

Programarea calculatoarelor

9 C++ Tablouri (partea a II-a)

Adrian Runceanu
www.runceanu.ro/adrian

Curs 9

Capitolul 7. Tablouri

7.1. Generalități. Clasificare

7.2. Tablouri unidimensionale (vectori)

7.3. Tablouri bidimensionale (matrici)

7.4. Tablouri multidimensionale

7.4. Tablouri multidimensionale

Tablourile multidimensionale se pot declara astfel:

```
tip nume_tablou[dim1][dim2][dim3]...[dimn];
```

Compilerul C++ alocă un spațiu de memorie prin înmulțirea numărului de elemente ale fiecărei dimensiuni în parte ($\text{dim}_1, \text{dim}_2, \text{dim}_3, \dots, \text{dim}_n$) cu numărul de octeți ai tipului de date al tabloului respectiv.

7.4. Tablouri multidimensionale

Exemplu:

Declarări de tablouri multidimensionale:

```
int a[10][10][10];    // declararea unui tablou tridimensional cu  
                        // maxim 1000 de elemente (10*10*10),  
                        // fiecare de tip întreg  
float x[5][5][5];    // declararea unui tablou tridimensional cu  
                        // maxim 125 de elemente(5*5*5), fiecare de  
                        // tip real simplă precizie  
char s[20][10][30];  // declararea unui tablou tridimensional cu  
                        // maxim 6000 de elemente(20*10*30),  
                        // fiecare de tip caracter
```

7.4. Tablouri multidimensionale

Exemplu:

Program care afișează numărul de octeți
alocați pentru:

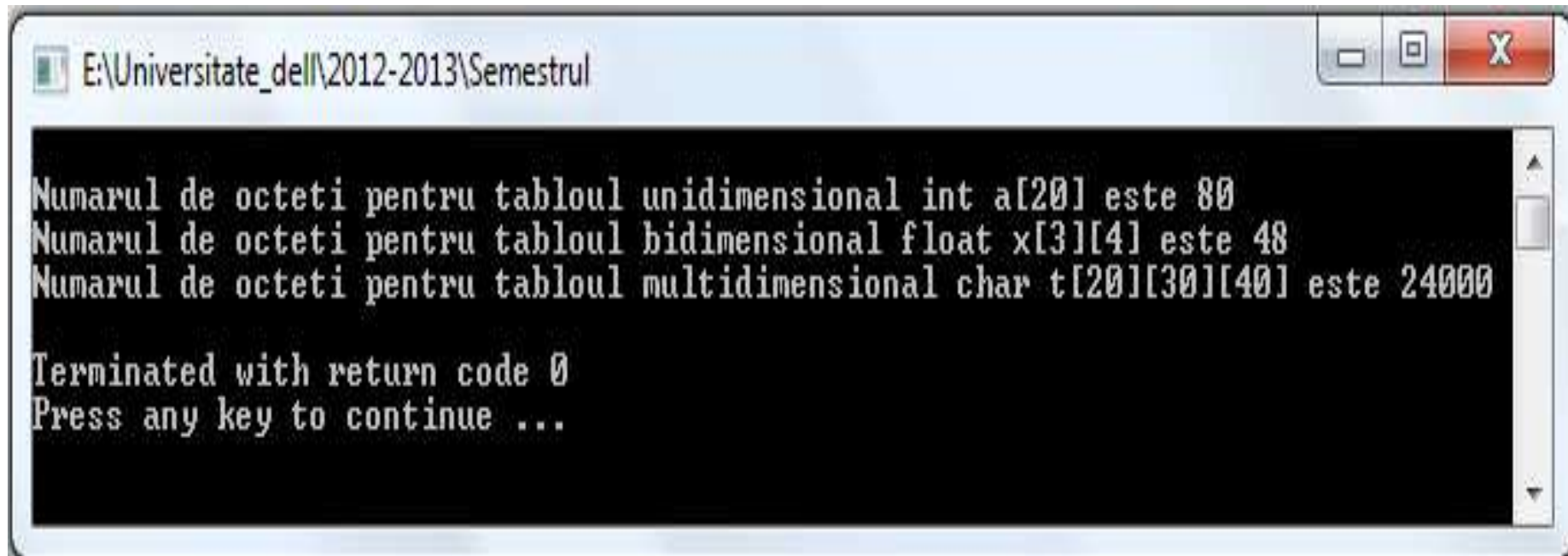
- ✓ un tablou unidimensional
 - ✓ un tablou bidimensional
 - ✓ un tablou multidimensional (cu trei dimensiuni)
- utilizând operatorul **sizeof**:

7.4. Tablouri multidimensionale

```
#include <iostream.h>
int main(void)
{
    int a[20];
    float x[3][4];
    char t[20][30][40];
    cout<<"\nNumărul de octeți pentru tabloul unidimensional
    int a[20] este "<< sizeof(a);
    cout<<"\nNumărul de octeți pentru tabloul bidimensional
    float x[3][4] este "<< sizeof(x);
    cout<<"\nNumărul de octeți pentru tabloul
    multidimensional char t[20][30][40] este "<< sizeof(t);
}
```

