

Gjuhë programuese C++

MSc. Vehbi NEZIRI



Java 8

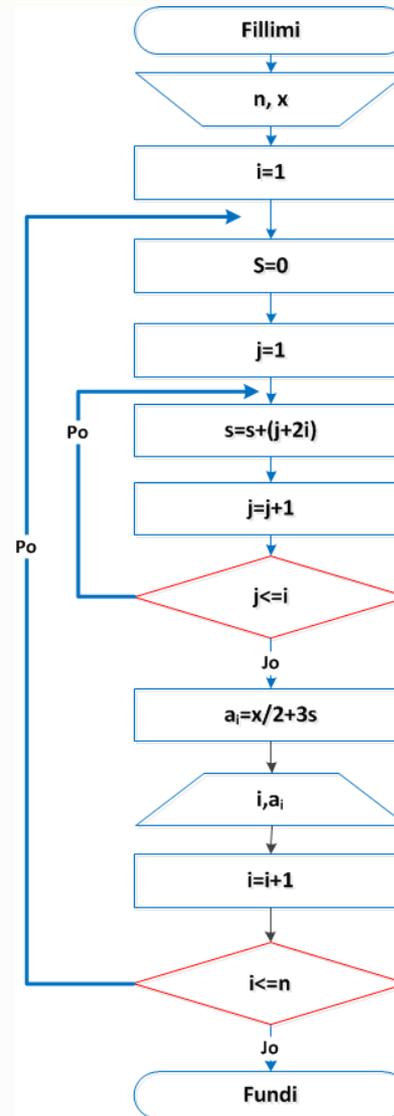
- Vepime me vektorë
 - Formimi i vektorit me vlera të caktuara
 - Anëtari maksimal dhe minimal
- Sortimi i anëtarëve
 - Nga A-Z
 - Nga Z-A
- Vepime me matrica
 - Mbi diagonale,
 - Në diagonale,
 - Nën diagonale



Shembull 8.1

- Të vizatohet bllok-diagrami dhe të formohet vektori $A(n)$ duke llogaritur anëtarët a_i përmes shprehjes së mëposhtme nëse dihet vlera x dhe n .

$$a_i = \frac{x}{2} + 3 \sum_{j=0}^i (j + 2i)$$



```
#include <iostream>
using namespace std;
```

```
int main()
{
    int const n=5,
    x=1;
    int i,j;
    double s,A[n];
    for (i=0;i<n;i++)
    {
        s=0;
        for (j=1;j<=i;j++)
            s=s+(j+2*i);
        A[i]=x/2.+3*s;
        cout << "A["
            << i
            << "]="
            << A[i]
            << "\n";
    }
    system ("pause");
    return 0;
}
```



Shembull 8.2

- Të formohet vektori $B(n)$ nga vlerat e anëtarëve të vektorit $A(n)$ duke shfrytëzuar shprehjen

$$b_i = i + 2a_i^2 - 2$$

```
#include <iostream>
using namespace std;
int main()
{
    int const n=5;
    int i,A[n]={1,4,-2,6,-1},B[n];
    for (i=0;i<n;i++)
    {
        B[i]=i+2*(A[i]*A[i])-2;
        cout << "B["
        << i
        << "]= "
        << B[i]
        << "\n";
    }
    system ("pause");
    return 0;
}
```



Shembull 8.3

- Të gjendet anëtari minimal dhe maksimal i vektorit të dhënë:

$A(n) = \{21, 34, -12, 6, -1, 6, 1\}$

```
#include <iostream>
using namespace std;
int main()
{
    const int m=7;
    int i, a[m]={21,34,-12,6,-1,6,1};
    int min = a[0];
    int max=a[0];
    for (int i = 0; i < m; i++)
    {
        if (a[i] < min)
        {
            min = a[i];
        }
        if (a[i] > max)
        {
            max = a[i];
        }
    }
    cout<<"Anëtari minimal : " <<min<<endl;
    cout<<"Anëtari maksimal : " <<max<<endl;

    system("pause");
    return 0;
}
```



