

## PROBLEME MATRICI

1. Scrieți un program **C++** care citește de la tastatură două numere naturale nenule  $n$  și  $m$  ( $2 \leq m \leq 10$ ,  $2 \leq n \leq 10$ ) și care construiește în memorie și apoi afișează o matrice **A** cu  $n$  linii (numerotate de la 1 la  $n$ ) și  $m$  coloane (numerotate de la 1 la  $m$ ) cu proprietatea că fiecare element **A**<sub>*ij*</sub> memorează cea mai mică dintre valorile indicilor  $i$  și  $j$  ( $1 \leq i \leq n$ ,  $1 \leq j \leq m$ ).

Matricea se va afișa pe ecran, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

**Exemplu:** pentru  $n=4$  și  $m=5$  se va afișa matricea alăturată.

```
1 1 1 1 1
1 2 2 2 2
1 2 3 3 3
1 2 3 4 4
```

2. Scrieți un program **C++** care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 24$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane ale cărei elemente vor primi valori după cum urmează:

- elementele aflate pe diagonala principală a matricei vor primi valoarea **0**
- elementele de pe prima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea  $n$
- elementele de pe a doua coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea  $n-1$
- ...
- elementele de pe ultima coloană, cu excepția celui aflat pe diagonala principală vor primi valoarea **1**

Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în exemplu).

**Exemplu:** pentru  $n=4$  se va afișa matricea alăturată.

```
0 3 2 1
4 0 2 1
4 3 0 1
4 3 2 0
```

3. Scrieți un program **C++** care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 24$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane ale cărei elemente vor primi valori după cum urmează:

- elementele aflate pe diagonala secundară a matricei vor primi valoarea **0**
- elementele de pe prima linie, cu excepția celui aflat pe diagonala secundară vor primi valoarea  $n$
- elementele de pe a doua linie, cu excepția celui aflat pe diagonala secundară vor primi valoarea  $n-1$
- ...
- elementele de pe ultima linie, cu excepția celui aflat pe diagonala secundară vor primi valoarea **1**

Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii (ca în exemplu).

**Exemplu:** pentru  $n=4$  se va afișa matricea alăturată.

```
4 4 4 0
3 3 0 3
2 0 2 2
0 1 1 1
```

4. Scrieți un program **C++** care citește de la tastatură două numere naturale  $n$  și  $m$  ( $2 \leq m \leq 10$ ,  $2 \leq n \leq 10$ ) și care construiește în memorie și apoi afișează o matrice **A** cu  $n$  linii (numerotate de la 1 la  $n$ ) și  $m$  coloane (numerotate de la 1 la  $m$ ) cu proprietatea că fiecare element **A**<sub>*ij*</sub> memorează cea mai mare dintre valorile indicilor  $i$  și  $j$  ( $1 \leq i \leq n$ ,  $1 \leq j \leq m$ ).

Matricea se va afișa pe ecran, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

**Exemplu:** pentru  $n=4$  și  $m=5$  se va afișa matricea alăturată.

```
1 2 3 4 5
2 2 3 4 5
3 3 3 4 5
4 4 4 4 5
```

5. Scrieți un program **C++** care citește de la tastatură două numere naturale  $n$  și  $p$  ( $2 \leq n \leq 20$ ,  $1 \leq p \leq 20$ ) și construiește în memorie un tablou bidimensional cu  $n$  linii și  $p$  coloane. Tabloul va fi construit astfel încât, parcurgând tabloul linie cu linie de sus în jos și fiecare linie de la stânga la dreapta, să se obțină șirul primelor  $n \cdot p$  pătrate perfecte impare, ordonat strict crescător, ca în exemplu. Tabloul astfel construit va fi afișat pe ecran, fiecare linie a tabloului pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii.

**Exemplu:** pentru  $n=2$ ,  $p=3$ , programul va afișa tabloul alăturat:

```
1 9 25
49 81 121
```

6. Se consideră tabloul bidimensional cu  $n$  linii și  $n$  coloane ce conține numere naturale cu cel mult patru cifre fiecare. Scrieți programul **C++** care citește de la tastatură numărul natural  $n$  ( $2 \leq n \leq 23$ ) și cele  $n \cdot n$  elemente ale tabloului și apoi afișează pe ecran elementele primului pătrat concentric, separate prin câte un spațiu. Pătratul este parcurs în sensul acelor de ceasornic începând din colțul său stânga-sus, ca în exemplu. Primul pătrat concentric este format din prima și ultima linie, prima și ultima coloană a tabloului.

