBURCHAK), trapetsiya (TRAPETSIYA) va parallelogram (PARALLELOGRAM) voris sinflari yaratilsin va ularda yuzani hisoblovchi yuza() funksiyasi polimorf ko‘rinishda aniqlansin.
12. Berilgan sonlar ketma-ketligini tartiblash uchun TARTIBLASH sinfi aniqlansin. Uning vorisi sifatida PUFKCHA, ORAGA\_JOYLASH va PUFKCHA\_SURISH sinflari yaratilsin. Ularda mos ravishda 15.6, 15.7 masalarda keltirilgan algoritmlar va “*pufakcha usuli*” algoritmining takomillashgan varianti amalga oshirilsin: massivning yonma-yon joylashgan  $X_i$  va  $X_{i+1}$  elementlari o‘rni almashgandan keyin tartiblanish buzilmagan holda  $X_{i+1}$  element chap tomonga mumkin qadar ko‘chiriladi. Barcha sinflarda tartiblashni amalga oshiradigan virtual

Tartiblash() funksiya-a'zo aniqlasin.

13. Satr ko'rinishida berilgan matematik funksiyaning aniq integralini taqribiy hisoblash uchun quyidagi sinflar aniqlansin:

- IFODA sinfi satr-ifodani qayta ishlash uchun;
- INTEGRAL sinfi IFODA sinfining vorisi bo'lib, satr ko'rinishida berilgan matematik ifodaning qiymati va integralini hisoblash uchun;
- INTEGRAL\_TRAPETSIYA sinfi INTEGRAL sinfining vorisi bo'lib, berilgan oraliq uchun satr ko'rinishidagi funksiya integralini trapetsiya usulida hisoblash uchun;
- INTEGRAL\_SIMPSON sinfi INTEGRAL sinfining vorisi bo'lib, berilgan oraliq uchun satr ko'rinishidagi funksiya integralini Simpson usulida hisoblash uchun.

Sinflarda integralni hisoblash virtual funksiya-a'zo ko'rinishida amalga oshirilsin.

14. Ko'rinishi  $\alpha_n \alpha_{n-1} \dots \alpha_0, \alpha_{-1} \dots \alpha_{-m} (\beta_0 \beta_1 \dots \beta_k)$  bo'lgan davriy o'nli kasrni ratsional kasrga aylantirish uchun quyidagilarni amalga oshirish kerak:

a) agar  $m = 0$ , yani oddiy davriy o'nli kasr uchun  $c \frac{a}{b}$  ko'rinishidagi ratsional kasrga o'tiladi, bu yerda  $a = \beta_0 \beta_1 \dots \beta_k$ ,  $b = \underbrace{9 \dots 9}_k$  va  $c = \alpha_n \alpha_{n-1} \dots \alpha_0$  ko'rinishidagi sonlar;

b) agar  $m > 0$ , yani aralash davriy o'nli kasr uchun  $\frac{a_0 - a_1}{b}$  ko'rinishidagi ratsional kasrga o'tiladi. Bu yerda  $a_0 = \alpha_n \alpha_{n-1} \dots \alpha_0 \alpha_{-1} \dots \alpha_{-m} \beta_0 \beta_1 \dots \beta_k$ ,  $a_1 = \alpha_n \alpha_{n-1} \dots \alpha_0 \alpha_{-1} \dots \alpha_{-m}$  va  $b = \underbrace{9 \dots 9}_k \underbrace{0 \dots 0}_m$  ko'rinishidagi sonlar.

Misol uchun: 1)  $2,(27) = 2 \frac{27}{99}$ ;      2)  $3,2(15) = 3 \frac{215 - 20}{990} = 3 \frac{213}{990}$ .

Berilgan davriy o'nli kasrni satr sifatida o'qish va unga mos ratsional kasrni chop qiladigan DAVRIY\_KASR sinfi aniqlansin. Uning vorisi sifatida yuqoridagi punktlarga mos ravishda ODDIY\_DK va ARALASH\_DK sinflari yaratilib ularda ratsional kasr hosil qiluvchi Ratsional\_kasr() funksiya-a'zosi polimorf tarzda aniqlansin.

15. Berilganlarni "k ta guruhlar o'rtalari" algoritmi bo'yicha guruhlash. Algoritm uchun m satr va n ustundan iborat matritsa va k soni boshlang'ich berilganlar hisoblanadi. Bu yerda matritsaning har bir satri  $(S_1, S_2, \dots, S_m)$  n o'lchovli fazodagi nuqtalar, k – guruhlar soni. Algoritmning birinchi qadamida k ta o'zaro kesishmaydigan  $C_1, C_2, \dots, C_k$

guruhlar markazlari ixtiyoriy ravishda  $S_1, S_2, \dots, S_m$  nuqtalar orasidan tanlanadi.

Guruh tarkibi quyidagi qoida bilan aniqlanadi:  $S_i \in C_j$  bo'ladi, agar  $\rho(S_i, C_j) = \min_{1 \leq r \leq k} \rho(S_i, C_r)$  bo'lsa. Bu yerda  $\rho(S_i, C_j)$  - bu  $S_i$  nuqta va  $C_j$  guruh markazi orasidagi masofa.

Guruhlar tarkibi aniqlangandan keyin har bir  $C_v$  guruh uchun uning markazi topiladi. Guruh markazi unga kiruvchi nuqtalar o'rta arifmetigi bilan aniqlanadi:

$$C_v = \frac{\sum_{S_i \in G_v} S_i}{|G_v|}, v = \overline{1, k}.$$

Keyingi qadamlarda guruhlarining tarkibi va markazi qayta aniqlanadi. Bu jarayonlar iterativ ravishda amalga oshiriladi. Guruhlar tarkibi va markazining o'zgarmay qolishi hisoblash jarayonini to'xtatish alomati bo'ladi. Berilgan  $n$  o'lchamli fazoda berilgan  $x = (x_1, x_2, \dots, x_n)$  va  $y = (y_1, y_2, \dots, y_n)$  nuqtalar orasidagi masofani hisoblashda Dekart, Chebishev va Xemming metrikalaridan foydalanilsin (25.5 masalaga qarang).

Masalani yechish uchun tayanch GURUH sinfi va uning vorislari sifatida GURUH\_EVKLID, GURUH\_XEMMING, GURUH\_CHEBISHEV sinflari aniqlansin. Barcha sinflarda masofa hisoblaydigan virtual RooXY() funksiya-a'zo bo'lsin.

16. Tekislikda berilgan sohani diskretlash deb sohanini elementar bo'laklarga ajratish tushiniladi. Soha  $(x_1, y_1), (x_2, y_2), (x_3, y_3)$  va  $(x_4, y_4)$  tugun nuqtalari bilan to'rtburchak k'rinishida beriladi. Sohani quyidagi usullarda diskretlash mumkin:

- 1) kvadrat elementlarga bo'lish usuli. Soha tomonlari o'zaro teng ikki bo'lakka bo'linadi;
- 2) kub elementlarga bo'lish usuli. Soha tomonlari o'zaro teng uchta bo'laklarga bo'linadi.

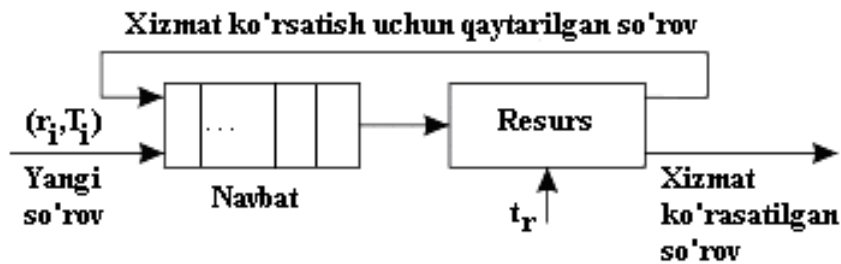
Tayanch DISKRETLASH sinfi va uning vorislari sifatida yuqorida qayd qilingan diskretlash usullariga mos ravishda DISKRET\_KV va DISKRET\_KUB sinflari yaratilsin. Voris sinflarda Diskretlash() virtual funksiyalari aniqlanib, ular diskretlangan sohaning tugun nuqtalari koordinatalarini qaytarsin.

17. Multimasalali operatsion sistemalarda jarayonlar o'rtasida resurslarni taqsimlash masalasi quyidagicha tavsiflanadi:

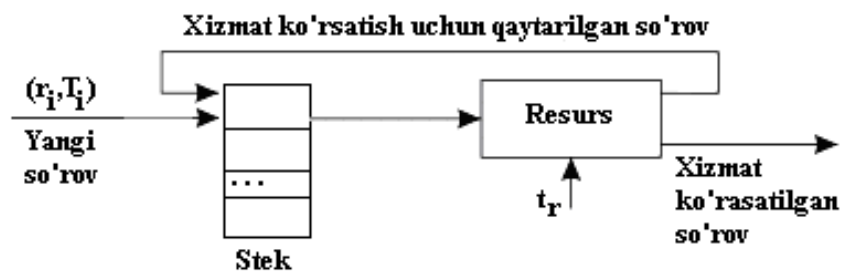
Butun  $n$  son va  $n$  ta jarayon (so'rovlar) berilgan. Har bir  $i$  ( $i = \overline{1, n}$ )-

jarayon  $(r_i, T_i)$  juftlik bilan beriladi. Bu yerda  $r_i$ - jarayon tomonidan talab qilinadigan resurs vaqti,  $T_i$  jarayonning o‘zidan oldingi  $T_{i-1}$  jarayondan qancha vaqt intervalidan keyin navbatga kelganligini bildiradi ( $T_0 = 0$ ). Har bir jarayonga resursning  $t_r$  vaqt kvanti ajratiladi. Agar jarayon uchun resurs talabi to‘liq bajarilsa u yo‘qoladi, aks holda u navbat oxiriga qaytib keladi. Jarayonga resurs ajratishning quyidagi usullari mavjud:

a) FCFS usuli. Navbatdagi jarayonlarga “*Birinchil kelganga birinchi xizmat*” tamoyili bo‘yicha resurs ajratiladi (rasmga qarang);



b) LCFS usuli. Navbatdagi jarayonlarga “*Oxirgil kelganga birinchi xizmat*” (stek) tamoyili bo‘yicha resurs ajratiladi (rasmga qarang).



Jarayonlar o‘rtasida resurslarni taqsimlashni amalga oshiruvchi RESURS sinfi va uning vorislari sifatida RESURS\_FCFS (FCFS usuli) va RESURS\_LCFS (LCFS usuli) sinflari aniqlansin. Barcha sinflarda berilgan  $T$  vaqt uchun  $[0...T]$  intervalning har bir vaqt birligida jarayonlarga resurs ajratilishini (yoki navbatda kutishini) chop etuvchi virtual Resurs\_Taqsimoti() funksiya-a’zo aniqlansin.

18. Bir o‘lchamli  $y = f(x)$  funksiyaning  $x_0, x_1, \dots, x_n$  nuqtalardagi  $f(x_0), f(x_1), \dots, f(x_n)$  qiymatlari jadval ko‘rinishida berilgan.  $f(x)$  funksiyaning  $x^* \notin \{x_0, x_1, \dots, x_n\}$  nuqtadagi  $y^*$  qiymatini Lagranj yoki Nyuton interpolatsion formulasi bilan hisoblash mumkin:

1) Lagranj interpolatsion formulasi -  $y^* = L_n(x^*) = \sum_{k=0}^n \frac{\prod_{j \neq k} (x^* - x_j)}{\prod_{j \neq k} (x_k - x_j)} f(x_k)$ ;

2) Nyuton interpolatsion formulasi -

$$y^* = P_n(x^*) = f(x_0) + (x^* - x_0)f(x_0, x_1) + (x^* - x_0)(x^* - x_1)f(x_0, x_1, x_2) + \dots + (x^* - x_0)(x^* - x_1) \dots (x^* - x_{n-1})f(x_0, x_1, \dots, x_n),$$

bu yerda  $f(x_j, x_{j+1}, \dots, x_{j+k}) = \sum_{i=j}^{j+k} \frac{f(x_i)}{\prod_{l=j, l \neq i}^{j+k} (x_i - x_l)}$ .

Jadval ko‘rinishidagi funksiya berilganlarini o‘qish, saqlash va chop etish bilan bog‘liq amallarni o‘z ichiga oluvchi tayanch JADVAL\_FUN sinfi aniqlansin. Uning vorislari sifatida LAGRANJ va NYUTON sinflari yaratilsin va ularda berilgan  $x^*$  bo‘yicha  $y^*$  qiymatni hisoblovchi Interpolyatsiya() virtual funksiyalar aniqlansin.

19. O‘zaro ta’sirda (bo‘g‘liqlikda) bo‘lgan ikkita turlarining, xususan quyonlar (Quyon sinfi) va tulkilar (Tulki sinfi) ko‘pyishini hisoblash modellariga asoslangan masala yechilsin.

Quyonlar sonini o‘shining cheklanmagan modelida (Quyon sinfining vorisi Quyon\_Cheklanmagan sinfi) turning keyingi yildagi soni qandaydir  $a$  - o‘shish koeffitsientiga bog‘liq hisoblanadi:

$$x_{n+1} = a * x_n .$$

Cheklangan o‘shish modelida esa ko‘pyib ketish natijasida, ozuqa etishmasligi, kasalliklar va boshqa ta’sirlar inobatga olinadi va yillik o‘shish formulasi quyidagi ko‘rinishga ega bo‘ladi (Quyon\_Cheklanmagan sinfining vorisi Quyon\_Cheklangan sinfi):

$$x_{n+1} = (a - b * x_n) * x_n .$$

Bu yerda  $b$  - ko‘pyib ketish koeffitsienti bo‘lib u odatda,  $a$  sonidan ancha kichik son.

Agar cheklangan o‘shish modelida turni ovlashlar soni –  $c$  ham inobatga olinsa, formulaning ko‘rinishi quyidagicha bo‘ladi (Cheklangan sinfining vorisi Quyon\_Ovlanish sinfi) :

$$x_{n+1} = (a - b * x_n) * x_n - c .$$

Odatda hayvonlarning biror turi boshqalardan ajralgan holda yashamaydi va ular bilan qandaydir aloqada boladi. Bunday munosabatlarining eng muhimi yirtqichlar va ularning qurbonlari munosabati (“yirtqich-qurbon”) bo‘lib, unda  $x_n$ - qurbonlar (quyonlar) soni va  $y_n$ - yirtqichlar (tulkilar) soni o‘zaro bog‘liq.

Agar qurbon bo‘lmasa yirtqichlar soni kamayib boradi va buni quyidagi rekurent formulada ko‘rsatish mumkin (Tulki sinfining vorisi Tulki\_Quyon\_kam sinfi):

$$y_{n+1} = d * y_n .$$

Bu yerda  $d(d < 1)$ - turning kamayish tezligi.

