

9-laboratoriya ishi.

Mavzu: Ikki o'lchovli massivlar. Ikki o'lchovli massivlar uchun massiv operatorlaridan foydalanib masalalarga dastur tuzish.

C ++ da bir o'lchovli massivlardan tashqari, ko'p o'lchovli massivlar ham mavjud. O'z navbatida, bunday massivlarning elementlari massiv bo'lib, unda elementlar ham massiv bo'lishi mumkin. Masalan, ikki o'lchovli massivni aniqlaymiz:

```
int numbers [3] [2];
```

Bunday massiv uchta elementdan iborat bo'lib, har bir element ikkita elementdan iborat massivdir. Bunday Massivga quyidagicha qiymat beramiz:

```
int numbers [3] [2] = {{1, 2}, {4, 5}, {7, 8}};
```

Ichki qavslar har bir qism massiv uchun elementlarni belgilaydi. Bunday massivni jadval shaklida ham ifodalash mumkin:

```
1 2
```

```
4 5
```

```
7 8
```

Qiymat berishda ichki qavslarni yozmaslik mumkin:

```
int numbers [3] [2] = {1, 2, 4, 5, 7, 8};
```

Iki o'lchovli massiv elementlariga murojaat qilish uchun ikkita indeks kerak:

```
int numbers [3] [2] = {{1, 2}, {3, 4}, {5, 6}};
```

```
cout << numbers [1] [0] << endl; // 3
```

```
numbers [1] [0] = 12; // elementni o'zgartirish
```

```
cout << numbers[1] [0] << endl; // 12
```

1-misol.

```
#include <iostream>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
const int rows = 3, columns = 2;
```

```
int numbers[rows][columns] = { {1, 2}, {3, 4}, {5, 6} };
```

```
for(int i=0; i < rows; i++)
```

```
{
```

```
for(int j=0; j < columns; j++)
```

```
{
```

```
cout << numbers[i] [j] << "t";
```

```

    }
    cout << endl;
    return 0;
}

```

2-misol. Ikki o'lchovli haqiqiy $B(N,M)$ massiv musbat elementlari yig'indisini hisoblashga dastur tuzing ($N \leq 50$, $M \leq 30$).

Belgilashlar: S –massiv elementlari yig'indisi; k, l – mos ravishda massivning satr va ustunlar soni; n, m – massiv satr va ustunlarining maksimal soni, i –satr raqami, j – ustun raqami bo'lsin.

Ikki o'lchovli massiv ishtirok etgan masalalarni yechishda ichma-ich joylashgan sikllarni tashkil qilish maqsadga muvofiqdir. Satr va ustunlarni tashkil qilish uchun i va j o'zgaruvchilarni kiritamiz. I tashqi sikl parametri, j esa ichki sikl parametri bo'lishi lozim. I va j ning har bir qiymatida $b_{ij} > 0$ shart tekshiriladi, agar shart bajarilsa $S = S + b_{ij}$ yig'indi hisoblanadi. Agar $S = 0$ bo'lsa massivning musbat elementlari mavjud emas.

```

#include <iostream>
using namespace std;
int main()
{ const int n=2, m=3;
  int i,j,k,l;
  float s;
  float b[n][m];
  s=0;
  cout<< "b massivni kiriting \n";
  for (i=0;i<n;i++)
  for (j=0;j<m;j++)
  {
    cin>>b[i][j];
    if (b[i][j]>0) s=s+b[i][j];
  }
  cout<<"s= \t"<<s;
  if (s=0) cout<<"b massivning musbat elementlari yo`q";
  return 0;
}

```

3-misol. Haqiqiy ikki o'lchovli massivning eng katta elementini toping.

Belgilashlar: b – massiv nomi; , $i=20$, $j=15$ – mos ravishda massivning satrlar va ustunlarining maksimal soni; k va l esa satr va ustunlar soni.

Dastlab b massivning barcha elementlari qiymatlarini kiritamiz. Massiv birinchi elementini eng katta element deb faraz qilamiz: $b_{max} = b_{11}$. Keyin esa

ichma-ich siklning ichida $b_{ij} > b_{max}$ shart tekshiriladi, agar bu shart bajarilsa $b_{max}=b_{ij}$ almastirish bajariladi.

```
#include <iostream>
using namespace std;
int main()
{
    const int n=2,m=5;
    int i,j,k,l;
    float bmax;
    float b[n][m];
    cout<< "b massivni kiriting \n";
    for (i=0;i<n;i++)
    for (j=0;j<m;j++)
        cin>>b[i][j];
    bmax=b[1][1];
    for (i=0;i<n;i++)
    for (j=0;j<m;j++)
        if (b[i][j]>bmax) bmax=b[i][j];
    cout<<"bmax="<<bmax;
    return 0;
}
```

4-misol. Ikki o'lchovli haqiqiy $B(N,M)$ massivning bosh diagonalining quyi qismida joylashgan elementlari yig'indisini hisoblang ($N \leq 50, M \leq 30$).

Dasturning ko'rinishi:

```
#include <iostream>
using namespace std;
int main()
{
    const int n=5, m=3;
    int i,j,k,l;
    float s;
    float b[n][m];
    s=0;
    cout<< "b massivni kiriting \n";
    for (i=0;i<n;i++)
    for (j=0;j<m;j++)
    {
        cin>>b[i][j];
        if (i>j) s=s+b[i][j];
    }
}
```

```
}  
cout<<"s="<<s;  
return 0;  
}
```