Exemplu: pentru  $n=5$  și tabloul alăturat, se va afișa:

1 2 3 4 5 1 6 2 7 6 5 4 3 7 2 6

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

7. Scrieți un program **C++** care citește de la tastatură două numere naturale  $n$  și  $p$  ( $1 < n \leq 20$ ,  $1 \leq p \leq 20$ ) și construiește în memorie un tablou bidimensional cu  $n$  linii și  $p$  coloane. Tabloul va fi construit astfel încât parcurgând matricea de la prima linie către ultima și fiecare linie de la stânga la dreapta să se obțină șirul primelor  $n \cdot p$  pătrate perfecte pare ordonat strict crescător. Tabloul astfel construit va fi afișat pe ecran, fiecare linie a tabloului pe câte o linie a ecranului, cu câte un spațiu între elementele fiecărei linii.

Exemplu: pentru  $n=2$ ,  $p=3$ , programul va afișa tabloul alăturat:

0 4 16  
36 64 100

8. Scrieți un program **C++** care citește de la tastatură două valori naturale nenule  $m$  și  $n$  ( $m \leq 10$ ,  $n \leq 10$ ) și apoi  $m \cdot n$  numere naturale nenule cu cel mult 4 cifre fiecare, reprezentând elementele unei matrice cu  $m$  linii și  $n$  coloane. Programul determină apoi valorile minime de pe fiecare linie a matricei și le afișează pe o linie a ecranului separate prin câte un spațiu.

Exemplu: pentru  $m=3$ ,  $n=5$  și matricea de jos se afișează pe ecran valorile 3 6 2.

3 6 5 4 7  
9 6 12 9 10  
5 13 7 2 3

(cea mai mică valoare de pe prima linie a matricei este 3, cea mai mică valoare de pe linia a doua este 6, cea mai mică valoare de pe linia a treia este 2).

9. Scrieți un program **C++** care citește de la tastatură un număr natural  $n$  ( $2 < n < 50$ ) și apoi construiește în memorie o matrice cu  $n$  linii și  $n$  coloane, numerotate de la 1 la  $n$ , ale cărei elemente primesc valori după cum urmează:

- elementele aflate pe diagonala secundară sunt toate nule;
- elementele de pe linia  $i$  ( $1 \leq i \leq n$ ), aflate deasupra diagonalei secundare, au valoarea egală cu  $i$ ;
- elementele de pe coloana  $n-i+1$  ( $1 \leq i \leq n$ ), aflate sub diagonala secundară, au valoarea egală cu  $i$ .

Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

1 1 1 0  
2 2 0 1  
3 0 2 1  
0 3 2 1

Exemplu: pentru  $n=4$  se va afișa matricea alăturată.

10. Scrieți un program **C++** care citește de la tastatură un număr natural  $n$  ( $2 < n < 50$ ) și apoi construiește în memorie o matrice cu  $n$  linii și  $n$  coloane, numerotate de la 1 la  $n$ , ale cărei elemente primesc valori după cum urmează:

- elementele aflate pe diagonala secundară sunt toate nule;
- elementele aflate deasupra diagonalei secundare sunt toate 1;
- elementele aflate sub diagonala secundară sunt toate 2.

Programul afișează pe ecran matricea construită, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

Exemplu: pentru  $n=4$  se va afișa matricea alăturată.

1 1 1 0  
1 1 0 2  
1 0 2 2  
0 2 2 2

11. Scrieți un program **C++** care citește de la tastatură un număr natural nenul cu exact **4** cifre, construiește în memorie și afișează apoi pe ecran o matrice având **4** linii și **4** coloane, completată astfel: elementele de pe prima coloană a matricei vor fi toate egale cu cifra unităților numărului dat, elementele de pe a doua coloană a matricei vor fi toate egale cu cifra zecilor numărului dat, elementele de pe a treia coloană a matricei vor fi toate egale cu cifra sutelor numărului dat, iar elementele de pe a patra coloană a matricei vor fi toate egale cu cifra miilor numărului dat.