Qurbonlar yetarlicha bo‘lganda yirtqichlar soni ularning sonlariga  $e$

soni bilan tavsiflanuvchi proporsionallik koeffitsienti bilan o'sadi (Tulki\_Quyong\_kam va Quyong\_Ovlanish sinfining vorisi Tulki\_Quyong\_etarli sinfi):

$$y_{n+1} = d * y_n + e * x_n * y_n.$$

Quyonglar ko'payishida qurbonlarni inobatga olish quyidagi formulada keltirilgan (Tulki\_Quyong\_etarli sinflarining vorisi Tulki\_Quyong sinfi):

$$x_{n+1} = (a - b * x_n) * x_n - c - f * x_n * y_n.$$

Bu yerda f - quyonglarning qurbon bo'lish koeffitsienti.

Mos sinflar shajaralarida hayvonlar sonini hisoblaydigan Qurbonlar() va Yirtqichlar() polimorf funksiya-a'zolari aniqlansin.

Turlar ko'payishining yuqorida keltirilgan barcha variantlarini inobatga olgan holda quyonglar va tulkilarning n (n>0) yildan keyingi miqdorlari hisoblansin.

20. O'zbek tilida berilgan so'zni ko'rsatilgan tilga tarjimasi amalga oshirilsin. Buning uchun taynch SATR sinfi aniqlanib, uning a'zolari sifatida S<sub>1</sub> va S<sub>2</sub> satrlar - mos ravishda tarjima qilinadigan so'z va uning tarjimasi, hamda satrlarni o'qish va chop qilish funksiyalari aniqlansin. SATR sinfining vorisi sifatida aniq bir tilga tarjimani amalga oshiruvchi TARJIMA\_UZB\_ING (ingliz tiliga), TARJIMA\_UZB\_NEM (nemis tiliga) va TARJIMA\_UZB\_RUS (rus tiliga) sinflari aniqlansin. Barcha voris sinflar o'zbek tilidagi so'z tarjimasini amalga oshiruvchi polimorf tarjima() funksiya-a'zoga ega bo'lsin.

21. Kompyuter vositasida tasodifiy sonlarni hosil qilish uchun taynch TSD\_SON sinfi va uning vorislari bo'lgan NEYMAN\_TSDS va REKURENT\_TSDS sinflari aniqlansin.

NEYMAN\_TSDS sinfida Djon fon Neyman tomonidan taklif qilingan usul (o'rta kvadratik usul) amalga oshirilsin. Bu usulda oldingi tasodifiy son kvadratga oshiriladi va natijagi sonning o'rtadagi raqamlar olinadi. Masalan, uch xonali tasodifiy son hosil qilish zarur bo'lsa va oldingi tasodifiy son 121 bo'lsa, uning kvadrati 14641 va o'rtadagi 3 ta raqam 464 bo'ladi.

REKURENT\_TSDS sinfida R<sub>1</sub>, R<sub>2</sub>, ..., R<sub>N</sub> (N > 0) tasodifiy sonlarni

$R_i = \left\{ \frac{a_i}{m} \right\}$  ko'rinishida hisoblash amalga oshirilsin. Bu yerda

$a_{i+1} = \lfloor [a_i] * m + c \rfloor$ . Formuladagi  $\lfloor x \rfloor - x$  sonning butun qismi, m va c - musbat butun sonlar (m soni c sonidan yetarlicha katta), a<sub>i</sub> - haqiqiy son, a<sub>0</sub> - qandaydir boshlang'ich butun son.

Misol uchun a<sub>0</sub> = 1, c = 3, m = 32749 qiymatlari uchun tasodifiy sonlarni hosil qilsih mumkin.



Sinflarda har bir usulga mos ravishda `Tasodifiy_son()` polimorf funksiya-a'zolar aniqlansin.

Berilgan `N` soni uchun yuqorida keltirilgan ikkita usulda `N` tadan tasodifiy sonlar hosil qilinsin. Har bir usul uchun sonlarning tushish chastotalari aniqlansin va sonlar ketma-ketligi o'zaro taqqoslansin.

22. Programmash muhitlarida vaqtni aniqlovchi turlar (masalan, `TDATE` turi) xotirada haqiqiy son ko'rinishida bo'lib, uning butun qismi qandaydir sanadan (masalan, 30.12.1899 sanasidan) o'tgan kunlar sonini bildirsa, kasr qismi sanaga bo'g'liqmas ravishda 24 soatlik sutka qismini ifodalaydi. Misol uchun 14.10.2013 17:17:52 vaqtga 41561,7207489352 soni mos keladi. `SANA_VAQT` tayanch sinfi yaratilsin. Uning vorislari sifatida berilgan `S` haqiqiy soniga ko'ra sanani turli ko'rinishda chop qiluvchi `sana_vaqt()` polimorf funksiya-a'zolariga ega voris sinflar yaratilsin:

- `SV_UZUN` sinfi - `<yil.oy.kun soat.minut.sekund>` ko'rinishda;
- `SV_SANA` sinfi - `<yil.oy.kun>` ko'rinishda;
- `SV_SOAT` sinfi - `< soat.minut.sekund>` ko'rinishda;
- `SV_OY_NOMI` sinfi – sanadagi oy nomini bilan, sekund ko'rsatilmagan ko'rinishda.

## 26. Qoliplar. STL kutubxonasi

Ushbu bo'limda masalalar STL kutubxonasidan foydalangan holda qoliplar vositasida yechiladi.

### *Namunaviy masala*

`C++` tilida yozilgan ilova matnidagi leksemalarning necha marta uchrashi (chastotasi) aniqlansin.

### *Yechish usuli*

Ilova bo'lagining matni satr ko'rinishida beriladi. `C++` tilida leksemalar bir-biridan, ajratuvchilar deb nomlanuvchi belgilar vositasida ajratiladi. Qo'yilgan masalani yechishda berilgan ajratuvchilar to'plami bo'yicha matndagi satrostilarini (leksemalarni) ajratish va ularning matnga kirishlar sonini aniqlash zarur bo'ladi. Buning uchun assotsiativ konteynerlardan, xususan, `set` va `map` sinflaridan foydalanish mumkin. Ajratuvchilar to'plamini saqlash uchun `set<char>` ko'rinishidagi belgilar to'plami (ajratuvchilar to'plam ostisi), leksema va uning matndagi uchrashlar soni juftligini ifodalash uchun esa `map<String, int>` ko'rinishida qolip initsializatsiyalanadi. Qaralayotgan masalada `map` konteynerini

indeksi string turida bo‘lgan butun sonlar massiviga o‘xshatish mumkin, massiv elementi – indeksning satrda uchrashlar soni.

Ilova shaklida Memo\_Ilova\_matni va Memo\_Leksemalar Memo komponenta-lari aniqlangan bo‘lib, ularda mos ravishda ilova matni va unung leksik tahlil natijasi chop etiladi. Ilova boshqaruvi Button\_LeksemalarClick tugmasini bosish hodisasini qayta ishlash funksiyasida amalga oshiriladi. Ilovada satrdan leksemani ajratib olish uchun String Leksema (String & matn) funksiyasi aniqlangan bo‘lib, u javob tariqasida berilgan satrdan (matn) qirqib olingan leksemani (leksema) qaytaradi.

### *Ilova matni*

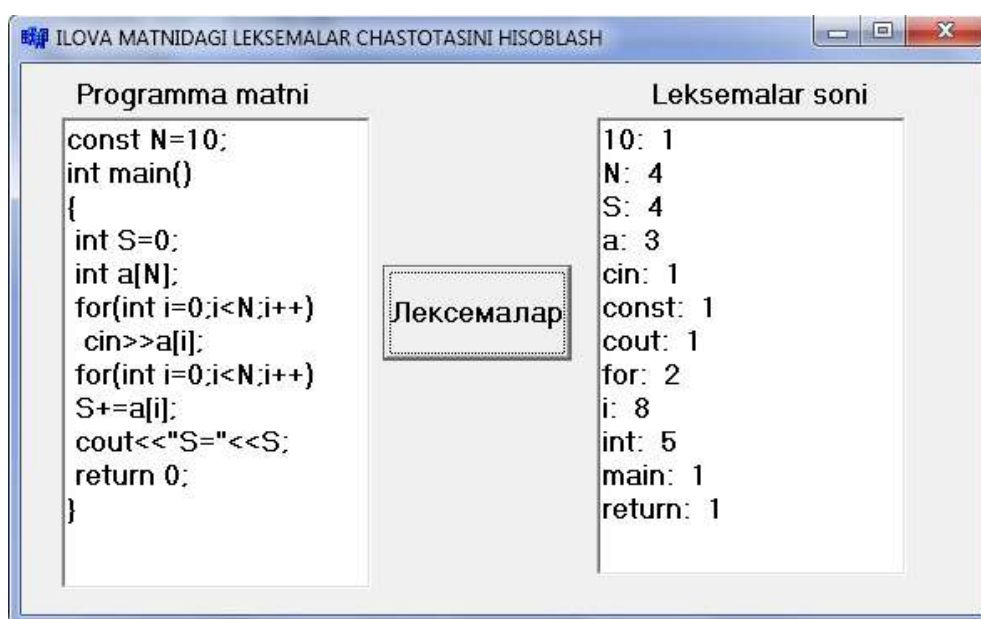
```
#include <map.h>
#include <set.h>
#include <string.h>
using namespace std;
char ajratuvchi[25]=
    {"", '=', '>', '<', '\n', '\r', ':', ';', '?', '!', ':', ',', '}', '{', ')', '(', '[', ']', '+', '-', '*', '/', '%', '&'};
set<char>ajratuvchilar(ajratuvchi,ajratuvchi+25);
typedef map<String,int> Leksemalar_Soni;
String Leksema(String & matn)
{
    int joy;                // satrdagi leksema o‘rni
    // satr boshida keluvchi ajratuvchilarni o‘chirish
    while(!matn.IsEmpty() && ajratuvchilar.count(matn[1])) matn.Delete(1,1);
    if(matn.IsEmpty()) return matn;
    joy=0;
    int uzunlik=matn.Length();
    //leksemni o‘ng chegarsini topish
    while(joy<uzunlik && !(ajratuvchilar.count(matn[++joy])));
    String leksema=matn.SubString(1,joy-1); // leksemani ajratib olish
    matn.Delete(1,joy-1); // leksemani satrdan o‘chirish
    return leksema;
}
void __fastcall TForm_Leksema::Button_LeksemalarClick(TObject *Sender)
{
    Leksemalar_Soni leksema_soni;
    String matn=Memo_Ilova_Matni->Lines->Text;
    while(!matn.IsEmpty())
    {
        String leksema=Leksema(matn);
```

```

if(!leksema.IsEmpty()) ++leksema_soni[leksema]; // indeksi leksema bo'lgan
    //massiv qiymatini bittaga oshirish, indeks mavjud bo'lmasa u yaratiladi
}
//iterator vositasida konteyner elementlarini chop etish
map<String,int>::const_iterator it=leksema_soni.begin();
for(it; it!=leksema_soni.end(); ++it)
Memo_Leksemalar_Soni->Lines->Add(it->first+": "+IntToStr(it->second));
}

```

Ilovaning ish holatlari quyidagi rasmda keltirilgan.



### ***Amaliy topshiriqlar***

1. STL kutubxonasidagi stack konteyner sinfidan foydalangan holda temir yo'l-dagi 'T' ko'rinishdagi saralash tugunining ishlashi modellashtirilsin. Unda berilgan ikki turdagi vagonlar mos ravishda chap va o'ng yo'llarga ajratilsin.
2. STL kutubxonasidagi stack konteyner sinfidan foydalangan holda labirintda yo'l topish masalasi yechilsin. Labirint kvadratlardan tashkil topgan matritsa ko'rinishida berilgan. Kvadrat ochiq yoki yopiq bo'lishi mumkin. Ochiq kvadratga kirish mumkin, yopiq - yo'q. Ochiq kvadratga tomonlaridan kirish mumkin, uchlaridan kirish mumkin emas.
3. Fayl tizimidagi katalogni boshqarish jarayonini modellashtiruvchi ilova yaratilsin. Katalogdagi har bir fayl <fayl nomi>, <yaratilish vaqti>, <faylga murojaatlar soni> malumotlardan iborat.

Quyidagi masalalar yechilsin:

- fayllar katalogining boshlang'ich holatini shakllantirish;
- fayl katalogini chop etish;
- berilgan vaqtdan oldin yaratilgan fayllar katalogdan o'chirilsin;

- eng ko‘p murojaat bo‘lgan fayllar nomlarinin chop etish.

Masalani yechishda STL kutubxonasidagi list konteyner sinfidan foydalanilsin.

4. Avtobus parki ishlashini modellashtiruvchi ilova yaratilsin. Avtobus haqidagi malumotlar - <avtobus nomeri>, <haydovchining familiyasi va ismusharifining bosh harflari> va <marshrut nomeri> maydonlaridan iborat.

Quyidagi masalalar yechilsin:

- parkdagi avtobus haqidagi boshlang‘ich berilganlarni ro‘yxat ko‘rinishida shakllantirish;

- avtobusni parkdan chiqishini imitatsiya qilish: avtobus nomeri kiritiladi; ilova parkdagi avtobuslar ro‘yxatidan mos avtobusni o‘chiradi va uni marshrutdagi avtobuslar ro‘yxatiga kiritadi.

- avtobusni parkga kelishiini imitatsiya qilish: avtobus nomeri kiritiladi; ilova marshrutdagi avtobuslar ro‘yxatidan ushbu avtobusni o‘chiradi va uni parkdagi avtobuslar ro‘yxatiga qo‘shadi;

- parkda va marshrutda bo‘lgan avtobuslar haqida ma‘lumot chiqarish.

Masalani yechishda STL kutubxonasidagi list konteyner sinfidan foydalanilsin.

5. Aviabiletlarga talablarni hisob qilish masalasi yechilsin. Talab quyidagilarni o‘z ichiga oladi - <uchish manzil>, <reys nomeri>, <yo‘lovchi familiyasi va shariflari> va <uchish sanasi>.

Ilova quyidagilarni amalga oshirishi zarur:

- yangi talabni ro‘yxatga qo‘shish;

- ko‘rsatilgan talabni ro‘yxatdan o‘chirish;

- ko‘rsatilgan reys va sana bo‘yicha talablarni chop etish;

- barch talablarni chop etish.

Masalani yechishda STL kutubxonasidagi list konteyner sinfidan foydalanilsin.

6. Kutubxonadagi kitoblar hisobini olib boruvchi ilova tuzilsin. Kitob haqidagi ma‘lumot quyidagilarni o‘z ichiga oladi - <muallifning ismu shariflari>, <kitob nomi>, <nashr yili> va <kitobning kutubxonadagi nusxalari soni>.

Ilova quyidagilarni amalga oshirishi zarur:

- yangi kitoblarni kutubxona ro‘yxatiga qo‘shish;

- ko‘rsatilgan kitobni ro‘yxatdan o‘chirish;

- mualliflar familiyalari bo‘yicha tartiblanga kitoblar ro‘yxatini chop etish;

- nasr yillari bo‘yicha tartiblanga kitoblar ro‘yxatini chop etish.

Masalani yechishda STL kutubxonasidagi multimap konteyner sinfidan foydalanilsin, kalit sifatida <muallifning ismu shariflari > maydoni ishlatilsin.