7.4. Tablouri multidimensionale

După compilarea și execuția programului, pe ecran vor apărea următoarele valori:



```
E:\Universitate_dell\2012-2013\Semestrul

Numarul de octeti pentru tabloul unidimensional int a[20] este 80
Numarul de octeti pentru tabloul bidimensional float x[3][4] este 48
Numarul de octeti pentru tabloul multidimensional char t[20][30][40] este 24000

Terminated with return code 0
Press any key to continue ...
```


Capitolul 7. Tablouri

7.5. Probleme cu vectori

7.6. Probleme cu matrici

Probleme rezolvate cu vectori

Problema 1:

Enunț:

Se citește un tablou unidimensional cu n ($1 \leq n \leq 100$) componente numere naturale.

Se cere să se construiască și să se afișeze un nou vector cu componentele **pătrate perfecte** ale vectorului inițial.

Probleme rezolvate cu vectori

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

Citirea numarului de
elemente ce vor fi
prelucrate in vector - n

```
int main(void)
{
```

```
    unsigned int x[100], y[100], n, i, j;
```

```
    cout<<"Dati numarul elementele ale tabloului n  
    = ";
```

```
    cin>>n;
```

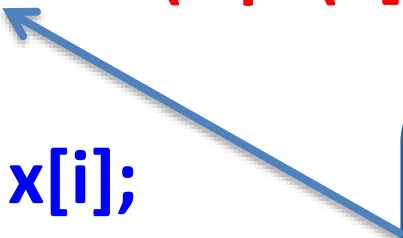
Probleme rezolvate cu vectori

```
cout<<"\nDati elementele tabloului ";  
for(i = 0; i < n; i++)  
{  
    cout<<"x["<<i+1<<"]="";  
    cin>>x[i];  
}
```

Citirea elementelor
si memorarea lor in
vectorul x

Probleme rezolvate cu vectori

```
j = 0;    // numărul de elemente din noul vector
for(i = 0; i < n; i++)
{
    if(sqrt(x[i]) == int(sqrt(x[i])))
    {
        y[j] = x[i];
        j++;
    }
}
```