Matricea va fi afișată pe ecran, câte o linie a matricei pe câte o linie a ecranului, iar elementele fiecărei linii vor fi separate prin câte un spațiu.

**Exemplu:** dacă se citește numărul **1359**, matricea construită va fi cea alăturată.

```
9 5 3 1
9 5 3 1
9 5 3 1
9 5 3 1
```

12. Scrieți un program **C++** care citește de la tastatură un număr natural cu exact **5** cifre și construiește în memorie o matrice cu **5** linii și **5** coloane, numerotate de la **1** la **5**, formată astfel:

- elementele de pe linia **1**, au toate valoarea egală cu cifra unităților numărului citit;

- elementele de pe linia **2**, au toate valoarea egală cu cifra zecilor numărului citit;

- elementele de pe linia **3**, au toate valoarea egală cu cifra sutelor;

- elementele de pe linia **4**, au toate valoarea egală cu cifra miilor;

- elementele de pe linia **5**, au toate valoarea egală cu cifra zecilor de mii.

```
1 1 1 1 1
3 3 3 3 3
7 7 7 7 7
8 8 8 8 8
2 2 2 2 2
```

Programul afișează pe ecran matricea astfel construită, câte o linie a matricei pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu.

**Exemplu:** dacă se citește numărul **28731** matricea construită va fi cea alăturată.

13. Scrieți un program **C++** care citește de la tastatură un număr natural **n** ( $2 < n < 16$ ), construiește în memorie și afișează pe ecran o matrice cu **n** linii și **n** coloane, în care elementele de pe cele două diagonale sunt egale cu **4**, iar restul elementelor sunt egale cu **3**. Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii.

**Exemplu:** pentru **n=5** se va afișa matricea alăturată.

```
4 3 3 3 4
3 4 3 4 3
3 3 4 3 3
3 4 3 4 3
4 3 3 3 4
```

14. Scrieți un program **C++** care citește de la tastatură un număr natural **n** ( $2 < n < 20$ ), construiește în memorie și afișează pe ecran o matrice cu **n** linii și **n** coloane, în care fiecare element de pe diagonala secundară are valoarea **n**, fiecare element aflat deasupra diagonalei secundare este mai mic cu o unitate decât vecinul aflat pe aceeași linie în dreapta lui și fiecare element aflat sub diagonala secundară este mai mare cu o unitate decât vecinul aflat pe aceeași linie în stânga lui. Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii.

**Exemplu:** pentru **n=5** se va afișa matricea alăturată.

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

15. Scrieți un program **C++** care citește de la tastatură un număr natural **n** ( $2 < n < 20$ ), construiește în memorie și afișează pe ecran o matrice cu **n** linii și **n** coloane, numerotate de la **1** la **n**. Fiecare element din matrice aflat pe o linie impară va fi egal cu numărul liniei pe care se află și fiecare element aflat pe o linie pară va fi egal cu numărul coloanei pe care se află. Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii.

**Exemplu:** pentru **n=5** se va afișa matricea alăturată.

```
1 1 1 1 1
1 2 3 4 5
3 3 3 3 3
1 2 3 4 5
5 5 5 5 5
```

16. Scrieți un program **C++** care citește de la tastatură un număr natural **n** ( $2 < n < 20$ ), construiește în memorie și afișează pe ecran o matrice cu **n** linii și **n** coloane, numerotate de la **1** la **n**, în care fiecare element aflat pe o coloană impară este egal cu suma dintre numărul liniei și numărul coloanei pe care se află și fiecare element aflat pe o coloană pară este egal cu numărul liniei pe care se află.

Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii.