7. “Yon daftar” ilova tuzilsin. Unda ixtiyoriy sondagi yozuvlar bilan ishlash, qandaydir alomat (masalan, familiya, tug‘ilgan kun yoki telefon nomeri) bo‘yicha yozuvni izlash, yozuvlarni qo‘shish va o‘chirish va turli maydonlar bo‘yicha yozuvlarni tartiblash amalga oshirilsin.

Masalani yechishda STL kutubxonasidagi map yoki multimap konteyner sinfidan foydalanilsin.

8. Kwartira almashtirish va almashish variantini izlash bo'yicha talabnomalar hisobini olib boruvchi ilova tuzilsin. Har bir talabnoma ikkita kvartira to'g'risida ma'lumotdan tashkil topgan: talab qilingan (izlanayotgan) va mavjud. Har bir kvartira haqidagi ma'lumot o'z ichiga <xonalar soni>, <maydon>, <qavat> va <tuman> maydonlarini o'z ichiga oladi.  
Quyidagi masala ostilari yechilsin:
- almashtirish bo'yicha talabnomani kiritish;
  - kartotekadan kvartirani izlash: talab va taklif xonalar soni va qavati bo'yicha ustma-ust tushsa, "maydon" ko'rsatgichi bo'yicha farq 10% dan oshmasa mos karta chop etiladi va u ro'yxatdan o'chiriladi. Aks holda talabnoma kartotekaga kiritiladi;
  - barcha kartotekani chop etish.
- Masalani yechishda STL kutubxonasidagi list konteyner sinfidan foydalanilsin.
9. Temir yo'l vokzalidagi poyezdlar harakati bo'yicha avtomatlashtirilgan tizim yarailsin. Tizim har bir poyezd haqidagi ma'lumot <poyezd nomeri>, <borish stansiyasi> va <jo'nash vaqti> maydonlarini o'z ichiga oladi.  
Quyidagi masala ostilari yechilsin:
- tizimga boshlang'ich ma'lumotlarni kiritish (klaviatura yoki fayldan);
  - barcha poyezdlar bo'yicha ma'lumotlarni chop etish;
  - nomeri ko'rsatilgan poyezd haqida ma'lumotlarni chop etish;
  - so'ralgan manzilga boradigan poyezdlar haqida ma'lumotlarni chop etish.
- Masalani yechishda STL kutubxonasidagi vector konteyner sinfidan foydalanilsin.
10. O'zbekcha-inglizcha va inglizcha-o'zbekcha lug'at yaratilsin. Lug'atning "berilganlar bazasi" so'zning sinonimik tarjima variantlaridan tashkil topgan bo'lsin.  
Quyidagilar amalga oshirilsin:
- lug'at "berilganlar bazasi"ni yuklash (fayldan);
  - ish rejimini tanlash:
    - o'zbekcha-inglizcha;
    - inglizcha-o'zbekcha.
  - tanlangan ish rejimiga mos ravishda so'zni boshqa tildagi tarjima variantlari chop etilsin.
- Berilganlar bazasi STL kutubxonasidagi map konteyner sinfi vositasida yaratilsin.
11. STL kutubxonasidagi konteyner sinflardan foydalangan holda foydalanuvchi va robot (kompyuter) o'rtasida "xoch va nol" o'yini amalga oshirilsin.
12. STL kutubxonasidagi konteyner sinflardan foydalangan holda "15 o'yini" boshqotirmasi amalga oshirilsin. Nomerlarning boshlang'ich joylashuvi tasoddiy bo'lsin. Maqsad holatga olib keluvchi kataklarni ko'chirish ketma-ketligi ikki xil rejimda ko'rsatilsin: uzluksiz (qadamlarni ma'lum bir pauzada

berish orqali); qadamba-qadam (klaviatura tugmasini bosilishiga bogʻliq holda).

13. Saylovda qatnashadigan nomzodlar roʻyxatini shakllantiruvchi ilova yaratilsin. Har bir nomzod oʻzi haqida quyidagi maʼlumotlarni taqdim etadi: <familiya va ismu shariflari>, <tugʻilgan yili>, <tugʻilgan joyi>, <mashhurlik indeksi>.

Quyidagilar masala ostilari bajarilsin:

- daʼvogarni nomzodlar roʻyxatiga kiritish. Nomzodning mashhurlik indeksi quydagicha aniqlanadi:

- prezident tomonidan qoʻllab-quvvatlanadi -70;
- oppozitsiydagi partiya tomonidan qoʻllab-quvvatlanadi -15;
- oppozitsiydagi nomzod boʻlib, 1-nomerli nomzod foydasiga oʻz nomzodini qaytarib oladi – 10;
- boshqa holatlarda – 5.

- nomzod talabi boʻyicha uni roʻyxatdan oʻchirish;

- saylov uchun roʻyxatni shakllantirish va chop etish.

Berilganlani saqlash uchun STL kutubxonasidagi `priority_queue` konteyner sinfidan foydalailsin. Talab qilingan amallarni bajarish uchun sinfda < (kichik) amali mos ravishda aniqlanishi zarur boʻladi. Saylov uchun roʻyxatni shakllantirish va chop etish navbatdan tanlash orqali amalga oshiriladi.

14. STL kutubxonasidagi `map` konteyner sinfidan foydalangan holda avtobus parki ishlashini modellashtirish masalasi echilsin (26.4-masalaga qaralsin). Kalit sifatida <avtobus nomeri> olinsin.
15. STL kutubxonasidagi `multimap` konteyner sinfidan foydalangan holda avia-biletarga talablarni hisob qilish masalasi yechilsin (26.5-masalaga qaralsin). Kalit sifatida <uchish manzili> olinsin.
16. STL kutubxonasidagi `vector` konteyner sinfidan foydalangan holda kutubxonadagi kitoblar hisobini olib boruvchi ilova tuzilsin (26.6-masalaga qaralsin).
17. STL kutubxonasidagi `list` konteyner sinfidan foydalangan holda “*Yon daftar*” masalasi yechilsin (26.7-masalaga qaralsin).
18. STL kutubxonasidagi `set` konteyner sinfidan foydalangan holda kvartira almashirish va almashish variantini izlash masalasi yechilsin (26.8-masalaga qaralsin).
19. STL kutubxonasidagi `set` konteyner sinfidan foydalangan holda temir yoʻl vokzalidagi poyezdlar harakati boʻyicha avtomatlashtirilgan tizim yaratilsin (26.9-masalaga qaralsin).
20. STL kutubxonasidagi `set` konteyner sinfidan foydalangan holda oʻzbekcha-inglizcha va inglizcha-oʻzbekcha lugʻat yaratilsin (26.10-masalaga qaralsin).

## 27. Grafika

C++ Builder muhitida grafik chizishlar bo'yicha ko'rsatmalar va namunaviy masalalar 3-ilovada keltirilgan.

### *Amaliy topshiriqlar*

- Quyidagi funksiyalar grafiklari  $x \in (-3,3)$  oraliq uchun qurilsin:  
a)  $y = 3x^2$ ;      b)  $y = 6x^2 + 3x$ ;      d)  $y = x^3 - 2x^2 + 3$ .
- Ekkranda gorizontaal bo'yicha o'zgarmas tezlikda soha chegaralariga urilib qaytuvchi shar hosil qilinsin.
- Quyidagi funksiyalarning aniqlanish sohasi tekshirilsin va grafigi qurilsin:  
a)  $y = \frac{1}{x}$ ;      b)  $y = x + \frac{3}{x} - 2$ ;      d)  $y = \frac{1}{x^2 + 3x + 1}$ ;  
e)  $y = 3 - \frac{2}{x} - \frac{3}{x^2}$ ;      g)  $y = \frac{x^2}{x^2 - 3x + 1}$ ;      f)  $y = \frac{1}{x^2} + 2x + 1$ .
- Ekkranda gorizontaal bo'yicha o'zgarmas tezlikda chapdan o'ngga siklik harakat qiluvchi satr (reklama) hosil qilinsin.
- Funksiya grafigi chizilsin:  $|y| + \frac{1}{|y|} = |x| + \frac{1}{|x|}$ ,  $|x| < 2$ .
- Parametrik tenglamalar bilan aniqlangan chiziqlar chizilsin:  
a) markazi koordinata boshida bo'lgan  $r$  radiusli aylana:  
$$x = r \cdot \cos(t), \quad y = r \cdot \sin(t), \quad t \in (0, 2\pi).$$
  
b) katta va kichik yarim o'qlari koordinata o'qlariga parallel va radiuslari mos ravishda  $r_1$  va  $r_2$  bo'lgan ellips:  $x = r_1 \cdot \cos(t)$ ,  $y = r_2 \cdot \sin(t)$ ,  $t \in (0, 2\pi)$ .  
d) paskal chig'anoq'i:  
$$x = a \cdot \cos^2(t) + b \cdot \cos(t), \quad y = a \cdot \cos(t) \cdot \sin(t) + b \cdot \sin(t),$$
  
$$a > 0, b > 0, t \in (0, 2\pi), b \geq 2a, a < b < 2a, a > b$$
  
hollarda ko'rilsin.  
e) kardioida:  $x = a \cdot \cos(t + \cos(t))$ ,  $y = a \cdot \sin(t + \cos(t))$ ,  $a > 0, t \in (0, 2\pi)$ .
- Koordinatalari quyidagi tengsizlik va tengsizliklar sistemasini qanoatlantiruvchi nuqtalar ekkranda yoritilsin.  
a)  $|y| + 2|x| \leq x^2 + 1$ ;      b)  $x^2 + y^2 \leq 2(|x| + |y|)$ ;  
d)  $4 \leq x^2 + y^2 \leq 2(|x| + |y|)$ ;      e)  $2y - x^2 \leq 4, x^2 + y^2 \geq 0$ .
- Mart oyi kunlarining haroratlarini ko'rsatuvchi  $t_1, t_2, \dots, t_{31}$  butun sonlar berilgan. Harorat grafigini quring. Nol haroratga mos keluvchi gorizontaal chiziqdan yuqorida va pastda to'g'ri chiziq kesmalari har xil ranglarga bo'yalsin.

9. Butun  $x, y, r, m, a, b$  sonlar berilgan. Markazi  $(x, y)$  nuqtada bo'lgan  $r$  radiusli aylana va yuqori chap uchi  $(m, n)$  nuqtada, bo'yi  $a$  va eni  $b$  bo'lgan to'g'ri to'rtburchak chizilsin. Aylana va to'rtburchak markazlarini tutashtiruvchi kesma yasalsin.

10. Butun  $n$  va  $r$  sonlari berilgan.  $r$  radiusli aylanaga ichki chizilgan ko'pburchak uchlari bo'lgan  $n$ -ta nuqta qurilsin. Har bir nuqta qolgan  $n-1$  nuqtalar bilan tutashtirilsin. Nuqtalarning koordinatasi quyidagi formulalar bilan berilgan:

$$x_t = r \cdot \cos\left(\frac{2\pi t}{n}\right), \quad y_t = r \cdot \sin\left(\frac{2\pi t}{n}\right) \quad (t = 1, 2, \dots, n).$$

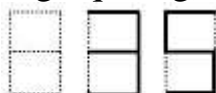
Tutashtiruvchi chiziqlarni takroran o'tkazmaslik uchun  $t$  nomerli nuqtani faqat  $t < j$  shartni qanoatlantiruvchi  $j$  nuqtalar bilan tutashtirish kerak.

11. Berilgan haqiqiy  $a, b, c$  va  $d$  ( $0 < a < b < c < d$ ) qiymatlar uchun quyidagi funksiyalar grafiklari qurilsin:

$$a) \quad y(x) = \begin{cases} 1 - \frac{b-x}{b-a}, & a \leq x \leq b; \\ 1 - \frac{x-b}{c-b}, & b < x \leq c; \\ 0, & \text{qolgan holatlarda.} \end{cases} \quad b) \quad y(x) = \begin{cases} 1 - \frac{b-x}{b-a}, & a \leq x \leq b; \\ 1, & b < x \leq c; \\ 1 - \frac{x-c}{d-c}, & c < x \leq d; \\ 0, & \text{qolgan holatlarda.} \end{cases}$$

12. Butun  $n$  va  $r$  sonlari berilgan. Tomonlarning uzunligi  $r$  bo'lgan kvadrat yasalsin. Kvadratning har bir uchida bittadan va har tomonida  $n-1$  tadan nuqtalar joylashtirilsin. Tomonlardagi o'zaro qo'shni nuqtalar orasidagi masofa bir xil bo'lib,  $r/n$  soniga teng. Bu usul bilan hammasi bo'lib  $4n$  ta nuqta quriladi va ularni  $1, \dots, 4n$  sonlari bilan nomerlash mumkin (nomerlash kvadratning yuqori chap uchidan boshlanib, soat millari yo'nalishida amalga oshiriladi). Tartib nomeri  $t$  bo'lgan har bir nuqtani  $j > t$  shartni qanoatlantiruvchi ( $j-1 < 4n$ ) sonidan kichik bo'lgan Fibonachchi sonlarini beradigan  $j$  nomerli nuqtalar bilan tutashtirilsin.

13. Ekranida ishlayotgan elektron soat tasviri hosil qilinsin. Soatdagi raqamlar ko'rinishi oddiy elektron soatlaridagi kabi yetti segmentli (bo'lakli) qolipga mos kelishi kerak (rasmga qarang).



14. Ekran tekisligida quyidagi shartlar ostida aylanuvchi kesma yasalsin:

- kesma o'rtasiga nisbatan;
- birorta uchiga nisbatan;
- kesmani  $1/3$  nisbatda bo'luvchi nuqta atrofida.



15. Ikkita ko'rsatkich-kesmalarni (soat millarini) qo'zg'almas nuqta atrofida aylanishi amalga oshirilsin. Bunda kesmalar birining (kattasining) bir marta to'liq aylanib chiqishiga ikkinchisining  $1/12$  aylanishiga mos keladi
16. Ekran tekisligida quyidagi shartlar bo'yicha soat millari harakati bo'yicha (yoki aksincha) aylanuvi to'g'ri burchakli uchburchak tasvirlansin:  
 a) markazi atrofida;                      b) uchlaridan biri atrofida.
17. Ekranida aylana o'lchami va uning joylashuvi boshqarilsin. Boshlang'ich holatda aylana markazi ekran markazida joylashsin va radiusi  $r$  deb hisoblansin. Boshqarish klaviaturaning quyidagi tugmalari yordamida amalga oshirilsin. Agar '+' tugmasi bosilsa, aylana radiusi 5 pikselga kattalashsin, agar '-' tugmasi bosilsa, aylana radiusi besh pikselga kichraysin. Klaviaturaning yo'nalish tugmalari bosilsa, aylana mos yo'nalishda 5 pikselga ko'chsin.
18.  $h$  balandlikdan  $v$  boshlang'ich tezlik bilan gorizontaal yo'nalishda otilgan jismning Yer sharining tortish kuchi ta'siri ostidagi harakati tasvirlansin. Havо qarshiligi hisobga olinmasin.
19. Yadro atrofida  $k$  ta elektronning berilgan elliptik orbita bo'ylab o'zgarmas tezlikdagi harakati tasvirlansin. Elektronlar harakati koordinata o'qlarini  $t$  radianga burish asosida amalga oshiriladi:  

$$(x, y) \rightarrow (x \cdot \cos t + y \cdot \sin t, -x \cdot \sin t + y \cdot \cos t)$$
20. Ko'rinmas aylana bo'ylab harakat qiluvchi to'g'ri chiziq kesmasining tasviri hosil qilinsin.
21. Ekranida gorizontaal yo'nalishda o'zgarmas  $v$  tezlikda dumalab ketayotgan g'ildirak tasvirlansin.
22. Ekranida berilgan fokusli, qavariq linza uchun sham tasvirining oraliq masofaga bog'liq ravishda kattalashishi (kichrayishi) ko'rsatilsin. Biror obyekt tasvirini kattalashtirish (kichraytirish) quyidagi formula yordamida koordinata o'qlarini cho'zish (siqish) orqali amalga oshiriladi:  

$$(x, y) \rightarrow (s_x \cdot x, +s_y \cdot y),$$
  
 bu yerda  $s_x, s_y$  – mos ravishda OX va OY o'qlar bo'yicha cho'zish (siqish) kattaligi.
23. Ekranida chuqurchalari bo'lmagan bilyard taxtasi ustidagi shar harakati tasvirlansin.
24. "Samolyotni urish" o'yini. Ekraning yuqori qismida gorizontaal uchib borayotgan samolyotni urib tushirish kerak. Urish qurilmasi ekran pastida joylashgan, u oldinga va orqaga harakat qilishi mumkin. Otilgan o'qning harakatida Yerning tortish kuchi inobatga olinsin.

