

TABLOURI BIDIMENSIONALE

Declararea matricelor

Să ne reamintim definiția tabloului:

Definiție:

Un tablou este o colecție de date de același tip, memorate într-o zonă contiguă de memorie.

Un tablou care conține două sau mai multe linii și două sau mai multe coloane, poartă numele de *tablou bidimensional* sau *matrice*.

Declararea unui tablou bidimensional:

Pentru un tablou bidimensional, la declarare trebuie precizate două dimensiuni (**numărul de linii** și **numărul de coloane**).

tip **nume**[**număr_linii**][**număr_coloane**]

unde: **tip** poate fi orice tip din C

nume este un identificador C reprezintă numele tabloului

numar_linii este o expresie constantă întreagă pozitivă, reprezentând numărul de linii ale tabloului

numar_coloane este o expresie constantă întreagă pozitivă, reprezentând numărul de coloane ale tabloului.

Exemplu

int a[20][30]

Matricea permite memorarea a maxim 20 de linii si maxim 30 de coloane (600 elemente)

Accesarea elementelor unui tablou bidimensional:

Fiecare element al tabloului bidimensional este bine determinat de doi indici: **indicele de linie** și **indicele de coloană**.

Indicele de linie este o expresie întreagă pozitivă și poate lua valori în intervalul

[0 , număr_linii - 1]

Indicele de coloană este o expresie întreagă pozitivă și poate lua valori în intervalul **[0 , număr_coloane - 1]**

Exemplu: Dacă dorim să utilizăm o matrice cu trei linii și două coloane, cu elemente numere întregi cu numele **a** vom declara:

int a[3][2];

a	Coloana 0	Coloana 1
linia 0	a[0][0]	a[0][1]
linia 1	a[1][0]	a[1][1]
linia 2	a[2][0]	a[2][1]

Dacă dorim să atribuim valoarea 3 elementului de pe linia 1 coloana 0, vom scrie:
a[1][0]=3.

Operații elementare cu matrice

Citirea matricei:

```
//Citirea se realizează parcurgând matricea linie cu linie
cout<<"dati nr de linii:";cin>>n;
cout<<"dati nr de coloane:";cin>>m;
cout<<"dati elem:";
for(i=0;i<n;i++)
    for(j=0;j<m;j++)
        {cout<<"a["<<i<<"]["<<j<<"]=";
          cin>>a[i][j];
        }
```

Afișarea matricei:

```
// afișarea se realizează linie cu linie
int i,j;
for(i=0;i<n;i++)
    {for(j=0;j<m;j++)
      cout<<a[i][j]<<' ';
      cout<<endl;
    }
```

În operațiile de citire și afișare, s-a parcurs matricea pe linii, adică elementele de pe linia de indice 0 (a[0][0], a[0][1], a[0][2], ... a[0][m-1]), apoi elementele de pe linia de indice 1 (a[1][0], a[1][1], a[1][2], ... a[1][m-1]), elementele de pe linia de indice n-1 (a[n-1][0], a[n-1][1], a[n-1][2], ... a[n-1][m-1]).

	coloana 0	coloana 1	...	coloana m-1
linia 0			...	
linia 1			...	
...	...	...	...	...
linia n-1			...	

Matricea poate fi parcursă și pe coloane (coloana 0, coloana 1, ... , coloana m-

1).

În exemplul următor vom calcula sumele elementelor de pe coloanele matricei:

```
int i,j,s;
for (j=0;j<m;j++)
{ s=0;                                // pentru fiecare coloana, se initializeaza suma s cu 0
  for(i=0;i<n;i++)                    // aici parcurgem o coloana
    s=s+a[i][j];
  cout<<" suma elementelor de pe coloana "<<j<<"="<<s<<endl;
}
```

De cele mai multe ori vom folosi numerotarea liniilor și a coloanelor de la 1. Vom avea nevoie la declararea matricei de un spațiu în plus la memorie.