**Exemplu:** pentru **n=5** se va afișa matricea alăturată.

```
2 1 4 1 6
3 2 5 2 7
4 3 6 3 8
5 4 7 4 9
6 5 8 5 10
```

17. Scrieți un program **C++** care citește de la tastatură un număr natural  $n$  ( $2 < n < 10$ ), construiește în memorie și afișează pe ecran o matrice cu  $n$  linii și  $n$  coloane, numerotate de la 1 la  $n$ , în care fiecare element aflat pe prima linie sau pe prima coloană din matrice este egal cu suma dintre numărul liniei și numărul coloanei pe care se află, iar fiecare dintre celelalte elemente este egal cu suma dintre elementul vecin aflat pe aceeași linie cu el, dar pe coloana din stânga sa și elementul vecin aflat pe aceeași coloană cu el, dar pe linia de deasupra sa.

Elementele matricei vor fi afișate pe ecran, câte o linie a matricei pe câte o linie a ecranului cu câte un spațiu între elementele fiecărei linii.

```

2 3 4 5 6
3 6 10 15 21
4 10 20 35 56
5 15 35 70 126
6 21 56 126 252

```

**Exemplu:** pentru  $n=5$  se va obține matricea alăturată.

18. Scrieți un program **C++** care citește de la tastatură numerele naturale  $m$  și  $n$  din intervalul  $[1, 100]$ , apoi construiește în memorie și afișează pe ecran un tablou bidimensional cu  $m$  linii și  $n$  coloane astfel încât prin parcurgerea acestuia linie cu linie de sus în jos și fiecare linie de la stânga la dreapta, se obțin în ordine descrescătoare toate numerele naturale de la 1 la  $m \cdot n$ , ca în exemplu.

Fiecare linie a tabloului este afișată pe câte o linie a ecranului, elementele aceleiași linii fiind separate prin câte un spațiu.

```

12 11 10
9 8 7
6 5 4
3 2 1

```

**Exemplu:** pentru  $m=4$  și  $n=3$  se va construi și afișa tabloul alăturat.

19. Se consideră un tablou bidimensional cu  $m$  linii și  $n$  coloane ( $1 \leq m \leq 100, 1 \leq n \leq 100$ ), ale cărui elemente aparțin mulțimii  $\{0, 1, 2\}$ . Scrieți un program **C++** care citește de la tastatură valorile  $m$ ,  $n$  și elementele tabloului și care afișează pe ecran numerele de ordine ale coloanelor pentru care produsul elementelor situate pe ele, este maxim. Liniile și coloanele tabloului se numerotează de la 1 la  $m$  respectiv de la 1 la  $n$ . Numerele se vor afișa separate prin câte un spațiu.

```

2 1 1 0
1 1 1 1
2 2 2 1
1 2 1 1

```

**Exemplu:** pentru  $m=4$  și  $n=4$  și tabloul alăturat se va afișa, nu neapărat în această ordine: 1 2

20. Scrieți un program **C++** care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 6$ ) și elementele unui tablou bidimensional  $A$  cu  $n$  linii și  $n$  coloane, care memorează numere naturale mai mici decât 100, și afișează pe ecran (dacă există), separate prin câte un spațiu, elementele din matrice care au proprietatea că sunt egale cu produsul celorlalte elemente aflate pe aceeași coloană.

**Exemplu:** pentru matricea din figura alăturată se afișează, nu neapărat în această ordine, valorile 10 22 ( $10=1 \cdot 5 \cdot 2 \cdot 1$ ;  $22=1 \cdot 11 \cdot 2 \cdot 1$ ).

1	1	5	10	2
6	11	1	1	4
6	2	5	5	3
1	22	12	2	10
16	1	7	1	8

21. Scrieți un program **C++** care citește de la tastatură un număr natural  $n$  ( $2 < n \leq 10$ ) și construiește în memorie o matrice  $A$  cu  $n$  linii și  $n$  coloane în care toate elementele de pe prima linie, prima și ultima coloană au valoarea 1 și oricare alt element  $A_{ij}$  din matrice este egal cu suma a 3 elemente situate pe linia  $i-1$ : primul aflat pe coloana  $j-1$ , al doilea pe coloana  $j$ , iar al treilea pe coloana  $j+1$ , ca în exemplu.