25. Post mashinasini ishlashi, yani ko'rsatmalar ketma-ketligiga (ilovaga) mos ravishda lenta bo'ylab karetkaning harakati ko'rsatilsin.

Post mashinasi quyidagilardan tashkil topadi:

- bir xil kataklardan tashkil topgan cheksiz lenta. Kataklar bo'sh bo'lishi (0 yoki bo'sh) yoki nishonlangan bo'lishi (1 yoki qandaydir belgi) mumkin;

- lentada chap yoki o'ngga bitta katakka harakatlanadigan hamda katakda nishon bor yoki yo'qligini tekshira oladigan va uni o'chiradigan va yozadigan karetk.

Post mashinasining ayni paytdagi holati lentaning holati va karetk joylashuvi bilan tavsiflanadi. Lentaning holati – bu qaysi kataklar bo'sh, qaysilari nishonlanganligi bilan aniqlanadi. Qadam – bu karetkaning chapga yoki o'ngga bitta katakka surilishidir. Lenta holati ilova bajarilishi natijasida o'zgarishi mumkin.

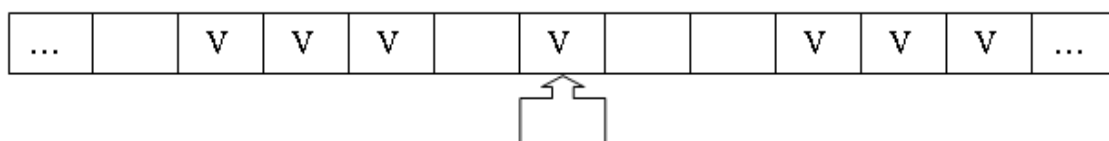
Karetk buyruq satrlaridan tashkil topgan programma orqali boshqariladi. Har bir buyruq "i.Kj" ko'rinishga ega bo'lib, bu yerda i- buyruq (satr) nomeri, K - karetk bajaradigan ish, j - navbatdagi buyruq nomeri (o'tish).

Post mashinasida jami oltita buyruq mavjud:

- 1) Vj - katakka nishon qo'yish va j-satrdagi buyruqqa o'tish;
- 2) Xj - katakdagi nishonni o'chirish va j-satrdagi buyruqqa o'tish;
- 3) <j - karetkani chapga surish va j-satrdagi buyruqqa o'tish;
- 4) >j - karetkani o'ngga surish va j-satrdagi buyruqqa o'tish;
- 5) ?j<sub>1</sub>;j<sub>2</sub> - agar katakda nishon bo'lmasa, j<sub>1</sub>-satr buyruqqa, aks holda j<sub>2</sub>-satrdagi buyruqqa o'tish;
- 6) ! - ilova tugashi (stop).

To'xtash buyrug'ida o'tish yo'q.

Quyidagi rasmda Post mashinasining ko'rinishiga misol keltirilgan.



26. "Quyvon va bo'ri" masalasi. Chegaralangan to'rtburchakli sohada bo'ri va quyvonning boshlang'ich joylari mos ravishda  $(x_b, y_b)$  va  $(x_q, y_q)$  nuqtalar bilan, ularning vaqt birligidagi ko'chishlari  $s_b$  va  $s_q$  bilan berilgan.

Soha bo'ylab bo'rining quyvonni quvlashdagi izi chizilsin.

Har bir vaqt birligida bo'ri va quyvonning harakati quyidagicha amalga oshiriladi:

- bo‘ri quyon tomonga  $s_b$  masofaga ko‘chadi;
- quyon bo‘ridan uzoqlashadigan tomonga  $s_q$  masofaga ko‘chadi.

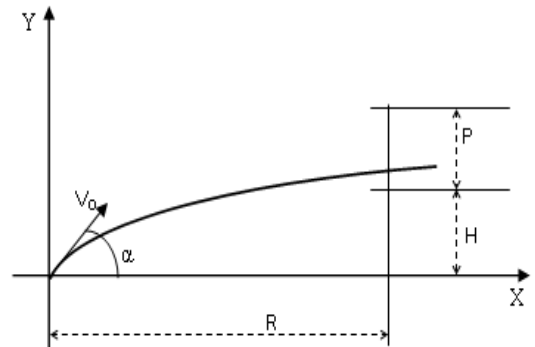
Bo‘ri va quyon harakatida soha chegarasini (to‘siqlarni) inobatga olish kerak bo‘ladi, ya‘ni ular to‘siqlardan “qochishi” zarur bo‘ladi.

Quvlash jarayoni bo‘rining navbatdagi ko‘chishida quyon koordinatasini kesib o‘tganda yoki oldindan ko‘rsatilgan  $T$  vaqtdan keyin tugaydi.

27. To‘pdan  $\alpha$  burchak ostida va  $V_0$  boshlang‘ich tezlikda otilgan snaryad traektoriyasi quyidagi tenglama bilan beriladi:

$$x = V_0 t \cos \alpha ; y = V_0 t \sin \alpha - gt^2 / 2 ,$$

bu yerda  $g = 9,8 \text{ m/s}^2$  - erkin tushish tezlanishi,  $t$  - vaqt. Berilgan  $V_0$  tezlikda va  $\alpha$  burchak ostida otilgan snaryad



$R$  masofadagi yerdan  $H$  balandlikdagi va o‘z balandligi  $P$  bo‘lgan nishonga otilgan snaryad harakati chizilsin. Agar snaryad nishonga tegsa, uning portlashi namoyon etilsin.

## Foydalanilgan adabiyotlar

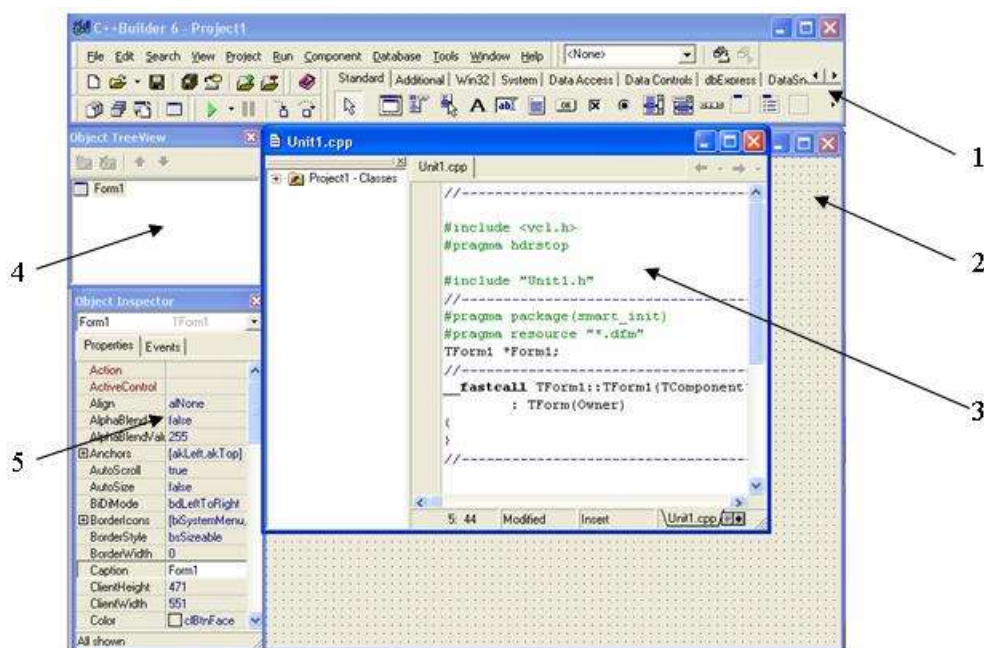
1. Абрамов С.А., Гнездилова Г.Г., Капустина Е.Н., Селюн М.И. Задачи по программированию.-М.: “Наука”, 1988.-224с.
2. Павловская Т.С. Щупак Ю.С. С/С++. Структурное программирование. Практикум.-СПб.: “Питер”,2002.-240с.
3. Павловская Т.С. Щупак Ю.С. С++. Объектно- ориентированное программирование. Практикум.-СПб.: “Питер”,2005.-265с.
4. Дейтел Х.М., Дейтел П.Дж. Как программировать на С++: “Бином-Пресс”, 2008, 1454с.
5. Пильшиков В.Н. Сборник упражнений по языку Паскаль: Учеб.пособие для вузов.-М.: “Наука”. 1986, 160с.
6. Мадрахимов Ш.Ф., Гайназаров С.М. С++ тилида программалаш: Услубий кўлланма.-ЎзМУ, Тошкент, 2009.-196 б.
7. Informatika va programmalash. O‘quv qo‘llanma. Mualliflar: A.A.Xaldjigitov, Sh.F.Madrahimov, U.E.Adamboev, O‘zMU, 2005 yil, 145 bet.
8. Паскал тилида программалаш бўйича масалалар тўплами. Ўқув кўлланма. Муаллифлар: А.А.Холджигитов, Ш.Ф.Мадрахимов, А.М.Икромов, С.И.Расулов, ЎзМУ, 2005 йил, 94 бет.

# 1-ilova

## C++ Builder muhiti haqida qisqacha ma'lumot

C++ Builder programmalash muhiti vizual ravishda monitor ekranida bir vaqtda ochilgan bir nechta darchalar ko'rinishida amalga oshiriladi. Darchalarning soni, joylashuvi va ko'rinishi ayni paytdagi ilova tuzuvchi zaruratidan kelib chiqqan holda o'zgarishi mumkin.

Ekranida C++ Builder muhiti quyidagi darchalar majmuasi ko'rinishida paydo bo'ladi:



Yuqorida keltirilgan rasmda:

1. Asosiy menyu paneli;
2. Foydalunuvchi formasi(shakli);
3. Ilova (ilova) kodini kiritish maydoni;
4. Obyektlarning o'zaro munosabatini daraxt ko'rinishini ifodalovchi darcha;
5. Obyektlar inspektori darchasi.

**Asosiy menyu paneli** doimiy ravishda ekranda ko'rinib turadi va ilova yaratish jarayonini boshqarish uchun xizmat qiladi. Unda loyihani boshqarish uchun zarur bo'lgan barcha vositalar mavjud. Piktogrammalar asosiy menyuning nisbatan ko'p murojaat qilinadigan buyruqlariga murojaat qilishni yengillashtiradi. Komponentlar menyusi orqali ilova tuzuvchi tomonidan formaga joylashtiriladigan qandaydir vizual elementni (komponentni) tavsiflovchi C++ Builder muhitining standart servis ilovalar

majmuasiga murojaat qilish amalga oshiriladi. Har bir komponenta ilova tuzuvchi tomonida beriladigan o'zining ma'lum bir xossalari (parametrlariga) ega. Masalan, rangi, darcha sarlavhasi, tugmadagi yozuv, shrift o'lchami, turi va hokazo.

**Obyektlar inspektori darchasi** tanlangan komponentalar xossalarini o'zgartirish uchun mo'ljallangan bo'lib, ikkita sahifaga ega. Properties (Xossalar) sahifasida komponentalar xossalarini o'zgartirishga, Events (hodisalar)- u yoki bu hodisaga komponentaning aks ta'sirini belgilash uchun xizmat qiladi.

**Foydalunuvchi formasi (shakli)** o'zida ilovaning Windows-darcha loyihasini ifodalaydi. Bu darchada ilovani yaratish jarayonida zarur komponentalar joylashtiriladi. Shuni qayd etish kerakki, komponentalar loyihalash bosqichida formada qanday ko'rinishda joylastirilgan bo'lsa, xuddi shu holda ilova ishlaganda namoyon bo'ladi.

**Ilova (ilova)** kodini **kiritish maydoni** ilova matnini ko'rish, yozish va tahrirlashga mo'ljallangan. Birinchi yuklanganda bu darchada ilovaning Windows darcha ko'rinishida normal amal qilishi uchun zarur bo'lgan minimal opertorlarni o'ziga olgan matn bo'ladi.

C++ Builder muhitida ilova, forma bilan bog'liq qandaydir hodisa ro'y berganda qilinishi kerak bo'lgan algoritmlarni yozish asosida tuziladi (masalan, sichqoncha ko'rastkichi bilan tugmani "*bosganda*" yuzaga keladigan OnClick hodisasi yoki formani yaratishdagi OnCreate hodisasi). Formadagi qayta ishlanadigan har bir hodisa uchun (Events sahifasi yordamida) ilova matnida funksiya tashkil etiladi va uning '{' va '}' belgilar bilan chegaralangan sohasiga (tanasiga) ilova tuzuvchi tomonida C++ tilida zarur algoritm yoziladi.

C++ Builder muhitida ikki xil usulda ilova yaratish mumkin:

- ekranning matn rejimida amal qiluvchi va buyruq satridagi interfeysga ega konsol ilovalar;
- vizual komponentlar asosida yaratiluvchi Windows ilovalari. Ular ishga tushganda ekranda forma ko'rinishida namoyon bo'ladi.

