10-Ma’ruza. Umumiy takrorlanish algoritmlari va ichma-ich takrorlanishlar

Ma’ruza rejasi:

10.1 C++tilida takrorlanuvchi jarayonlarni dasturlash

10.2 Continue operatori

Kalit so’zlar: *delete, masofa keltirish, delete[], new, indeks, this, indeksirlash, [] bo’sh xotira, void*, konteyner, ro’yxat, manzil, nolinchi ko’rchsatkich, tugun, adres olish &, bo’shatish, ko’rsatkich, virtual destruktor, xotira, xotira chiqishi, destruktor, toifani o’zlashtirish, resurslar chiqishi, a’zo destruktori.*

8.1 C++tilida takrorlanuvchi jarayonlarni dasturlash

Agar dastur bajarilish jarayonida operator yoki operatorlar guruhi bir necha marta qayta-qayta bajarilsa, bunday jarayonlarni takrorlanuvchi (siklik) jarayon deyiladi. C++ tilida siklni 3 xil ko’rinishda tashkil qilish mumkin.

1. Sharti avval tekshiriladigan takrorlanish (oldshartli sikl):

while (shart) operator (lar);

Bu yerda operatorlar while da ko’rsatilgan shart yolg’on bo’lgunicha takrorlanadi. Takrorlanish tanasi murakkab bo’lsa, ya’ni 1 tadan ortiq operatorlar qatnashsa, ularni alohida {} ichiga olish kerak bo’ladi.

Masalan: b = 2*(a+5); a € [1, 10]; h=1;

```
# include <iostream.h>
# include <math.h>
void main ( )
{  int a=1, b;
 while (a<=10)
 {  b = 2*(a+5); cout << “b=” <<b;
 cout << “a=” <<a << endl;
 a++ ; }
 }
```

Ekranda 10 ta a va b larning qiymatlari paydo bo’ladi.

2- misol.

```
# include <iostream.h>
 void main ( )
 {  int i = 10;
   while ( i++ < =15)
   cout << “Salom!!!”<< endl;  }
```

Ekranda 5 marta “Salom!!!” yozuvi paydo bo’ladi.

2. Sharti keyin tekshiriladigan takrorlanish (so’ngshartli sikl):

```
do
operator (lar)
```

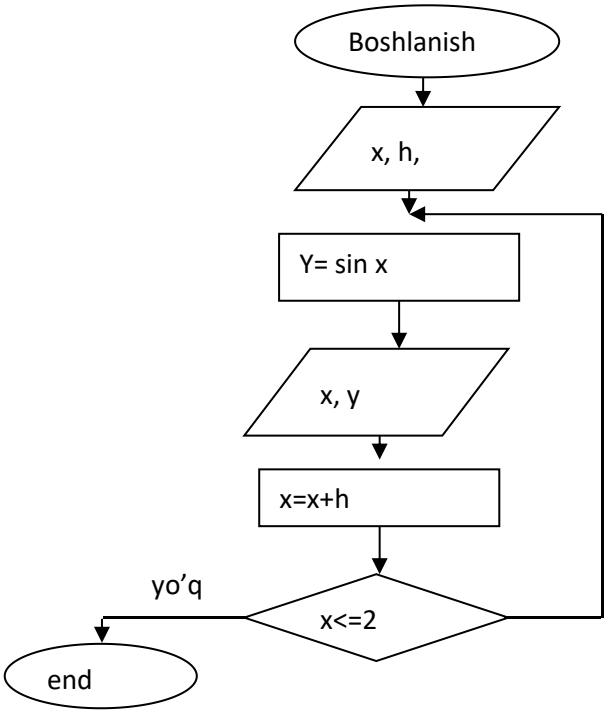
while (shart);

Takrorlanish while da ko’rsatilgan shart yolg’on bo’lgunicha davom etadi.

Masalan: y=sinx; x € [1,2] ; h=0.1

```
# include <iostream.h>                    do
# include <math.h>                        { y=sin(x); cout << “x=”<<x<<”
 void main ( )                          y=”<<y<<endl;    x+ = 0.1;  }
{ float x=1, y;                          while (x<=2); getch(); }
```

Masalaning algoritmi quyidagi ko’rinishga ega bo’ladi:



2- misol. Dastur klaviaturadan 20 soni kiritilishini kutadi.

```
# include <iostream>                     cout << “Sonni qayta
void main ( )                          kiriting!=”<<n; }
{ int n;                                 while (n!=20);
  do                                     }
  { cin >> n;
```

3. Parametrli takrorlanish (sikl):

Umumiy ko’rinishi

```
for (bosh qiymat; shart; o’zgarish qadami)
operator (lar);
```

Operatorlar 1 tadan ortiq bo’lsa ularni alohida qavslar -{} ichiga olinadi.

1-misol. Y = cos x; x € [2,3] ; h=0,2;

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

```
void main ( )
{ float x, y;
 for (x=2; x<=3; x+ = 0.2)
 { y=cos(x);  cout << "x="<< x <<
"  y="<< y << endl; }
}
```

2-misol. 100 gacha bo'lgan juft sonlarni ekranga chiqarish dasturi.

```
# include <iostream.h>
 void main ( )
{   int i = 2;
 while (i<=100)
 { cout << "i="<< i;  i += 2; }  }
```

….
for (int i=2; i<=100; i +=2)
cout << "i="<< i;
	do
	cout << "i="<< i; i += 2;
	while (i<=100);

3-misol: 1 dan 100 gacha bo'lgan 3 raqami bilan tugaydigan sonlarni ekranga chiqarish dasturini tuzing (2 xil usulda).