Matricea va fi afișată pe ecran, linie cu linie, numerele de pe aceeași linie fiind separate prin câte un spațiu.

```

1 1 1 1 1
1 3 3 3 1
1 7 9 7 1
1 17 23 17 1
1 41 57 41 1

```

**Exemplu:** pentru  $n=5$ , se afișează matricea alăturată.

22. Scrieți un program **C++** care citește de la tastatură un număr natural  $n$  ( $2 < n \leq 15$ ) și construiește în memorie o matrice  $A$  cu  $n$  linii și  $n$  coloane în care orice element aflat pe prima linie sau pe prima coloană are valoarea 1 și oricare alt element  $A_{ij}$  din matrice este egal cu suma a două elemente din matrice, primul aflat pe linia  $i$  și pe coloana  $j-1$  iar cel de-al doilea pe coloana  $j$  și pe linia  $i-1$ . Matricea va fi afișată pe ecran, linie cu linie, numerele de pe aceeași linie fiind separate prin câte un spațiu.

**Exemplu:** pentru  $n=4$ , se obține matricea alăturată.

1	1	1	1
1	2	3	4
1	3	6	10
1	4	10	20

**23.** Scrieți un program **C++** care construiește în memorie o matrice cu **10** linii și **7** coloane ale cărei elemente sunt numere întregi (cu maximum **3** cifre fiecare), citite de la tastatură, și afișează pe ecran, suma tuturor elementelor situate pe conturul matricei determinat de prima și ultima linie respectiv prima și ultima coloană a acestei matrice.

**24.** Scrieți programul **C++** care citește de la tastatură un număr natural **n** ( $n \leq 20$ ), construiește în memorie și afișează pe ecran, matricea cu **n** linii și **n** coloane, în care se vor memora în ordinea strict crescătoare a valorii, pe linii și coloane, primele **n** numere naturale nenule, pare, care nu sunt divizibile cu **3**. Fiecare linie a matricei se va afișa pe câte o linie a ecranului, cu elementele de pe aceeași linie separate prin câte un spațiu.

**Exemplu:** pentru **n=4** se va construi și afișa matricea alăturată.

```
 2  4  8 10
14 16 20 22
26 28 32 34
38 40 44 46
```

25. Scrieți programul C++ care citește de la tastatură un număr natural  $n$  ( $n \leq 50$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane, ale cărei elemente sunt numere întregi citite de la tastatură. Pentru fiecare coloană a matricei, în ordine, programul afișează pe ecran cel mai mic număr de pe respectiva coloană. Numerele afișate vor fi separate prin câte un spațiu.

**Exemplu:** pentru  $n=4$  și matricea alăturată, se vor afișa pe ecran valorile: -7 18 -10 2.

```
122 103 5 10
-7 18 -10 2
107 999 59 4
1 200 100 7
```

### Laborator nr. 5 - Tablouri bidimensionale (matrici)

#### Problema 1:

Sa se scrie un program care calculeaza produsul a doua matrici (tablouri bidimensionale)  $A_{n \times m}$ ,  $B_{m \times p}$ .

```
#include <iostream.h>
void main(void)
{
    int n,m,p,i,j,k,a[10][10],b[10][10],c[10][10];
    cout<<"Dati dimensiunile matricei "<<endl;
    cout<<"Dati numarul de linii n = ";cin>>n; cout<<"Dati
    numarul de coloane m = ";cin>>m; for(i=1;i<=n;i++)
        for(j=1;j<=m;j++)
        {
            cout<<"a["<<i<<","<<j<<"]=" ";
            cin>>a[i][j];
        }
    cout<<"Elementele matricei A sunt: "<<endl;
    for(i=1;i<=n;i++)
    {
        for(j=1;j<=m;j++)
            cout<<a[i][j]<<" ";
        cout<<endl;
    }
    cout<<"Dati numarul de linii m = ";cin>>m;
    cout<<"Dati numarul de coloane p = ";cin>>p;
    for(i=1;i<=m;i++)
        for(j=1;j<=p;j++)
        {
            cout<<"b["<<i<<","<<j<<"]=" ";
            cin>>b[i][j];
        }
    cout<<"Elementele matricei B sunt: "<<endl;
    for(i=1;i<=m;i++)
    {
```

```
        for(j=1;j<=p;j++)
            cout<<b[i][j]<<" ";
        cout<<endl;
    }
    for(i=1;i<=n;i++)
        for(j=1;j<=p;j++)
            for(k=1;k<=m;k++)
                c[i][j]=a[i][k]*b[k][j];
    cout<<"Elementele matricei produs "<<endl;
    for(i=1;i<=n;i++)
    {
        for(j=1;j<=p;j++)
            cout<<c[i][j]<<" ";
        cout<<endl;
    }
}
```