### **C++ Builder muhitining konsol regimida ishlash**

C++ Builder konsol rejimida amal qiluvchi ilovalarni yaratish uchun quyidagi amallar ketma-ketligi bajariladi:

Bosh menyu orqali quyidagi punktlar ketma-ketligi tanlanadi:

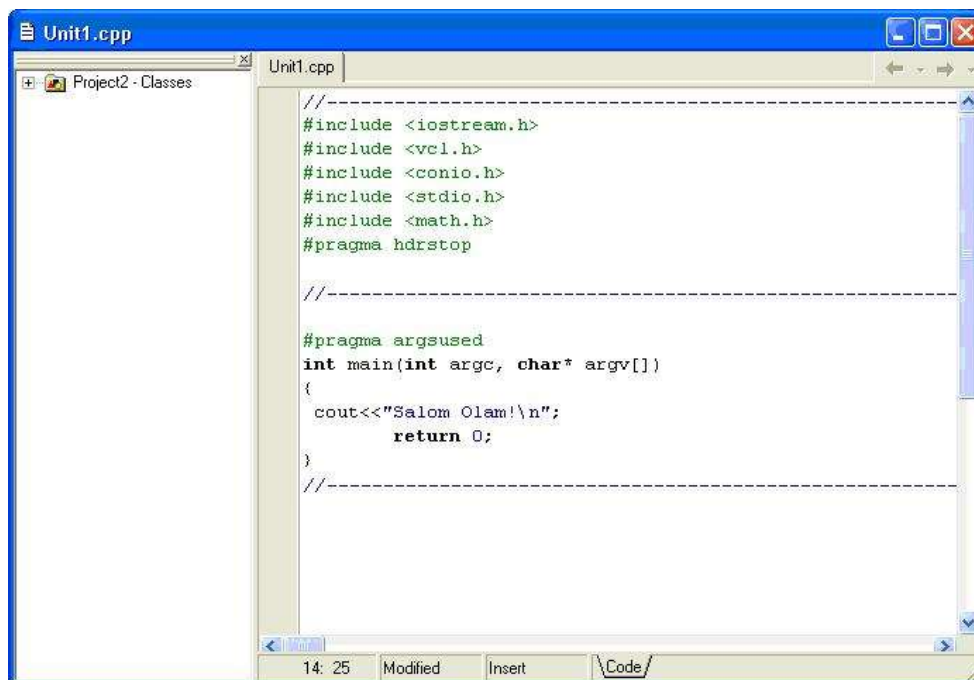
File → New → Other → Consol wizard;

Yaratilayotgan ilovaning qanday turi (C yoki C++), ishlatiladigan kutubxonalar (VCL, CLX), ko'poqimlilik, konsol ilova ekanligi qayd qilish

yo‘li bilan aniqlanadi. Konsol ilovani yaratishni tanlash bosqichlari quyidagi rasmlarda keltirilgan.



Quyidagi rasmda tahrir maydonida konsol ilovasi matnini terish holati ko‘rsatilgan.



Ilovani ishga tushirish uchun bosh menyudagi “Run” tugmasi bosiladi (yoki F9 tugmasi) va ilova ishlashining natijasi alohida darchada ko‘rinadi. Misol uchun quyidagi rasmda ilovaning ish natijasi keltirilgan.



Shuni qayd etish kerakki, ilova natijasi alohida shaklda chop etilishi bilan boshqaruv darhol C++ Builder muhitiga qaytiladi. Natijalar dar-chasini ekranda ushlab turish uchun sun'iy usullar qo'llaniladi (cheksiz takrorlash, klaviatura tugmasini bosishni kutish va boshqa usullar).

### **C++ tilida bajariluvchi ilova yaratish bosqichlari**

Quyida C++ tilida ilova tuzilishini tushuntirish uchun sodda ilova keltirilgan.

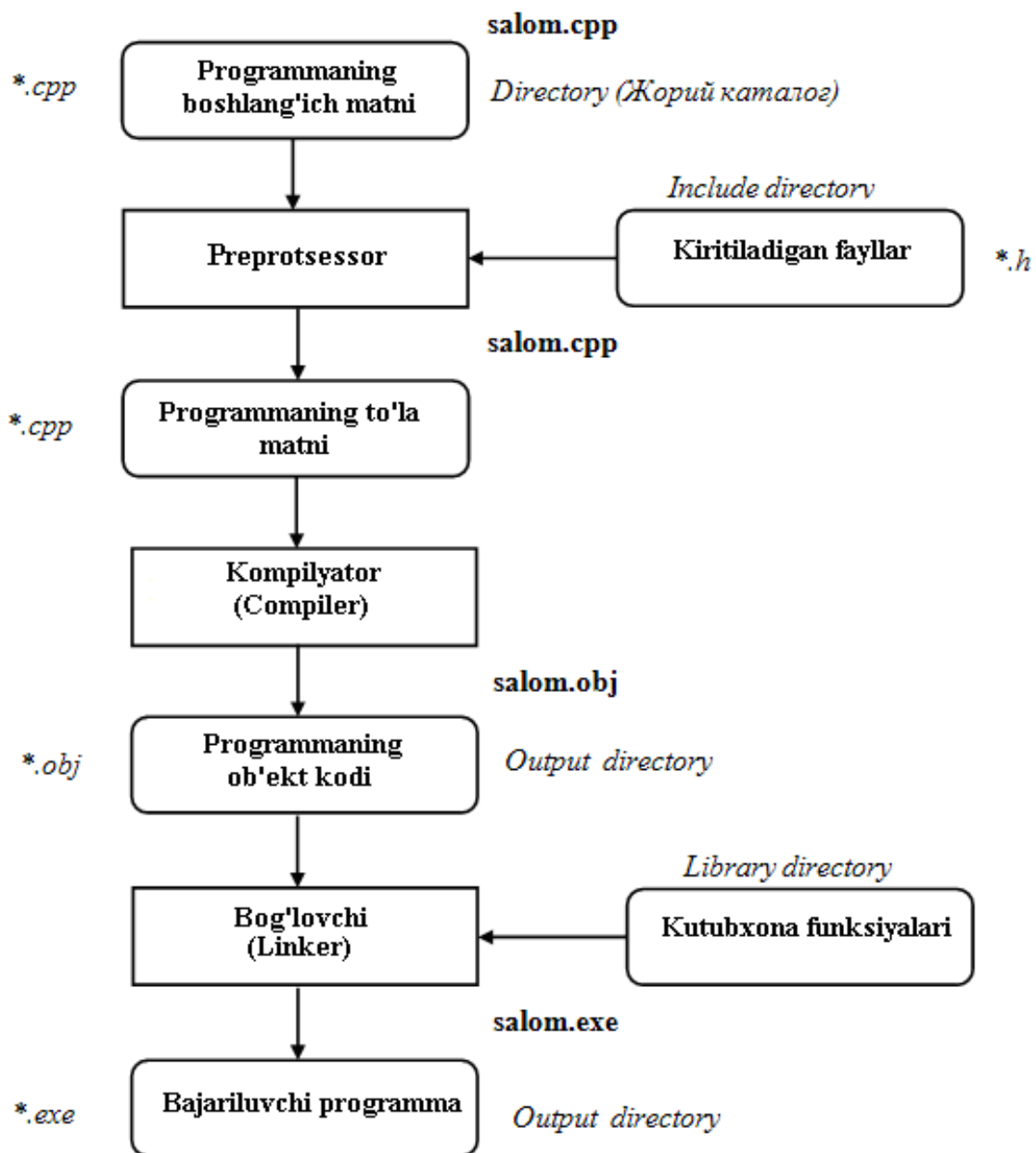
```
#include <iostream.h>           // sarlavha faylni qo'shish
int main()                       // bosh funksiya tavsifi
{                                  // blok boshlanishi
    cout<<"Salom Olam!\n";      // satrni chop etish
    return 0;                     // funksiya qaytaradigan qiymat
}                                  // blok tugashi
```

Ilovaning 1-satrida "#include" preprotssessor ko'rsatmasi bo'lib, ilova kodiga standart oqimli o'qish-yozish funksiyalari va uning o'zgaruvchilari e'loni joylashgan «iostream.h» sarlavha faylini qo'shadi (mnemonika: 'i'(input) - kiritish (o'qish); 'o'(output) - chiqarish (yozish); "stream"- oqim; 'h'(head - sarlavha). Kelishuv bo'yicha standart oqim - ekranga chiqarish hisoblanadi. Keyingi qatorlarda ilovaning yagona, bosh funksiyasi - main() funksiyasining sarlavhasi keltirilgan. Shuni qayd etish kerakki, C++ ilova sida albatta main() funksiyasi bo'lishi shart va ilova shu funksiyani bajarish bilan o'z ishini boshlaydi. Funksiya nomi oldidagi "int" kalit so'zi funksiya bajarilganda natija sifatida butun son qiymatini qaytarishi kerakligini bildiradi. Bu holat funksiyaning matematikadagi tavsifiga mos keladi. Keyingi qatordan funksiya tanasi - figurali qavsga olingan amallar ketma-ketligi keladi. Bizning holda funksiya tanasi ikkita amaldan iborat. Birinchisi, konsol rejimida belgilar ketma-ketligini oqimga chiqarish amali qo'llanilgan. Buning uchun «iostream.h» sarlavha faylida aniqlangan cout qurilmasidan foydalanilgan. Bu yerda "<<" - berilganlarni uzatish amali ("..ga joylashtir"), oqimga chiqariladigan (chop qilinadigan) ifoda sifatida satr-o'zgarimas kelgan. Ikkinchisi, funksiya o'z ishini tugatganligini anglatuvchi va undan chiqishni amalga oshiruvchi "return 0;" operatoridir. Odatda, bajarilishi normal tugagan funksiyalar 0 qiymatini qaytaradi. Shu qoidaga rioya qilgan holda ilovadagi oxirgi amalda 0 qiymati qaytariladi.

Bajariluvchi ilova ni hosil qilish uchun programa matni kompilyatsiya qilinishi kerak. Kompilyatsiya jarayonining o'zi ham ikkita bosqichdan tashkil topadi. Boshida preprotssessor ishlaydi, u matndagi kompilyatsiya direktivalarini bajaradi, xususan "#include" direktivasi bo'yicha ko'rsatilgan



kutubxonalardan C++ tilida yozilgan modullarni ilova tarkibiga kiritadi. Shundan so‘ng kengaytirilgan ilova matni kompilyatorga uzatiladi. Kompilyator o‘zi ham ilova bo‘lib, uning uchun kiruvchi ma‘lumot sifatida C++ tilida yozilgan ilova matni hisoblanadi. Kompilyator ilova matnini leksema (atomar – bo‘linmas) elementlarga ajratadi va uni leksik, keyinchalik sintaksik tahlil qiladi. Leksik tahlil jarayonida u matnni leksemalarga ajratish uchun «*probel ajratuvchisini*» ishlatadi. Probel ajratuvchisiga - probel belgisi, '\t' - tabulyasiya belgisi, '\n'- keyingi qatorga o‘tish belgisi, boshqa ajratuvchilar va izohlar kiradi.



C++ tilida bajariluvchi ilovani yaratish bosqichlari (yuqoridagi rasmga qarang):

1. Matn tahririda (odatda programmalash muhitining tahririda) ilova matni teriladi, bu faylning kengaymasi “.cpp” turida bo‘ladi, masalan “salom.cpp”;

2. Ilova matni yozilgan fayl kompilyatorga uzatiladi. Agar xatoliklar bo‘lsa, ular to‘g‘rilanadi;

3. Kompilyator tomonidan ilova matniga sarlavha fayllar kiritiladi (“#include” preprocessor ko‘rsatmasiga muvofiq);

4. Kompilyator “.obj” (“salom.obj”) kengaymali obyekt fayli hosil qiladi;

5. Komponovka (jamlovchi) yordamida obyekt faylga kutubxonalaridan zarur funksiyalari qo‘shiladi va “.exe” kengaymali bajariluvchi fayl - ilova hosil bo‘ladi (“salom.exe”);

6. Ilovani ishga tushirish uchun buyruq satrida ilova nomini terish va “Enter” tugmasini bosish yetarli.

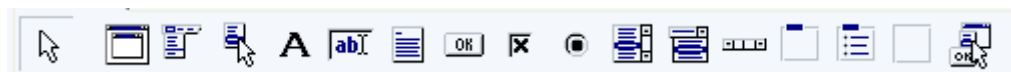
Bosqichlarda yuzaga keluvchi fayllarning nomlari boshlangich matn faylining nomi bilan bir xil bo‘ladi.

Yuqorida keltirilgan ilova bajarilishi natijasida ekranga “*Salom Olam!*” satri chop qilinadi.

### **C++ Builder muhitida vizual komponentalardan foydalanish**

Ilovalar interfeysini yaratish uchun C++Builder vizual komponentalarining keng to‘plamini taqdim qiladi. Ularning asosiylari komponentalar palitrasining Standart, Additional va Win32 varaqlarida joylashgan.

**Standart** varag‘idagi interfeys komponentalarining aksariyati Windows boshlang‘ich talqinlarida ishlatilgan interfeys komponentalaridan iborat:



Frames – freymilar;



MainMenu - asosiy menu;



PopupMenu - namoyon bo‘luvchi menu;



Label - nishon, forma sohasidagi biror joyni belgilash;



Edit - bir satrli matn tahrir;



Memo - ko‘pqatorli matn taxriri;



Button - standart tugma;









CheckBox - bog‘liqmas tanlash (belgilash) darchasi;



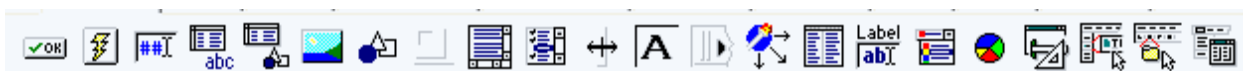
RadioButton - bog‘liqmas tanlash (yagona);


















ListBox - tanlanadigan satrlar ro‘yxati;

-  ComboBox - ochiluvchi ro'yxatga ega tanlash;
-  **ScrollBar** - soha bo'yicha harakatlantirish yo'lagi;
-  GroupBox - nomlanadigan konteyner soha;
-  RadioGroup - o'zaro bir-birini inkor qiluvchi tanlashlar guruhi;
-  Panel - konteyner soha;
-  ActionList - amallar (bog'lanadigan funksiyalar) ro'yxati.

**Additional** varag'ida komponentalar quyidagicha joylashgan:



-  BitBtn - rasmi tugma;
-  SpeedButton - tezkor murojaat tugmasi;
-  MaskEdit - qolip buyicha berilganlarni kiritish uchun bir qatorli tahrir;
-  StringGrid - satrlarning ikki o'lchamli jadvali;
-  DrawGrid - rasm joylashtirish mumkin bo'lgan jadval;
-  Image - grafik shakl joylashtiriladigan soha;
-  Shape – standart geometrik shakllar;
-  Bevel – bo'rtirilgan (o'yilgan) shaffof to'rtburchak soha;
-  ScrollBox – harakatlanishi mumkin bo'lgan darcha;
-  CheckListBox – tanlashlar (belgilashlar) ro'yxati;
-  Splitter – sohani ajratuvchi chiziq;
-  StaticText - statik (turg'un) satr;
-  ControlBar - vositalar paneli uchun konteyner;
-  ApplicationEvents - ilova hodisalari;
-  Chart – diagrammalar, grafiklar chizish vositasi.