Sortimi i vektorit

A 5 1 12 -5 16 → **Vektori i pasortuar**

1 5 1 12 -5 16 → **5 > 1, ndërro**

1 5 12 -5 16 → **5 < 12, po**

1 5 12 -5 16 → **12 > -5, ndërro**

1 5 -5 12 16 → **12 < 16, po**

2 1 5 -5 12 16 → **1 < 5, po**

1 5 -5 12 16 → **5 > -5, ndërro**

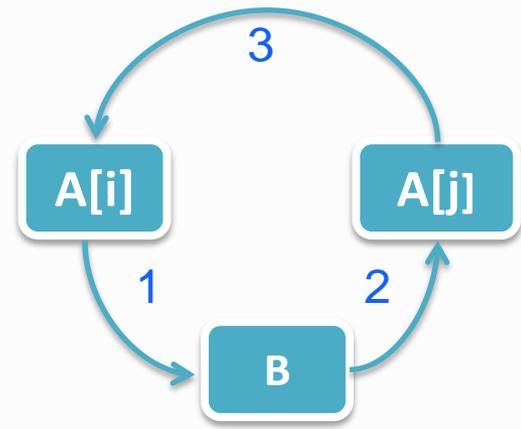
1 -5 5 12 16 → **5 < 12, po**

3 1 -5 5 12 16 → **1 > -5, ndërro**

-5 1 5 12 16 → **1 < 5, po**

4 -5 1 5 12 16 → **-5 < 1, po**

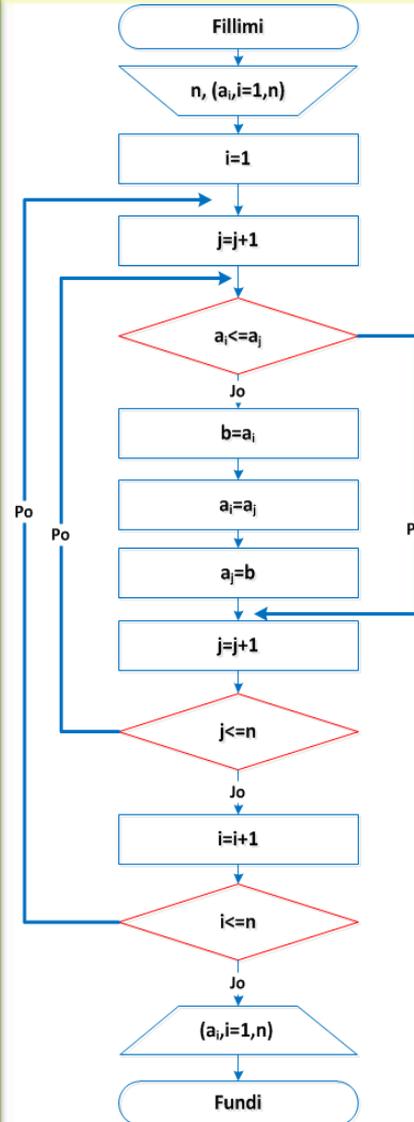
B -5 1 5 12 16 → **Vektori i sortuar**





Shembull 8.4

- Të vizatohet bllok-diagrami dhe të sortohen anëtarët e vektorit $A(m)$ prej anëtarit me vlerë më të vogël kah anëtari me vlerë më të madhe: $A[m]=\{5,1,12,-5,16\}$



```
#include <iostream>
using namespace std;

int main()
{
    int const n=7;
    int A[n]={5,1,12,-5,16},i,j,b;
    for (i=0;i<n-1;i++)
        for (j=i+1;j<n;j++)
            if (A[i]<=A[j])
            {
            }
            else
            {
                b=A[i];
                A[i]=A[j];
                A[j]=b;
            }
    cout << "A=[ ";
    for (i=0;i<n;i++)
        cout << A[i]
        << " ";
    cout << "]"
    << "\n";
    system("pause");
    return 0;
}
```



Shembull 8.5

- Të formohet matrica $M(m,n)$. Vlerat e anëtarëve të mbushen në mënyrë të rastësishme prej 0-99 duke përdorur funksionin `rand()`. Më pas të llogaritet shuma e anëtarëve të matricës dhe të gjendet anëtari më i madh në secilin rresht.

```
#include <iostream>
#include <iomanip>
#include <ctime> //përdoret për srand dhe rand
using namespace std;

int main()
{
    const int m=8, n=10;
    int matrica[m][n], i=0, j=0, max, s=0;
    srand(time(NULL));

    for(i=0;i<m;i++)
    {
        for(j=0;j<n;j++)
        {
            matrica[i][j]=(rand()%100);
            cout<<setw(4)<< matrica[i][j];
            s+=matrica[i][j];
        }
        cout<<endl;
    }
    cout<<endl
    <<"Shuma e anëtarëve të matricës s="
    <<s
    <<endl;
    for (i = 0; i < m; i++)
    {
        max = matrica[i][0];
        for (j = 1; j < n; j++)
            if (max < matrica[i][j])
                max = matrica[i][j];
        cout<< "Anëtari më i madh në rreshtin "
        << i + 1 << " = "
        << max
        << endl;
    }

    system ("pause");
    return 0;
}
```