……
int i=3;
while (i <=100)
{ cout << "i="<< i;
i + = 10; }
	…..
	for (i = 3;i <= 100; i + = 10)
	cout << "i="<< i;

4-misol. Qadimiy masala. Bir odam 100 so'm bilan bozorga bordi. Bozorda 1 ta sigir 10 so'm, 1 ta qo'y 3 so'm, 1 ta echki 0.5 so'm va xarid qilingan qoramollarning umumiy soni 100 ta bo'lsa, nechta sigir, qo'y va echki sotib olindi?

Sigirlar soni: x, qo'ylar soni y, echkilar soni z deb olinsa,

```
 # include <iostream.h>
int main ( )
{
int x, y, z, s;
for (x=1; x<=100; x++)
for (y=1; y<=100; y++)
if (19*x + 5*y = = 100)
{ z = 100 – x – y;
 cout << "x="<<x;
 cout << "y="<<y;
 cout << "z="<<z; }
 return 0;
 }
```

```
 # include <iostream.h>
 int main ( )
 {
 int x, y, z, s;
 for (x=1; x<=100; x++)
 for (y=1; y<=100; y++)
 for (z=1; z<=100; z++)
 if (x + y + z = = 100)
 {  cout << "x="<<x;
  cout << "y="<<y;
  cout << "z="<<z; }
  return 0;
  }
```

10.2 Continue operatori

Bu operator yordamida sikl parametrining biror qiymatida hisoblashni to'xtatib, keyingi qiymatida hisoblashni davom ettirish mumkin. Masalan: y = 2x funksiyasini x € [1,18] oraliqda h=1 qiymat bilan hisoblash kerak, lekin x=6 yoki x=13 qiymatlarida hisoblashni bajarmaslik kerak.

```
# include <iostream.h>                    { if (( x = = 6) || (x = = 13)) continue;
void main ( )                             y = 2*x; cout << "x="<< x <<
{ int x, y;                             "  y=" << y << endl;
for (x=1; x<=18; x++)                     } }
```

2 - misol. 10 ta ketma-ket kiritiladigan butun musbat sonlar yig'indisini hisoblash dasturini tuzing. Agar son manfiy bo'lsa, yig'indiga qo'shmaslik kerak.

```
# include <iostream.h>                    if ( x < 0)  continue;
void main ( )                             s = s + x; }        // s +=x deb yozsa
{   int x, i, s=0;                      ham bo'ladi.
  for ( i =1; i <=10; i ++)               cout << "s="<< s << endl;    }
  { cin >> x;
```

3-misol. $Y = x^n$ funksiyasini rekurrent formula orqali hisoblash dasturini tuzing. Bu yerda n - butun son, x – ixtiyoriy haqiqiy son.

```
# include <iostream.h>                    for ( int n=1; n<=10; n++)
# include <conio.h>                       y = y * x;          // y * = x;
void main ( )                             cout <<"y="<<y<<endl;
{ float x=2.56, y=1;                      getch ( );  }
```

4-misol. $Y= \sum_{n=1}^{8} \frac{x^2}{n!} = x^2 + \frac{x^2}{1*2} + \frac{x^2}{1*2*3} + ... + \frac{x^2}{1*2*3*4*5*6*7*8}$

```
x - ixtiyoriy haqiqiy son.                for ( int n=1; n<=8; n++)
# include <iostream.h>                    { p = p * n; y = y + x*x / p; }
# include <conio.h>                       cout << "y="<<y<<endl;
void main ( )                             getch ( );
{ float x=3.75, y=0; long p=1;            }
```

5-misol. $S=\cos x + \frac{\cos 2x}{2} + \frac{\cos 3x}{3} + ... + \frac{\cos nx}{n}$;

bu yerda $\frac{\pi}{5} \le x \le \frac{9\pi}{5}$ n=10

```
# include <iostream.h>                    { float a, b, h, x, s, pi=3.14;
# include <conio.h>                       a = pi / 5;  b=9 * pi / 5; h=(b-a) / 10;
# include <math.h>                        x = a; cout. precision (3);
   void main ( )                          while ( x<=b )
```

```
{ s = 0;                                    x = x + h;
for ( int n=1; n<=10; n++)                  }
s = s + cos(n*x) / n;                       getch ( ); }
cout <<"s="<<s<<endl;
```

6-misol. Boy bilan olim bahslashibdilar. Olim boyga har kuni (30 kun) 100000 so'm beradigan bo'libdi. Boy esa olimga 1-kun 1 tiyin, 2-kun 2 tiyin, 3-kun 4 tiyin, 4-kun 8 tiyin va h.k. pul beradigan bo'libdi. Bahsda kim yutadi? Dasturini tuzing.

Olim → 30*100000 = 3000000 so'm

Boy → $\sum_{i=0}^{30} 2^i$ sum → 10737418 so'm

```
# include <iostream.h>                      k = k / 100;
# include <conio.h>                         s = 30*100000;
# include <math.h>                          cout <<"boy olimga
void main ( )                       beradi:"<<k<<endl;
{ int s, s1=1, k = 0;                       cout <<"olim boyga beradi:" << s <<
for ( int i=1; i<=30; i++)          endl;
{ k = k + s1;                               getch ( );
s1 = s1*2; }                                }
```

Nazorat savollari:

1. Sharti avval tekshiriladigan takrorlanish
2. Sharti keyin tekshiriladigan takrorlanish
3. Parametrli takrorlanish
4. Dasturda takrorlanishlarni tashkil etish.
5. Takrorlanuvchi dastur nima?
6. Murakkab takrorlanishlar
7. **continue** operatori
8. **return** operatori.