### **Vizual komponentalarning xossalari**

Barcha vizual komponentalar uchun TControl sinfi asos hisoblanadi va u elementning o'lchami va joylashuvi, uning sarlavhasi, rangi va shunga o'xshash parametrlaridan iborat asosiy funksional atributlarni ta'minlaydi. TControl sinfi vizual komponentalar uchun umumiy bo'lgan xossalar,

hodisalar va metodlarni o‘z ichiga oladi. Vizual komponentalarni ikkita katta guruhga ajratish mumkin:

1. To‘g‘ri to‘rtburchakli boshqaruv elementlari;
2. To‘g‘ri to‘rtburchakli bo‘lmagan boshqaruv elementlari.

To‘g‘ri to‘rtburchakli boshqaruv elementi o‘zida ma’lum bir maqsad uchun aniqlangan maxsus to‘g‘ri to‘rtburchakni ifodalaydi. Bu elementlarga misol tariqasida boshqaruv tugmalarini, tahrir maydonlarini, harakatlanish yo‘laklarini ko‘rsatishimiz mumkin. Ular uchun TWinControl asos sinf hisoblanadi.

To‘g‘ri to‘rtburchak elementlari qiymat kiritish fokuslarini ilishi mumkin. Elementni fokus olganligi ikki xil usulda ko‘rsatiladi:

1. Tahrir kursori yordamida;
2. To‘g‘ri to‘rtburchak orqali.

Matn tahrirlari bo‘lgan Edit va Memo komponentalari o‘z sohasida tahrir kursori (matn kursori) paydo bo‘lishi orqali fokus (boshqaruvni) olganligini bildiradi.

Mantlarni tahrirlash bilan bog‘liq bo‘lmagan komponentalarda qora punktir chiziqli to‘g‘ri to‘rtburchak paydo bo‘lishi uning fokus olganligini anglatadi. Masalan, Button tugmasi fokus olganda sarlavha atrofida to‘g‘ri to‘rtburchak paydo bo‘ladi, ListBox komponentasida esa ro‘yxatdagi ayni paytda tanlangan satrni ajratilgan holda (aksariyat hollarda ko‘k fonda) ko‘rsatishi boshqaruvni olganligini bildiradi. Bulardan tashqari, to‘g‘ri to‘rtburchak boshqaruv elementlari konteyner sifatida o‘z ichida boshqa boshqaruv elementlarini olishi mumkin. Bunday hollarda boshqaruv elementi o‘z ichidagilarga “ota” hisoblanadi.

To‘g‘ri to‘rtburchak bo‘lmagan boshqaruv elementlari TGraphicControl sinfining avlodlari hisoblanadi. Bu guruh elementlari qiymat kiritish fokusini olmaydi va interfeys elementlari uchun “ota” bo‘la olmaydi. To‘g‘ri to‘rtburchak bo‘lmagan boshqaruv komponentalarining afzalligi – ularni nisbatan kam resurs talab qilishida.

Xossalar ilovalarni yaratishda komponentalar tashqi ko‘rinishi va amal qilishini boshqarish imkonoyatini beradi. Odatda komponenta xossasining qiymatlari ilovalarni yaratish vaqtida Object Inspector yordamida amalga oshiriladi. Keltiriladigan misollarda tushunarli bo‘lishi uchun xossalarga qiymat berish operatori yordamida amalga oshiriladi. Shuni ta’kidlab o‘tish kerakki, komponentalar barcha xossaga ega bo‘lmasligi mumkin. Masalan, Edit tahriri Caption xossasiga ega emas, Label yozuvi ReadOnly xossasiga ega emas va hakoza.

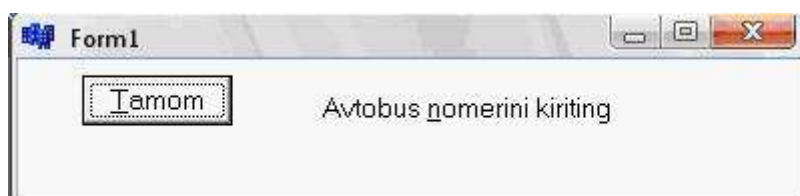
Quyida komponentalarda uchraydigan asosiy xossalar izohi keltirilgan.

**Caption** xossasi komponenta sarlavhasini yozish satrini o‘z ichiga oladi. Sarlavha satridagi ayrim belgilar tagiga chizilgan bo‘lishi mumkin. Ular tezkor murojaat tugmalarining kombinatsiyasini bildiradi. Ko‘rsatilgan belgini <Alt> tugmasi bilan bir vaqtda bosilishi shu sarlavhadagi komponentada sichqoncha tugmasini bosish bilan bir xil amalni yuzaga keltiradi. Tezkor murojaat belgisini, shu belgi oldiga ‘&’ belgisini qo‘yish orqali belgilanadi, masalan:

Label1->Caption="Avtobus &nomerini kiriting";

Button1->Caption="&Tamom";

Caption xossasining qiymatlarini Object inspector darchasi Properties varag‘innig mos qatoridagi satr maydonida kiritish ham mumkin. Ilova ko‘rinishi quyidagicha bo‘ladi:



**Align** xossasi komponentani u joylashgan konteyner ichidagi joylashuv holatlarini aniqlaydi. Aksariyat hollarda konteyner sifatida Form formasi yoki Panel paneli keladi.

Align xossasi quyidagi qiymatlarning birini qabul qilishi mumkin:

alNone - to‘g‘rilash amalga oshirilmaydi. Komponenta ilovani yaratish paytida qayerga joylashtirilgan bo‘lsa, shu joyda qoladi;

alTop - komponenta konteynerning yuqori qismiga ko‘chiriladi, komponenta balandligi o‘zgarmaydi, eni esa konteyner eniga teng bo‘ladi;

alBottom - alTop xossasiga o‘xshash, faqat komponenta konteyner pastiga joylashadi;

alLeft - komponenta konteynerning chap tomoniga ko‘chadi, eni o‘zgarmaydi, bo‘yi esa konteyner bo‘yiga tenglashadi.

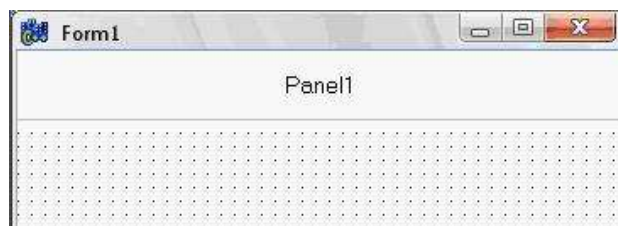
alRight - alLeft xossasiga o‘xshash, faqat komponenta konteynerning o‘ng tomoniga ko‘chadi;

alClient - komponenta konteynerni to‘la egallaydi.

Misol. Panelni formaga nisbatan to‘g‘rilash.

Panel1->Align=AlTop;

Natijada Panel1 paneli Form1 formaning yuqori qismiga joylashadi.



**Color** komponenta fonining rangini aniqlaydi. Color xossasining qiymati 4 baytli 16 lik sanoq sistemasidagi son bo'lib, uning katta bayti rang palitrasini aniqlaydi (odatda \$00), kichik uchta bayti qizil, yashil va ko'k ranglarning RGB intensivligini aniqlaydi. Bu ranglarni turli nisbatdagi aralashmasidan natijaviy rang hosil bo'ladi. Masalan, \$000000 - qora, \$FFFFFF - oq, \$0000FF - qizil, \$00FF00 - yashil, \$FF0000 - ko'k rangni beradi. Ranglar qiymatini konstantalar orqali berish ancha qulay. Masalan, clAqua - och ko'k (\$FFFF00), clBlack - qora (\$000000) va h.k.

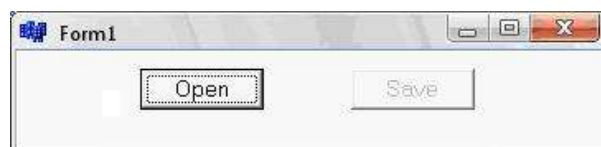
**Ctrl3D** xossasi boolean turida bo'lib, vizual komponenta ko'rinishini aniqlaydi. Agar Ctrl3D qiymati false bo'lsa komponenta ikki o'lchamli tasvir ko'rinishida, agar true bo'lsa - uch o'lchamli tasvirlanadi (kelishilgan holdagi ko'rinish).

**Cursor** xossasi sichqoncha ko'rsatgichining ko'rinishini aniqlab beradi. Bu ko'rinishlar rang barang, ular ichida eng asosiylari quyidagi konstantalar bilan beriladi:

- crDefault - sichqoncha ko'rsatgichi kelishuv bo'yicha (odatda strelka);
- crNone - ko'rsatkich ko'rinmaydi;
- crArrow - ko'rsatkich strelka ko'rinishida;
- crCross - ko'rsatkich xoch ko'rinishida;
- crHourGlas - ko'rsatkich qum soati ko'rinishda.

**Enabled** xossasi boolean turida bo'lib komponentani faolligini, ya'ni sichqonchadan yoki klaviaturadan kelayotgan signallarga aks ta'sir bera olishini aniqlaydi. Agar xossa true (kelishilgan holat qiymati) qiymatga ega bo'lsa, komponenta faol hisoblanadi. Komponentaning faol bo'lmagan holatini sarlavhaning yoki matnning oqargan rangda ekanligi bildiradi. Ayrim hollarda qandaydir komponenta o'chirilgan (blokirovka qilingan) bo'ladi, agar u bilan bog'liq qandaydir amalni bajarishni iloji bo'lmasa. Masalan, Save (Button2) tugmasi xujjatni saqlash amalini bajaradigan bo'lsin. Agar saqlanadigan xujjatning o'zi bo'lmasa, saqlash amalining ma'nosi yo'q. Bunday hollarda tugma faol bo'lmagan holatga o'tkaziladi:

Button2->Enabled=false;



**Font** xossasi vizual komponenta akslanuvchi matn shriftini aniqlaydi. O‘z navbatida TFont sinfi shrift parametrlarini boshqarishga imkon beruvchi xossalarga ega. Bularning ichida asosiylari quyidagilar:

Name - shrift nomini aniqlaydi;

Size - punktlarda shrift o‘lchamini beradi;

Style – shrift stilini beradi;

Color – matn rangini boshqaradi.

Masalan, Label1 yozuvining rangini berish:

Label1->Font->Color=clGreen;

Label1->Color=clWhite;

Label1 sarlavhasi yashil fonddagi oq rangdagi matn ko‘rinishida bo‘ladi.

**Height** va **Width** xossalari komponentaning mos ravishda vertikal va gorizontal o‘lchamlarini piksillarda beradi.

**Left** va **Top** xossasi komponentaning u joylashgan konteynerga (forma yoki panelga) nisbatan koordinatalarini aniqlaydi. O‘z o‘rnida forma ham komponenta va uning koordinatalari ekranning chap yuqori burchagiga nisbatan aniqlanadi.

**Hint** xossasi kursor komponenta sohasida va bir necha soniya harakatsiz holatda bo‘lganda ekranga yordamchi matnni chiqaradi. Yordamchi matn sariq fonda chiqadi va komponentaning ishlatilish maqsadini qisqacha yoritadi. Yordamchi matn ekranda paydo bo‘lishi uchun boolean turidagi ShowHint xossasining qiymatini true deb aniqlash zarur.

**PopupMenu** xossasi sichqonchani o‘ng tugmasini bosganda paydo bo‘luvchi lokal menyuni aniqlaydi. Menyu paydo bo‘lishi uchun komponentaning (sohaning) AutoPopupMenu xossasiga (bool turidagi) true qiymatini berish kerak. Kelishuv bo‘yicha uning qiymati false bo‘ladi.

**TabOrder** xossasi konteynerdagi komponentalarning fokus olish tartibini aniqlaydi (Tab tugmasi bosilganda), ya'ni komponentalar “aylanib” chiqish ketma - ketligini aniqlaydi. Kelishuv bo‘yicha bu ketma-ketlik formani ko‘rinishini yaratishda komponentalarni konteynerga joylashtirish tartibiga mos keladi: birinchi komponentaning TabOrder xossasining qiymati 0, ikkinchisniki - 1 va hokazo. Tartibni o‘zgartirish uchun komponentaning TabOrder xossasiga zarur qiymatni berish kerak. Har bir konteyner boshqalariga bog‘liq bo‘lmagan tabulyatsiya tartibiga ega bo‘ladi. Ikkita komponenta bir xil tabulyatsiya tartibiga ega bo‘lishi mumkin emas .

**TabStop** xossasi TabOrder bilan birgalikda ishlatilib, komponentaning fokus olishi yoki yo‘qligini aniqlaydi. Agar TabStop xossasining qiymati true bo‘lsa komponenta fokus olishi mumkin, aks holda yo‘q. Vizual kom-

ponentialarning tabulatsiya tartibini Edit buyruqlar menyusida joylashgan EditTabOrder (tabulyatsiya tartibini o'zgartirish) muloqot darchasi orqali amalga oshirish mumkin.

**ReadOnly** xossasi bool turida bo'lib, boshqaruv elementiga unda joylashgan matnni kiritish yoki tahrirlash bilan bog'liq amallarga ruxsat bor yoki yo'qligini aniqlaydi. Agar ReadOnly xossasi true qiymatga ega bo'lsa unda matnga faqat o'qish uchun murojaat qilish mumkin, agar ReadOnly xossasi false qiymati qabul qilsa, matnni tahrirlash mumkin bo'ladi. Masalan Edit satridan matnni faqat o'qish rejimida bermoqchi bo'lsak, quyidagi amallar ketma-ketligi bajarilishi kerak:

```
Edit1->Text="O'zgarimas matn";
```

```
Edit1->ReadOnly=true;
```

Shuni ta'kidlash kerakki, ReadOnly faqat ilova ishlagan paytdagina amal qiladi va ReadOnly xossasining qiymati true bo'lganda ham ilova ishlashida "*ichkaridan*" Text xossasining qiymatlarini o'zgartirishi mumkin.

Vizual komponentalar Color, Ctr3D, Font va Showhint xossalari uchun kelishuv bo'yicha qiymatni "*ota*" konteynerning (asosan forma) mos xossalarning qiymatini olishi mumkin.

Ko'rsatilgan xossalarning qiymat manbaini bool turidagi quyidagi xossalar aniqlab beradi:

ParentColor - fon rangi uchun;

ParentCtl3D - komponenta o'lcham ko'rinishi;

ParentFont - matn shrifti uchun;

ParentShowHint - yordamchi matnni ko'rsatish uchun.

Aksariyat hollarda kelishuv bo'yicha bu xossalar qiymatlari true bo'ladi. Agar ilova tuzuvchi komponentadagi mos xossalarni o'zgartirsa, unga mos keluvchi ParentXXX xossasining qiymati false bo'ladi.

**Parent** xossasi komponenta uchun "*ota*" boshqaruv komponentaga ko'rsatadi. "Ota" boshqaruv komponentasi sifatida konteyner keladi va o'z ichidagi vizual komponentalarni qanday tasvirlanishiga javob beradi. Formani loyihalashda va unga turli xil komponentalarni joylashtirishda komponentani qaysi konteynerga joylashishiga mos ravishda Parent xossasi avtomatik ravishda to'g'ri qiymat qabul qiladi. Komponentalarni dinamik ravishda yaratish vaqtida komponentaning Parent xossasiga "*qo'lda*" qiymat berishga to'g'ri keladi.

**Owner** xossasi mavjud bo'lib u komponentaning egasi bo'lgan elementga ko'rsatadi. Odatda komponentalar egasi ular joylashgan forma bo'ladi. Agar ega komponenta o'chirilsa, u egalik qiluvchi barcha komponentalar o'chib ketadi.