Shembull 8.6

- Të formohet matrica katrore $A(m,m)$. Dy rreshtat dhe shtyllat e para dhe dy rreshtat dhe shtyllat e fundit të jenë me 1-sha, kurse pjesa tjetër me zero.

```
1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 0 0 0 0 0 0 1 1
1 1 1 1 1 1 1 1 1 1
1 1 1 1 1 1 1 1 1 1
```

```
#include<iostream>
#include<iomanip>
using namespace std;
int main()
{
    const int m=8;
    int i=0,j=0,a[m][m];
    i=0;

    while(i<m)
    {
        j=0;
        while(j<m)
        {
            if(i==0||i==1||i==m-1||i==m-2||
                j==0||j==1||j==m-1||j==m-2)
                a[i][j]=1;
            else
                a[i][j]=0;
            cout<<setw(4)<<a[i][j];
            j++;
        }
        i++;
        cout<<endl;
    }

    system("Pause");
    return 0;
}
```

Shembull 8.7

- Të formohet matrica $A(m,m)$ duke shfrytëzuar vektorin $B(m)$. Vlerat e anëtarëve të vektorit të vendosen sipas skemës së mëposhtme:

b_1	b_2	b_3	b_4	...	b_m
1	1	1	1	...	1
2	2	2	2	...	2
3	3	3	3	...	3
...
m	m	m	m	...	m

```
//Vendosja e vektorit në pozitë të caktuar
#include <iostream>
using namespace std;

int main()
{
    const int m=5;
    int i,j,A[m][m],B[m]={1,2,3,4,5};
    for (i=0;i<m;i++)
        for (j=0;j<m;j++)
            if (i==0)
                A[i][j]=B[j];
            else
                A[i][j]=i;
    cout<< "Matrica e formuar\n";
    for (i=0;i<m;i++)
    {
        for (j=0;j<m;j++)
        {
            cout.width(5);
            cout << A[i][j];
        }
        cout << "\n";
    }
    system("pause");
    return 0;
}
```

Shembull 8.8

- Të formohet matrica katrore $A(m,m)$ duke shfrytëzuar vektorët $A(m)$, $B(m)$, $C(m)$, $D(m)$, $E(m)$ dhe $F(m)$. Matrica të formohet sipas skemës së mëposhtme:

- $A[m]=\{6,-1,-6,4,5,1\}$;
- $B[m]=\{5,-2,-5,3,5,2\}$;
- $C[m]=\{4,-3,-4,4,5,3\}$;
- $D[m]=\{3,-4,-3,4,5,4\}$;
- $E[m]=\{2,-5,-2,4,5,5\}$;
- $F[m]=\{1,-6,-1,4,5,6\}$;

a_1	a_2	a_3	a_4	a_5	a_6
b_1	b_2	b_3	b_4	b_5	b_6
c_1	c_2	c_3	c_4	c_5	c_6
d_1	d_2	d_3	d_4	d_5	d_6
e_1	e_2	e_3	e_4	e_5	e_6
f_1	f_2	f_3	f_4	f_5	f_6



Zgjidhja e shembullit 8.8

```
//Formimi i matrices prej shumë vektoreve
```

```
#include <iostream>  
using namespace std;
```

```
int main()  
{
```

```
    const int m=6;
```

```
    int A[m]={6, -1, -6,4,5,1};
```

```
    int B[m]={5, -2, -5,4,5,2};
```

```
    int C[m]={4, -3, -4,2,5,3};
```

```
    int D[m]={3, -4, -3,4,5,4};
```

```
    int E[m]={2, -5, -2,4,5,5};
```

```
    int F[m]={1, -6, -1,4,5,6};
```

```
    int i,j,Z[m][m];
```

```
    for (i=0;i<m;i++)
```

```
        for (j=0;j<m;j++)
```

```
        {
```

```
            switch(i)
```

```
            {
```

```
                case 0:
```

```
                    Z[i][j]=A[j];
```

```
                    break;
```

```
                case 1:
```

```
                    Z[i][j]=B[j];
```

```
                    break;
```

```
                case 2:
```

```
                    Z[i][j]=C[j];
```

```
                    break;
```

```
                case 3:
```

```
                    Z[i][j]=D[j];
```

```
                    break;
```

```
                case 4:
```

```
                    Z[i][j]=E[j];
```

```
                    break;
```

```
                case 5:
```

```
                    Z[i][j]=F[j];
```

```
                    break;
```

```
            default:
```

```
                Z[i][j]=0;
```

```
                break;
```

```
            }
```

```
        }
```

```
        cout << "Matrica e formuar\n";
```

```
        for (i=0;i<m;i++)
```

```
        {
```

```
            for (j=0;j<m;j++)
```

```
            {
```

```
                cout.width(5);
```

```
                cout << Z[i][j];
```

```
            }
```