Problema 2:

Sa se scrie un program care calculeaza minimul si maximul dintr-o matrice cu n linii si m coloane ( $A_{n \times m}$ ) ( $1 \leq n, m \leq 30$ ).

```
#include <iostream.h>
void main(void)
{
int n,m,min,i,j,max,a[30][30];
cout<<"Dati dimensiunile matricei "<<endl;
cout<<"Dati numarul de linii n = ";cin>>n; cout<<"Dati
numarul de coloane m = ";cin>>m; for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
    {
        cout<<"a["<<i<<","<<j<<"]= ";
        cin>>a[i][j];
    }
cout<<"Elementele matricei A sunt: "<<endl;
for(i=1;i<=n;i++)
{
    for(j=1;j<=m;j++)
        cout<<a[i][j]<<" ";
    cout<<endl;
}
min=a[1][1];max=a[1][1];
for(j=1;j<=m;j++)
{
    if(a[1][j]<min) min=a[1][j];
    if(a[1][j]>max) max=a[1][j];
}
for(i=2;i<=n;i++)
    for(j=1;j<=m;j++)
    {
        if(a[i][j]<min) min=a[i][j];
        if(a[i][j]>max) max=a[i][j];
    }
cout<<"Elementul minim din matrice este
"<<min<<endl; cout<<"Elementul maxim din matrice
este "<<max<<endl;
}
}
```

Problema 3:

Sa se scrie un program care calculeaza transpusa unei matrici  $A_{n \times m}$   
( $1 \leq n, m \leq 30$ ).

```
#include <iostream.h>
void main(void)
{
int n,m,i,j,a[30][30],b[30][30];
cout<<"Dati dimensiunile matricii "<<endl;
cout<<"Dati numarul de linii n = ";cin>>n; cout<<"Dati
numarul de coloane m = ";cin>>m; for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
    {
        cout<<"a["<<i<<","<<j<<"]=" ";
        cin>>a[i][j];
    }
cout<<"Elementele matricii A sunt: "<<endl;
for(i=1;i<=n;i++)
{
    for(j=1;j<=m;j++)
        cout<<a[i][j]<<" ";
    cout<<endl;
}
for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
        b[j][i]=a[i][j];

cout<<"Matricea transpusa este "<<endl;
for(i=1;i<=n;i++)
{
    for(j=1;j<=m;j++)
        cout<<b[i][j]<<" ";
    cout<<endl;
}
}
```

Problema 4:

- Se considera o matrice  $A_{n \times n}$ . Sa se calculeze:
- suma elementelor de pe diagonala principala
  - produsul elementelor de pe diagonala secundara
  - minimele din elementele aflate deasupra, respectiv sub diagonala principala
  - maximele din elementele aflate deasupra, respectiv sub diagonala secundara

```
#include <iostream.h>
void main(void)
{
int n,m,i,j,a[30][30],min1,min2,max1,max2,suma,produs;
cout<<"Dati dimensiunile matricei "<<endl;
cout<<"Dati numarul de linii si de coloane n = ";cin>>n;
for(i=1;i<=n;i++)
    for(j=1;j<=n;j++)
    {
        cout<<"a["<<i<<","<<j<<"]=" ";
        cin>>a[i][j];
    }
cout<<"Elementele matricei A sunt: "<<endl;
for(i=1;i<=n;i++)
{
    for(j=1;j<=n;j++)
        cout<<a[i][j]<<" ";
    cout<<endl;
}
// ===== suma elementelor de pe diagonala pricipala
// ===== suma=0;
for(i=1;i<=n;i++)
    for(j=1;j<=n;j++)
        if(i==j) suma+=a[i][j];
cout<<"Suma elementelor de pe diagonala pricipala este "<<suma<<endl;
// ===== produsul elementelor de pe diagonala secundara
// ===== produs=1;
for(i=1;i<=n;i++)
    for(j=1;j<=n;j++)
        if(i+j==n+1) produs*=a[i][j];
cout<<"Produsul elementelor de pe diagonala secundara este "<<produs<<endl;
// ===== minimul elementelor de deasupra diagonalei
// principale == min1=32768;
for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
        if(i<j)
            if(min1>a[i][j]) min1=a[i][j]; cout<<"Minimul de
deasupra diag. pricipale este "<<min1;
// ===== minimul elementelor de sub diagonala principale
// ===== min2=32768;
for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
        if(i>j)
            if(min2>a[i][j]) min2=a[i][j]; cout<<"Minimul
de sub diag. pricipala este "<<min2;
// == maximul elementelor de deasupra diagonalei
```

```
principale ==== max1=-32768;
```

```
for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
        if(max1<a[i][j]) max1=a[i][j];
cout<<"Maximul de deasupra diag. principale este "<<max1;
// ==== maximul elementelor de sub diagonala principale
==== max2=-32768;
for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
        if(max2<a[i][j]) max2=a[i][j];
cout<<"Minimul de sub diag. principala este "<<max2;

}
```