**Constraints** xossasi interfeys elementlari o'lchamlarini cheklash uchun kiritilgan. Bu turdagi xossalar ichida eng muhimlari MinHeight, MaxHeight, MinWidth va MaxWidth xossalari hisoblanadi va ular mos ravishda boshqaruv elementining balandligi va enining maksimal va minimal qiymatlarini beradi. O'z navbatida boshqaruv elementining bo'yi va eni piksel o'lchamida Height va Width xossalarining qiymatlari orqali aniqlanadi. Boshqaruv elementi o'lchamlariga cheklov qo'yish uni ichida joylashgan boshqa elementlarni ko'rinmay qolishligini oldini olishga qaratilgan.

### Vizual komponentalarning hodisalari

Vizual kompanintalar turli ko'rinishdagi hodisalarni yuzaga keltirish va qayta ishlash imkoniyatlariga ega. Eng umumiy hodisalar guruhlariga qo'yidagilarni kiritish mumkun:

- boshqaruv elementlarini tanlash;
- sichqoncha ko'rsatkichini harakatlantirish (ko'chirish);
- klaviatura tugmasini bosish;
- boshqaruv elementi tomonidan qiymat kiritish fokusini olish va yo'qotish;
- obyektlni "*drag and drop*" usulida ko'chirish.

**OnClick** hodisasi boshqaruv elementini tanlaganda ro'y beradi. Odatda bu hodisa sichqoncha tugmasi bilan komponentani bosganda ro'y beradi. Ilova yaratishda OnClick juda keng ishlatiladi. Misol uchun Label1 yozuvini tanlagandagi hodisani ishlash.

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    Label1->Caption=TimeToStr(Time());
}
```

Sichqoncha bilan Button1 tugmasiga bosilganda Label1 joylashgan joyda ayni paytdagi vaqt akslanadi.

OnClick hodisasi boshqa hollarda ham yuzaga kelishi mumkin. Masalan, Button1 komponentasida kiritish fokus turgan holda probel yoki "*Enter*" tugmasi bosilsa (CheckBox komponentasi uchun probel tugmasi bosiladi) ham OnClick hodisasi ro'y beradi.

**OnMouseDown** hodisasi sichqoncha tugmasini bosganda ro'y beradi.

**OnMouseUp** hodisasi sichqoncha tugmasi qo'yib yuborganda ro'y beradi.

Umuman olganda, sichqoncha tugmasini bosib, qo'yib yuborilganda qo'yidagi hodisalar ketma - ketligi ro'y beradi :

OnMouseDown  $\Rightarrow$  OnClick (chap tugma uchun)  $\Rightarrow$  OnMouseUp.

Agar komponenta sohasiga sichqoncha tugmasini ikkilangan bosishi amalga oshirilsa, OnDblClick hodisasi ro‘y beradi. Bu holdagi hodisalar ketma - ketligi qo‘yidagicha bo‘ladi:

MouseDown ⇒ OnClick ⇒ OnMouseUp ⇒ OnDblClick ⇒ OnMouseDown ⇒ OnMouseUp.

C++ Builder yuqorida qayd qilingan holatlarni “qo‘lda” yuzaga keltirishga imkon beradi. Masalan, Button2 -> Click() amali Button2 tugmasi bosilishini immitatsiya qiladi.

**OnMouseMove** hodisasi vizual komponenta ustida sichqoncha ko‘rsatgichini harakat qilishi davomida uzluksiz ravishda yuzaga kelib turadi. Bu hodisa funksiyasining ko‘rinish quyidagicha:

```
void __fastcall TForm1::FormMouseMove(TObject *Sender,  
                                     TShiftState Shift, int X, int Y) { }
```

Bu yerda Sender - sichqoncha ko‘rsatgichi qaysi boshqaruv elementi ustida ekanligini, X va Y sichqoncha ko‘rsatgichining Sender boshqaruv elementi koordinata tizimidagi koordinatalarini ko‘rsatadi. Shift parametri <Alt>, <Ctrl> va <Shift> tugmalarining holatini bildiradi. Bu parametr quyidagi qiymatlarning kombinatsiyasini olishi mumkin:

- SsShift - <Shift> tugmasi bosilgan;
- SsAlt - < Alt > tugmasi bosilgan;
- SsCtrl - < Ctrl > tugmasi bosilgan;
- SsLeft - sichqonchanning chap tugmasi bosilgan;
- SsMiddle - sichqonchanning o‘rta tugmasi bosilgan;
- SsDouble - ikkilangan bosish amalga oshirilgan.

Masalan, sichqoncha ko‘rsatkichining koordinatalarini chop qilish:

```
void __fastcall TForm1::FormMouseMove(TObject *Sender,  
                                     TShiftState Shift, int X, int Y)  
{  
  Form1->Caption="Sichqoncha ko‘rsatgichi koordinatalari: ("  
    + IntToStr(x) + ", " + IntToStr(y)+")" );  
}
```

Sichqoncha ko‘rsatgichini forma ustida harakatlantirganda uning koordinatalari forma sarlavhasida ko‘rsatiladi.

**OnKeyPress** va **OnKeyDown** hodisalari klaviatura tugmasini bosganda yuzaga keladi.

**OnKeyUp** hodisasi klaviatura tugmasini qo‘yib yuborganda yuzaga keladi. Klaviatura tugmasini bosganda hodisalar quyidagi ketma - ketlikda ro‘y beradi:

OnKeyDown ⇒ OnKeyPress ⇒ OnKeyUp.

Klaviaturani bosib turganda uzluksiz ravishda **OnKeyDown** hodisasi, tugma qo‘yib yuborilgandan keyin **OnKeyUp** hodisasi ro‘y beradi.

**OnKeyPress** hodisasi klaviatura tugmasi bosilganda ro‘y beradi va bosilgan tugmaga mos keluvchi belgining ASCII kodini qabul qiladi.

**OnKeyPress** hodisasini qayta ishlashga misol:

```
void __fastcall TForm1::Edit1KeyPress(TObject *Sender, char &Key)
{
    if(Key=='!') Key=0;
}
```

Bu yerda Edit1 tahriridagi matnni terishda foydalanuvchiga ‘!’ belgisini ishlatish man qilinadi.

**OnEnter** hodisasi boshqaruv elementlari fokus olganida yuz beradi (sichqoncha yoki <**Tab**> tugmasi yordamida).

**OnExit** hodisasi darcha boshqaruv elementi fokus yo‘qotganda yuz beradi.

## 2-ilova

### Matematik funksiyalar kutubxonasi (math.h)

Funksiya prototipi	Bajaradigan amali
int abs(int i)	i sonining absolyut qiymatini qaytaradi
double acos(double x)	radianda berilgan x argumentning arkko sinus qiymatini qaytaradi
double asin(double x)	radianda berilgan x argumentning arksinus qiymatini qaytaradi
double atan(double x)	radianda berilgan x argumentning arktangens qiymatini qaytaradi
double atan2(double x, double y)	radianda berilgan x/y argumentning arktangens qiymatini qaytaradi
double ceil(double x)	haqiqiy x qiymatini unga eng yaqin katta butun haqiqiy songacha yaxlitlaydi
double cos(double x)	x radianga teng bo'lgan burchakning kosinusini qaytaradi
double cosh(double x)	x radianga teng bo'lgan burchakning gipebolik cosinusini qaytaradi
double exp(double x)	$e^x$ qiymatini qaytaradi
double fabs(double x)	haqiqiy sonni absolyut qiymatini qaytardi
double floor(double x)	haqiqiy x qiymatini unga eng yaqin kichik songa aylantiradi va uni haqiqiy son ko'rinishida qaytaradi
double fmod(double x, double y)	x sonini y soniga bo'lish natijasidagi qoldiqni qaytaradi. Amal % amaliga o'xshash, faqat haqiqiy son qaytariladi
double frexpr(double x, int *expPtr)	x sonining mantissasini va darajasini ajratib, mantissa qiymatini qaytaradi va darajasini ko'rsatilgan expPtr adresiga joylashtiradi
double hypot(double x, double y)	to'g'ri burchakli uchburchakning katetlari bo'yicha gipotenuzani hisoblaydi
long int labs(long int num)	num uzun butun sonining absolyut qiymatini qaytaradi
double ldexp(double x, int exp)	$x \cdot 2^{\text{exp}}$ qiymatini qaytaradi
double log(double x)	x sonining natural logarifmini qaytaradi
double log10(double x)	x soning 10 asosli logarifmini qaytaradi

double modf (double x, double *intptr)	x sonining kasr qismini qaytaradi va butun qismini intptr adresga joylaydi
double poly (double x, int n, double c[])	$c[n]x^n+c[n-1]x^{n-1}+ \dots+c[1]x+c[0]$ polinom qiymatini hisoblaydi
double pow(double x, double y)	$x^y$ qiymatini hisoblaydi
double pow10(int p)	$10^p$ qiymatini hisoblaydi
double sin(double x)	x radianga teng bo'lgan burchakning sinusini qaytaradi
double sinh(double x)	x radianga teng bo'lgan burchakning giperbolik sinusini qaytaradi
double sqrt(double x)	x sonining kvadrat ildizini qaytaradi
double tan(double x)	x radianga teng bo'lgan burchakning tangensini qaytaradi
double tanh(double x)	x radianga teng bo'lgan burchakning giperbolik tangensini qaytaradi

## 3-ilova

### C++ Builder muhitida grafik shakllarni chizish

C++ Builder muhitida chizish sirti - TCanvas sinfi ilova ishlash paytida rasm chizish imkonini beradi. Bu sinf obekti sirt bo'yicha ko'chish, grafik primitivlar chizish, rasmlarni va sirtning biror qismini nusxalash, hamda matnni chop qilish imkonini beruvchi xossa va usullarni o'z ichiga oladi.

Har bir Canvas xossasiga ega komponenta o'z navbatida qalam, mo'yqalam va shrift obektlarini tarkibiga oladi va mos ravishda Pen, Brush va Font xossalariga ega bo'ladi.

**Pen** xossasi rangga (Canvas->Pen->Color), chizishning piksellardagi qalinlikka (Canvas->Pen->Width), chizilayotgan chiziq toifasiga (Canvas->Pen->Style) ega. Chiziq toifasi quyidagi qiymatlarni qabul qilishi mumkin:

psSolid – uzluksiz chiziq (kelishuv bo'yicha);

psDash – tire belgilaridan hosil bo'lgan chiziq;

psDot – nuqtalardan tashkil topgan chiziq;

psDashDot – nuqta va tirelardan iborat chiziq;

psDashDotDot – tire va nuqtalardan iborat chiziq;

psClear – ko'rinmas chiziq;

psInsideFrame – chizish sirtini chegaralovchi to'g'ri to'rtburchak ichidagi chiziq.

**Brush** xossasi geometrik shakllar, masalan, to'g'ri to'rtburchak va ellips ichini to'ldirish naqshini aniqlaydi. U quyidagi xossalarga ega:

Canvas->Brush->Color – mo'yqalam ranggi;

Canvas->Brush->Style – mo'yqalam toifasini aniqlaydi va u quyidagi qiymatlarni qabul qilishi mumkin:

bsSolid – berilgan rang bilan shakl yuzasi to'liq bo'yaladi;

bsClear -shakl yuzasi bo'yalmaydi;

bsHorizontal - shakl yuzasi parallel chiziqlar bilan to'ldiriladi;

bsVertical - shakl yuzasi vertikal chiziqlar bilan to'ldiriladi;

bsFDiagonal - shakl yuzasi yuqoriga qaragan chiziqlar bilan to'ldiriladi;

bsFDiagonal - shakl yuzasi diagonal chiziqlar bilan to'ldiriladi;

bsCross - shakl yuzasi to'r bilan to'ldiriladi;

bsDiagCross - shakl yuzasi egri chiziqlardan hosil bo'lgan to'r bilan to'ldiriladi.

Canvas obyektining muhim xossalaridan biri `Pixels[x][y]` xossasi bo'lib, u ko'rsatilgan koordinatadagi piksel rangini aniqlaydi. Bu xossa qiymatini o'qish va unga qiymat yozish mumkin.

Geometrik shakllar chizish uchun quyida keltirilgan funksiyalardan foydalanish mumkin:

`Arc(int x1,int y1,int x2,int y2,int x3,int y3,int x4,int x4)` – yoy chizish. Bu yerda  $(x_1,y_1)$  va  $(x_2,y_2)$  – mos ravishda yoy chiziladigan to'rtburchak sohaning chap yuqori va o'ng past uchlari koordinatasi.  $(x_3,y_3)$  va  $(x_4,y_4)$  nuqtalar mos holda yoy boshlanishi va oxiri koordinatasi.

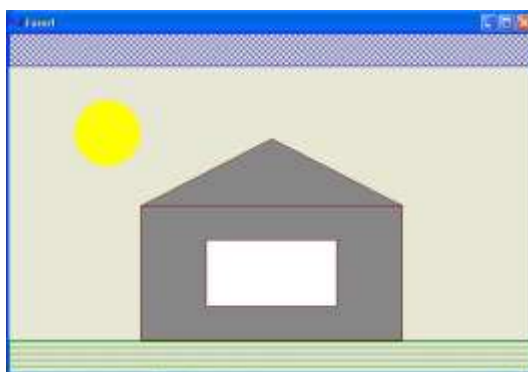
`Chord(int x1,int y1,int x2,int y2,int x3,int y3,int x4,int x4)` – ellips vatarini chizish. Bu yerda  $(x_1,y_1)$  va  $(x_2,y_2)$  – mos ravishda yoy chiziladigan to'rtburchak sohaning chap yuqori va o'ng past uchlari koordinatasi.  $(x_3,y_3)$  va  $(x_4,y_4)$  nuqtalar mos holda vatar boshlanishi va oxirining koordinatalari.

`Ellipse(int x1,int y1,int x2,int y2)` – rang bilan to'ldirilgan ellipsni chizish. Bu yerda  $(x_1,y_1)$  va  $(x_2,y_2)$  – mos ravishda yoy chiziladigan to'rtburchak sohaning chap yuqori va o'ng past uchlari koordinatasi.

`Rectangle(int x1,int y1,int x2,int y2)` – rang bilan to'ldirilgan to'g'ri to'rtburchakni chizish. Bu yerda  $(x_1,y_1)$  va  $(x_2,y_2)$  – mos ravishda to'rtburchak sohaning chap yuqori va o'ng past uchlari koordinatasi.

**Masala.** Forma sirtida Canvas xossalaridan foydalangan holda uy rasmini chizish.

Uy rasmini chizishda ellips, to'g'ri to'rtburchak, ko'pburchak shakllari ishlatilgan. Rangni boshqarish, shakl yusasini to'ldirish uchun qalam va mo'yqalam xossalari ishlatilgan. Rasm o'lchamlari forma o'lchamiga mos ravishda o'zgaradi.



Quyida `Unit1.cpp` fayli matni keltirilgan.