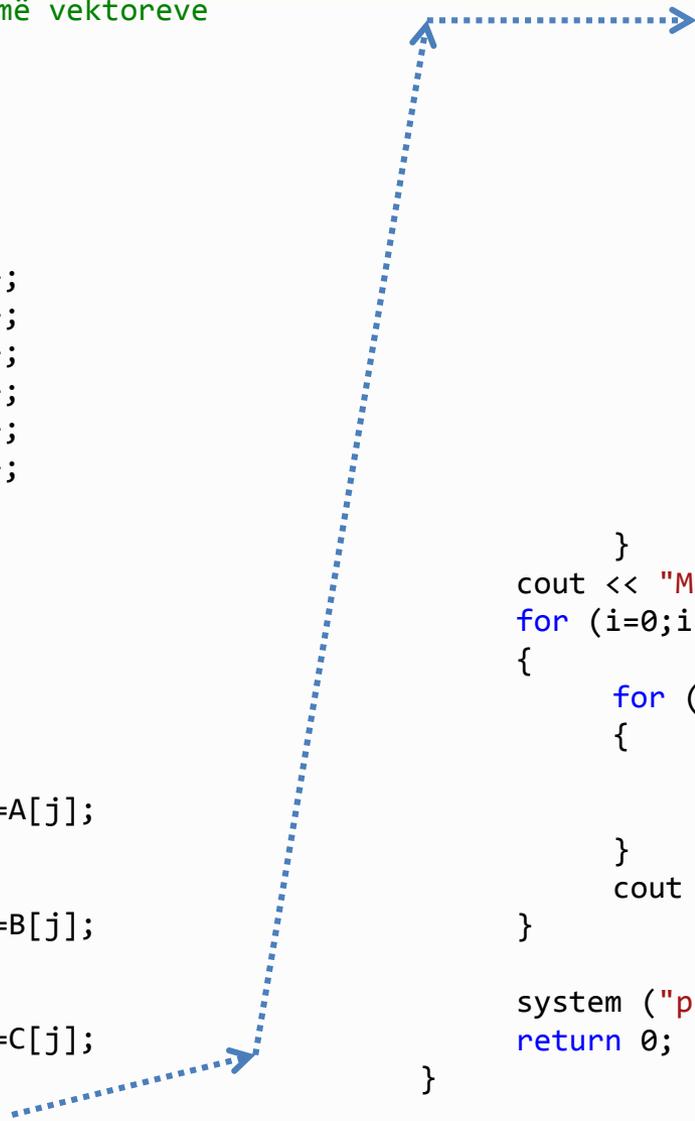
```
            cout << "\n";
```

```
        }
```

```
        system ("pause");
```

```
        return 0;
```

```
    }
```





Shembull 8.9

- Të bashkohen matrica $A(m,m)$ dhe $B(m,m)$ sipas skemës së mëposhtme

```
int A[m][m]={{1,1,1},
              {1,1,1},
              {1,1,1}};
```

```
int B[m][m]={{2,2,2},
              {2,2,2},
              {2,2,2}};
```

a_{00}	...	$a_{0,n-1}$	b_{00}	...	$b_{0,n-1}$
...
$a_{m-1,0}$...	$a_{m-1,n-1}$	$b_{m-1,0}$...	$b_{m-1,n-1}$

```
//kombinimi i dy matricave|
#include <iostream>
using namespace std;

int main()
{
    const int m=3;
    int A[m][m]={{1,1,1},
                 {1,1,1},
                 {1,1,1}};
    int B[m][m]={{2,2,2},
                 {2,2,2},
                 {2,2,2}};

    int i,j,C[m][m*m];

    for (i=0;i<m;i++)
        for (j=0;j<(2*m);j++)
            if (j<m)
                C[i][j]=A[i][j];
            else
            {
                C[i][j]=B[i][j-m];
            }

    cout << "Matrica e formuar" << "\n";
    for (i=0;i<m;i++)
    {
        for (j=0;j<(2*m);j++)
        {
            cout.width(4);
            cout << C[i][j];
        }
        cout << "\n";
    }

    system("pause");
    return 0;
}
```



Shembull 8.10

- Të formohet matrica e tipit `char A(m,n)` dhe të ruhet emrat e personave të mëposhtëm dhe më pas në varësi të indeksit të shtypur të shfaqet emri i personit.

"Robert De Niro"

"Ismail Kadare"

"Karl Gega "

"Aleksandër Moisiu"

"Hasan Prishtina"

```
1 #include <iostream>
2 #include<iomanip>
3 using namespace std;
4
5 int main()
6 {
7     char A[6][50]={"Robert De Niro",
8                  "Ismail Kadare",
9                  "Karl Gega",
10                 "Aleksandër Moisiu",
11                 "Hasan Prishtina"};
12
13     int z=0;
14     cout<<endl
15         <<"Shtyp nje numer nga 1 deri në 5 : ";
16     cin >>z;
17
18     if (z>=1 && z<=5)
19         cout<<endl
20             <<"Ju keni zgjedhur " <<A[z-1];
21     else
22         cout<<endl
23             <<"Zgjedhja juaj duhet te jete ne mes 1 dhe 5 ";
24     cout<<endl;
25     system("pause");
26     return 0;
27 }
```