**Problema 5:**

Se considera o matrice  $A_{n \times m}$  ( $1 \leq n, m \leq 30$ ) cu elemente numere intregi. Sa se determine linia (liniile) din matrice care contine cele mai multe elemente nenule.

```
#include<iostream.h>
void main(void)
{
int a[30][30],n,m,i,j,max,nr;
cout<<"Dati numarul de linii n = ";cin>>n; cout<<"Dati
numarul de coloane m = ";cin>>m; for(i=1;i<=n;i++)
    for(j=1;j<=m;j++)
        {
            cout<<"a["<<i<<","<<j<<"]=" ";
            cin>>a[i][j];
        }
cout<<endl<<"Matricea A are elementele:"<<endl;
for(i=1;i<=n;i++)
{
    for(j=1;j<=m;j++) cout<<a[i][j]<<" ";
    cout<<endl;
}
max=0;
for(i=1;i<=n;i++)
{
    nr=0;
    for(j=1;j<=m;j++)
        if(a[i][j]!=0) nr++;
    if(max<nr) max=nr;
}
for(i=1;i<=n;i++)
{
    nr=0;
    for(j=1;j<=m;j++)
        if(a[i][j]!=0) nr++;
    if(max==nr) cout<<"Linia "<<i<<" are "<<max<<" elemente
nenule"<<endl;
}
}
```

### Probleme propuse spre rezolvare

1. Se considera o matrice  $A_{n \times m}$  ( $1 \leq n, m \leq 30$ ) având componente numere întregi. Se cere să se calculeze suma componentelor de pe marginea matricei.
2. Se considera o matrice  $A_{n \times m}$  ( $1 \leq n, m \leq 30$ ) având componente numere întregi. Să se determine maximumul fiecărei coloane și minimumul fiecărei linii.
3. Să se construiască o matrice pătratică ( $n=m$ ) de dimensiune  $n^2$  ( $1 \leq n \leq 30$ ) cu primele numere pare începând cu 2.
4. (Puncte 5) Fie un tablou bidimensional  $A(n, m)$ . Un punct 5 al acestui tablou este un element  $a[i_0][j_0]$  cu proprietatea:  
$$a[i_0][j_0] = \min \{ a[i_0][j], \text{ unde } 0 \leq j \leq m-1 \} = \max \{ a[i][j_0], \text{ unde } 0 \leq i \leq n-1 \}$$
Scrieți un program C++ care determină punctele 5 (dacă există) ale unui tablou bidimensional.
5. Fiind dat un tablou bidimensional  $A(n, m)$  de numere întregi, să se calculeze și să se afișeze suma elementelor aflate pe marginea (rama) tabloului.

Exemplu: Pentru  $n = 3$  și  $m = 4$  și  $A = \begin{pmatrix} 1 & 2 & 3 & 4 \\ 10 & 11 & 12 & 5 \\ 9 & 8 & 7 & 6 \end{pmatrix}$  se va afișa suma  $1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 10 = 47$ .