```
#include <vcl.h>
#pragma hdrstop
#include "Unit1.h"
#pragma package(smart_init)
```

```

#pragma resource "*.dfm"
TForm1 *Form1;
__fastcall TForm1::TForm1(TComponent* Owner) : TForm(Owner) { }
void __fastcall TForm1::FormResize(TObject *Sender)
{
int w, h, wm, hm;
Form1->Refresh();
wm=Form1->ClientWidth; w=wm/8;
hm=Form1->ClientHeight; h=hm/10;
// Osmon
Form1->Canvas->Brush->Color=clBlue;
Form1->Canvas->Brush->Style=bsDiagCross;
Form1->Canvas->Pen->Color=clBlue;
Form1->Canvas->Rectangle(0,0,wm,h);
// Maysalar
Form1->Canvas->Brush->Color=clGreen;
Form1->Canvas->Brush->Style=bsHorizontal;
Form1->Canvas->Pen->Color=clGreen;
Form1->Canvas->Rectangle(0,hm-h,wm,hm);
// Quyosh
Form1->Canvas->Brush->Color=clYellow;
Form1->Canvas->Brush->Style=bsSolid;
Form1->Canvas->Pen->Color=clYellow;
Form1->Canvas->Ellipse(w,2*h,2*w,2*h+w);
// Uycha
Form1->Canvas->Brush->Color=clGray;
Form1->Canvas->Brush->Style=bsSolid;
Form1->Canvas->Pen->Color=clMaroon;
Form1->Canvas->Rectangle(2*w, hm-5*h, 6*w, hm-h);
TPoint point[4];
point[0].x=2*w; point[0].y=hm-5*h;
point[1].x=4*w; point[1].y=hm-7*h;
point[2].x=6*w; point[2].y=hm-5*h;
point[3].x=2*w; point[3].y=hm-5*h;
Form1->Canvas->Polygon(point,3);
Form1->Canvas->Brush->Color=clWhite;
Form1->Canvas->Brush->Style=bsSolid;
Form1->Canvas->Pen->Color=clMaroon;
Form1->Canvas->Rectangle(3*w, hm-4*h, 5*w, hm-2*h);
}

```



Ilova ishga tushishi bilan Formaning OnResize hodisasi ro‘y beradi va ekranda uycha rasmi paydo bo‘ladi.

### Matematik funksiyalar grafigini chizish

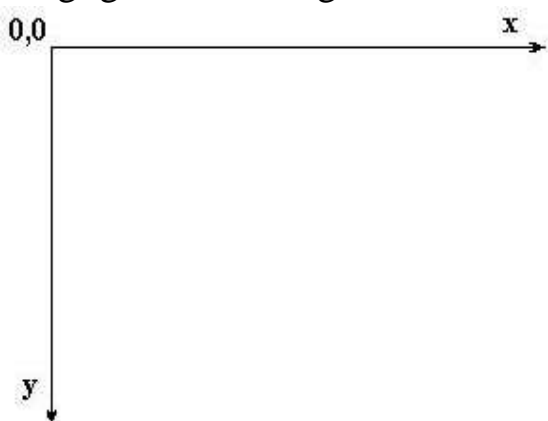
C++ Builder muhitida grafik shakllarni chizish Canvas komponentasi vositsida amalga oshiriladi. Ayrim vizual komponentalar bu komponentaga ega. Masalan, TForm, TImage, TPaintBox va boshqalar.

Formaning (Form1) grafik shakllar chizish sohasi bu mijoz sohasi hisoblanadi va uning o‘lchami ClientWidth (gorizontaliga) va ClientHeight (vertikaliga) bilan aniqlanadi.

Grafik chizish sohasi adreslanuvchi nuqtalarning to‘g‘ri burchakli massiv ko‘rinishida bo‘ladi va ixtiyoriy tasvir yonib yoki o‘chib turgan piksellar (tasvirning minimal elementi) kompozitsiyasidan hosil bo‘ladi. Bu nuqtalar ikkita butun son:  $n_x$  - nuqtaning gorizontal nomeri va  $n_y$  nuqtaning vertikal nomeri bilan adreslanadi:

$$0 \leq n_x \leq n_{x_{\max}}; 0 \leq n_y \leq n_{y_{\max}},$$

bu yerda  $n_{x_{\max}}$  va  $n_{y_{\max}}$  - mos ravishda chizish sohasining piksillardagi kengligi va balandligi.



Chizish sohasining chap yuqori burchagi (0,0) koordinataga ega bo‘ladi.  $(n_x, n_y)$  qurilma koordinatalari ham deyiladi va ular faqat butun qiymatlarni qabul qiladi.

Kompyuter grafikasida yana ikkita koordinata tizimi qabul qilingan. Birinchisi  $(p_x, p_y)$ -ekran koordinata tizimi bo‘lib, unda  $p_x$  - gorizontal bo‘yicha ekrandagi masofa,  $p_y$ -gorizontal

bo‘yicha. Bu yerda koordinata o‘qlari millimetr va duymlarda o‘lchanadi. Ikkinchi koordinata tizimi - dunyoviy (olam) koodinata tizimidir. U  $(x, y)$  dekart tizimi bo‘lib, ilova tuzuvchisi tomonidan aniqlanadi va tasvirlash qurilmasiga bog‘liq bo‘lmaydi:

$$X_{\min} < x < X_{\max}; Y_{\min} < y < Y_{\max}.$$

Dekart koordinatalar sistemasida X va Y o‘zgarish diapazonlari  $(X_{\min}, X_{\max}, Y_{\min}, Y_{\max})$  mavhum matematik ikki o‘lchamli fazoning to‘g‘ri burchakli sohasini aniqlaydi. Bu sohani qurilma koordinatasiga akslantirish quyidagicha amalga oshiriladi:

$$n_x = (x - X_{\min}) / (X_{\max} - X_{\min}) * n_{x_{\max}};$$

$$n_y = (y - Y_{\min}) / (Y_{\max} - Y_{\min}) * n_{y_{\max}},$$

bu yerda  $(x,y)$ - dekart koordinatasidagi nuqta va uning ekrandagi koordinatasi  $(n_x,n_y)$  bo'ladi.

**Grafik kursor.** Grafik kursor matn kursori bajaradigan ishni bajaradi, lekin u ekranda ko'rinmaydi. Ma'lumki matn kursori ekrandagi belgi o'rnini (80\*25 bo'lganida) ko'rsatadi va bu o'rinda belgi chop qilinganda avtomatik ravishda bir o'rin o'ngga suriladi. Grafik kursor esa chiqariluvchi grafik shaklning boshlang'ich koordinatasini ko'rsatadi va uni keyingi joyga  $(n_x,n_y)$  nuqtaga ko'chirish uchun maxsus funksiya ishlatiladi: `MoveTo(nx,ny);`

**Chiziqnlarni chizish.** Sohada chiziqni (kesmani) chizish uchun `LineTo()` funksiyasidan foydalaniladi. Masalan,  $(x_1,y_1)$  va  $(x_2,y_2)$  nuqtalarni tutashtiruvchi kesma chizish uchun quyidagi amallar bajarilishi kerak:

```
Form1->Canvas->MoveTo(x1,y1);
```

```
Form1->Canvas->LineTo(x2,y2);
```

Ekranda ko'p miqdordagi siniq chiziqlardan tashkil topgan shaklni chizish uchun `Canvas->Polyline(Jadval,n)` funksiyasidan foydalaniladi. U berilgan sondagi sonlar juftligi majmuasi bilan aniqlangan siniq chiziqni chizadi.  $n$  parametri siniq chiziq tugun nuqtalari soni. Jadval parametri `TPoint` turida bo'lib, grafik soha nuqta koordinatasini aniqlovchi strukturalar massivdir. Siniq chiziq tugun nuqtalari `Jadval` massivi sifatida beriladi. Quyida `PaintBox1` komponentasi sohasida  $\sin(x)$  funksiya grafigini chizish funksiyasi keltirilgan.

```
void Sin_Grafigini_Chizish()
{
    const double Pi=3.14151828;
    double Qadam =0.1;
    double Burchak_Radian=0;
    const int Nuqtalar_Soni=100;
    int Mashtab=50;
    TPoint Sin_func[Nuqtalar_Soni];
    int Absissa = PaintBox1->Height/2;
    for (int i=0; i<Nuqtalar_Soni; i++)
    {
        Sin_func[i].x =(int)(Mashtab * Burchak_Radian)+10;
        Sin_func[i].y =Absissa-(int)(Mashtab*sin(Burchak_Radian));
        Burchak_Radian+=Qadam;
    }
    PaintBox1->Canvas->Pen->Color=clBlack;
    PaintBox1->Canvas->Polyline(Sin_func,Nuqtalar_Soni-1);
}
```

Quyidagi ilovada  $\sin()$  funksiya grafigini chizishning boshqa varianti qaralgan.

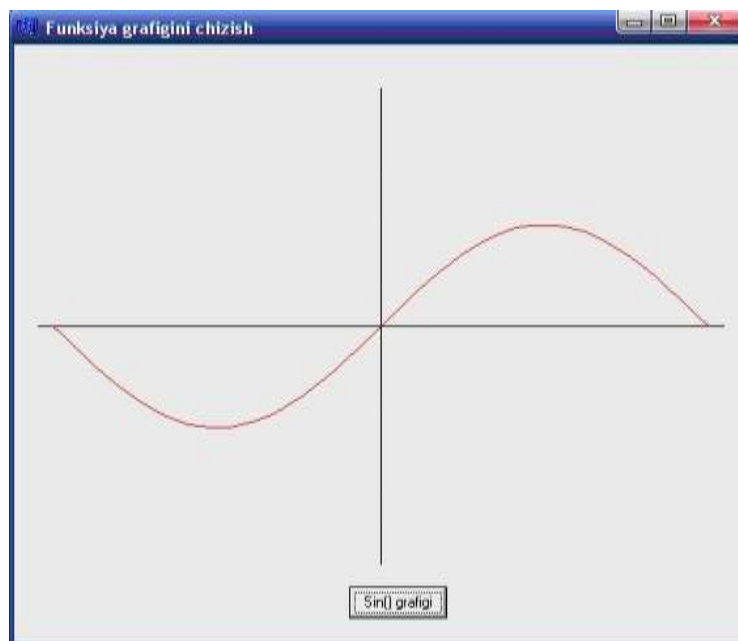
```
...
class FGrafika
{
private:
    TPaintBox * Soha;
    double(*f)(double);
public:
    FGrafika(TPaintBox * _Soha, double(*_f)(double)) {Soha=_Soha; f=_f; }
    void Fun_Grafika();
};
void FGrafika::Fun_Grafika()
{
    double Pi=3.141592654;
    int Xe0,Ye0,Xe,Ye, // Ekran koordinatalari
        Rect_X,Rect_Y, // Koordinata chegaralarining soha chegarasidan farqi
        Mashtab; // Soha koordinatasining Haqiqiysiga nisbati
    double h,X,Y; // Y=f(X) funksiyasi va h qadam
    Mashtab=80; // Mashtabni tanlash
    Rect_X=Rect_Y=10; // Chegaralar
    h=0.1; // chizish qadami
    Xe0=Soha->Width/2; // Koordinata markazini tanlash
    Ye0=Soha->Height/2;
    Soha->Canvas->MoveTo(Rect_X,Ye0);
    Soha->Canvas->LineTo(Soha->Width-Rect_X,Ye0); //OX-o'qi
    Soha->Canvas->MoveTo(Xe0,Rect_Y);
    Soha->Canvas->LineTo(Xe0,Soha->Height-Rect_Y); //OY-o'qi
    X=-Pi; // - Pi va Pi oralig'i uchun fuksiya grafigi chiziladi
    Y=f(X);
    Xe=Xe0+(int)(Mashtab*X);
    Ye=Ye0-(int)(Mashtab*Y);
    Soha->Canvas->MoveTo(Xe,Ye); //Grafik kursorni o'rnatish
    Soha->Canvas->Pen->Color=clRed;
    do
    {
        X=X+h;
        Y=f(X);
        Xe=Xe0+(int)(Mashtab*X);
        Ye=Ye0-(int)(Mashtab*Y);
```

```

// Koordinata chegarasida chizish
if(Xe > Rect_X && Xe < Soha->Width-Rect_X
  && Ye > Rect_Y && Ye < Soha->Height-Rect_Y) Soha->Canvas-
>LineTo(Xe,Ye);
}
while(X<=Pi);
}
 TForm1 *Form1;
__fastcall TForm1::TForm1(TComponent* Owner)
: TForm(Owner) {}
void __fastcall TForm1::Button1Click(TObject *Sender)
{
  FGrafika Sinus(PaintBox1,sin);
  Sinus.Fun_Grafika();
}

```

Ilova ishlashi natijasida Formadagi PaintBox1 komponenta sohasida quyidagi chizma paydo bo‘ladi.



## Mundarija

Kirish .....	2
1. Berilganlar turlari.....	4
2. Razryadlar ustida mantiqiy amallar.....	6
3. Matematik ifodalarni hisoblash .....	7
4. Hisoblashga doir sodda masalalar .....	10
5. Mantiqiy ifodalar .....	13
6. '?' shart amali .....	16
7. if-else, goto operatorlari .....	18
8. switch tarmoqlanish operatori .....	22
9. for takrorlash operatori .....	25
10. while, do-while takrorlash operatorlari .....	28
11. Funksiyalar .....	32
12. Vektorlar .....	35
13. Matritsalar.....	39
14. Statik massivlar funksiya argumenti sifatida.....	43
15. Dinamik massivlar funksiya argumenti sifatida.....	46
16. Satrlar.....	51
17. Rekursiv funksialar .....	56
18. Strukturalar .....	61
19. Matn fayllari .....	67
20. Binar fayllar .....	73
21. Ko'rsatkich turi.....	80
22. Sinflar. Inkapsulyatsiya .....	86
23. Vorislik .....	95
24. Operatorlarni qayta yuklash.....	102
25. Polimorfizm .....	109
26. Qoliplar. STL kutubxonasi .....	121
27. Grafika .....	127
Foydalanilgan adabiyotlar .....	132
1-ilova.....	133
2-ilova.....	148
3-ilova.....	150









4, 1, 8, 5, 12, 9, 16, 13, 20, 17, 24, 21, 28, 25, 32, 29, 36, 33, 40, 3  
7, 44, 41, 48, 45, 52, 49, 56, 53, 60, 57, 64, 61, 68, 65, 72, 69, 76,  
73, 80, 77, 84, 81, 88, 85, 92, 89, 96, 93, 100, 97, 104, 101, 108, 1  
05, 112, 109, 116, 113, 120, 117, 124, 121, 128, 125, 132, 129, 13  
6, 133, 140, 137, 144, 141, 148, 145, 152, 149, 156, 153, 160, 157

158, 159, 154, 155, 150, 151, 146, 147, 142, 143, 138, 139, 134, 1  
35, 130, 131, 126, 127, 122, 123, 118, 119, 114, 115, 110, 111, 10  
6, 107, 102, 103, 98, 99, 94, 95, 90, 91, 86, 87, 82, 83, 78, 79, 74,  
75, 70, 71, 66, 67, 62, 63, 58, 59, 54, 55, 50, 51, 46, 47, 42, 43, 38  
, 39, 34, 35, 30, 31, 26, 27, 22, 23, 18, 19, 14, 15, 10, 11, 6, 7, 2, 3