Exemplu

```
int a[21][31]
```

Matricea permite memorarea a maxim 20 de linii si maxim 30 de coloane (600 elemente)

Accesarea elementelor unui tablou bidimensional:

Fiecare element al tabloului bidimensional este bine determinat de doi indici: **indicele de linie** și **indicele de coloană**.

```
int a[4][2];
```

a	Coloana 1	Coloana 2
linia 1	a[1][1]	a[1][2]
linia 2	a[2][1]	a[2][2]
linia 3	a[3][1]	a[3][2]

Operații elementare cu matrice

Citirea matricei:

```
//Citirea se realizează parcurgând matricea linie cu linie
cout<<"dati nr de linii:"<<cin>>n;
cout<<"dati nr de coloane:"<<cin>>m;
cout<<"dati elem:";
for(i=1;i<=n;i++)
  for(j=1;j<=m;j++)
    { cout<<"a["<<i<<"]["<<j<<"]="<<endl;
      cin>>a[i][j];
    }
```

Afișarea matricei:

```
// afișarea se realizează linie cu linie
int i,j;
for(i=1;i<=n;i++)
    {for(j=1;j<=m;j++)
        cout<<a[i][j]<<' ';
        cout<<endl;
    }
```

Aplicații:

Se citesc două numere $n, m \in [1, 20]$, și apoi m linii și n coloane ale matricei a .

1. Calculați și afișați maximul din matrice.
2. Calculați și afișați maximul fiecărei linii din matrice.
3. Calculați și afișați maximul fiecărei coloane din matrice.
4. Calculați și afișați suma elementelor pare din matrice.
5. Calculați și afișați media elementelor pare din matrice.

Calculați și afișați maximul din matrice.

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

int main()
{
    int i,j,a[21][21],m,n,maxi=-9999999;
    cin>>m>>n;
    for (i=1;i<=m;i++)
        for (j=1;j<=n;j++)
            cin>>a[i][j];
    for (i=1;i<=m;i++)
        for (j=1;j<=n;j++)
            if (a[i][j]>maxi) maxi=a[i][j];
    cout<<maxi;
}
```

Calculați și afișați maximul fiecărei linii din matrice.

```
#include <iostream>
using namespace std;
int main()
{
    int i,j,a[21][21],m,n,maxi=-9999999;
    cin>>m>>n;
    for (i=1;i<=m;i++)
        for (j=1;j<=n;j++)
            cin>>a[i][j];
    for (i=1;i<=m;i++) //parcurgem matricea pe linii
    {
        maxi=-9999999;
        for (j=1;j<=n;j++)
            if (a[i][j]>maxi) maxi=a[i][j];
        cout<<"pe linia "<<i<<" maximul="<<maxi<<endl;
    }
}
```

Calculați și afișați maximul fiecărei coloane din matrice.

```
#include <iostream>
using namespace std;
int main()
{
    int i,j,a[21][21],m,n,maxi=-9999999;
    cin>>m>>n;
    for (i=1;i<=m;i++)
        for (j=1;j<=n;j++)
            cin>>a[i][j];
    for (j=1;j<=n;j++) //parcurgem matricea pe coloane
    {
        maxi=-9999999;
        for (i=1;i<=m;i++)
            if (a[i][j]>maxi) maxi=a[i][j];
        cout<<"pe coloana "<<j<<" maximul="<<maxi<<endl;
    }
}
```

Calculați și afișați suma elementelor pare din matrice.

```
#include <iostream>
using namespace std;
int main()
{
    int i,j,a[21][21],m,n,s=0;
    cin>>m>>n;
    for (i=1;i<=m;i++)
        for (j=1;j<=n;j++)
            cin>>a[i][j];
    for (i=1;i<=m;i++)
        for (j=1;j<=n;j++)
            if (a[i][j]%2==0) s=s+a[i][j];
    cout<<s;
}
```

Calculați și afișați media elementelor pare din matrice.

```
#include <iostream>
using namespace std;
int main()
{
    int i,j,a[21][21],m,n,nr=0;
    float s=0;
    cin>>m>>n;
    for (i=1;i<=m;i++)
        for (j=1;j<=n;j++)
            cin>>a[i][j];
    for (i=1;i<=m;i++)
        for (j=1;j<=n;j++)
            if (a[i][j]%2==0)
            {
                s=s+a[i][j];
                nr++;
            }
    if (nr==0) cout<<"NU exista elem pare";
    else cout<<s/nr;
}
```