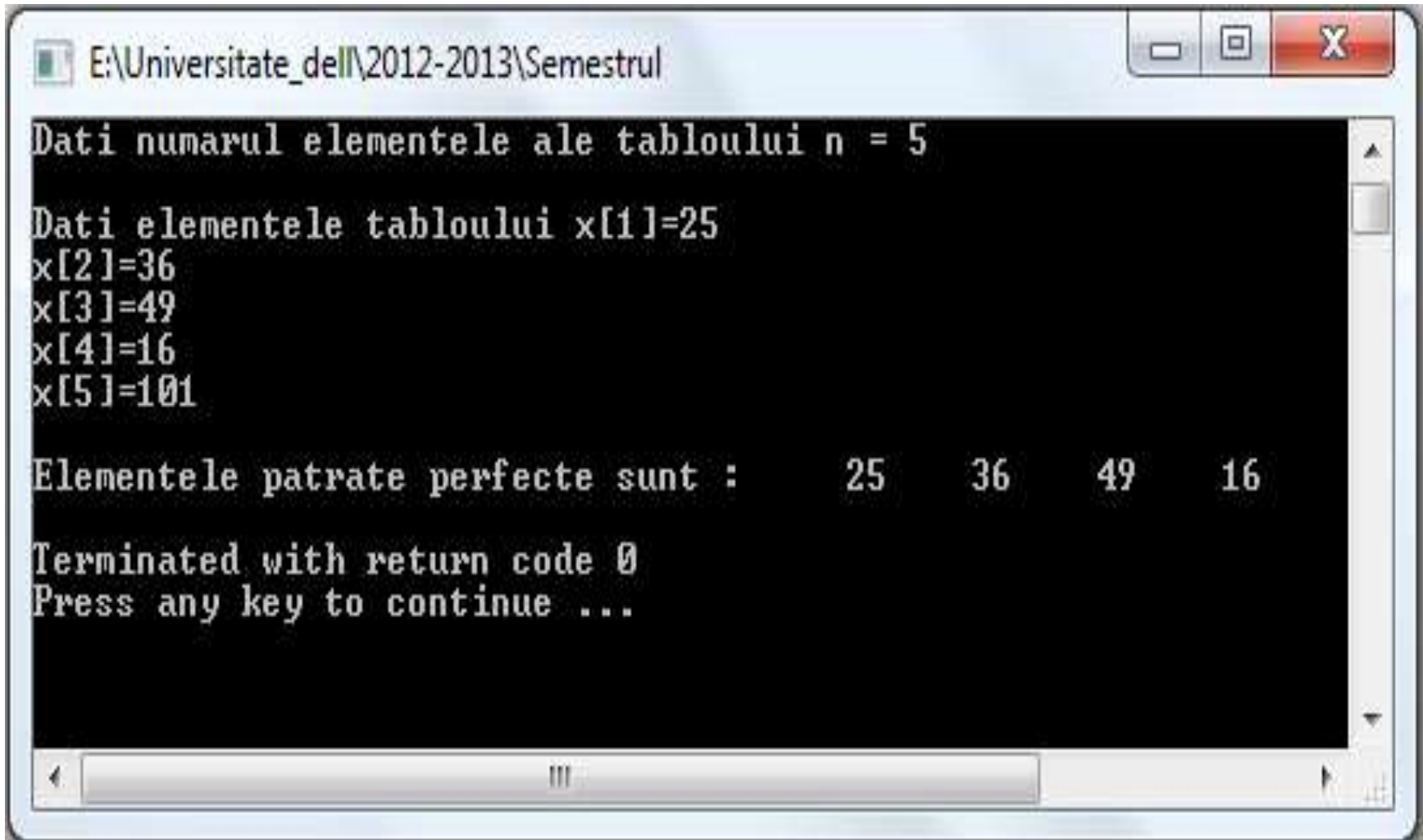


Verificarea
proprietatii de
patrat perfect

Probleme rezolvate cu vectori

```
cout<<"\nElementele patrate perfecte sunt : ";  
for(i = 0; i < j; i++) // j numarul de elemente din noul  
vector  
{  
    cout.width(6); // afișează pe 6 caractere  
    cout<<y[i];  
}  
}
```

Executia programului pe o serie de date de test:



```
E:\Universitate_de\2012-2013\Semestrul
Dati numarul elementele ale tabloului n = 5
Dati elementele tabloului x[1]=25
x[2]=36
x[3]=49
x[4]=16
x[5]=101

Elementele patrate perfecte sunt :      25      36      49      16

Terminated with return code 0
Press any key to continue ...
```

Probleme rezolvate cu vectori

Problema 2:

Enunț:

Se citește un tablou unidimensional cu **n** (**$1 \leq n \leq 100$**) componente numere naturale.

Se cere să se calculeze și să se afișeze suma elementelor din vector care sunt divizibile cu o valoare naturală dată **a**.

Probleme rezolvate cu vectori

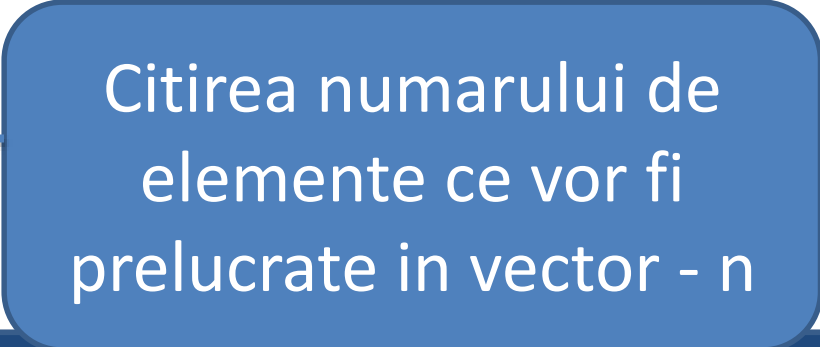
```
#include<iostream.h>
```

```
int main(void)  
{
```

```
    unsigned int x[100], a, n, i, s;
```

```
    cout<<"\nDati numarul de elemente ale  
    tabloului n = ";
```

```
    cin>>n;
```

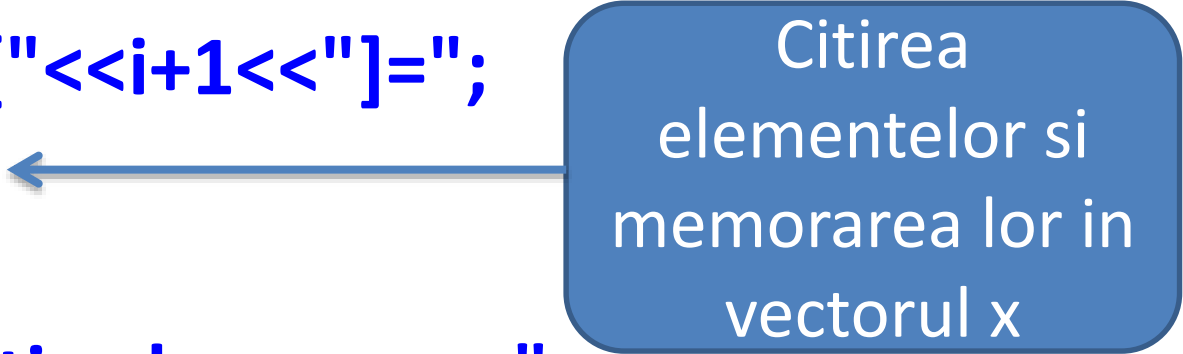


Citirea numarului de
elemente ce vor fi
prelucrate in vector - n

Probleme rezolvate cu vectori

```
cout<<"Dati elementele tabloului \n";  
for(i = 0; i < n; i++)  
{  
    cout<<"x["<<i+1<<"]="";  
    cin>>x[i];  
}  
cout<<"\nDati valoarea a = ";  
cin>>a;
```

Citirea
elementelor si
memorarea lor in
vectorul x



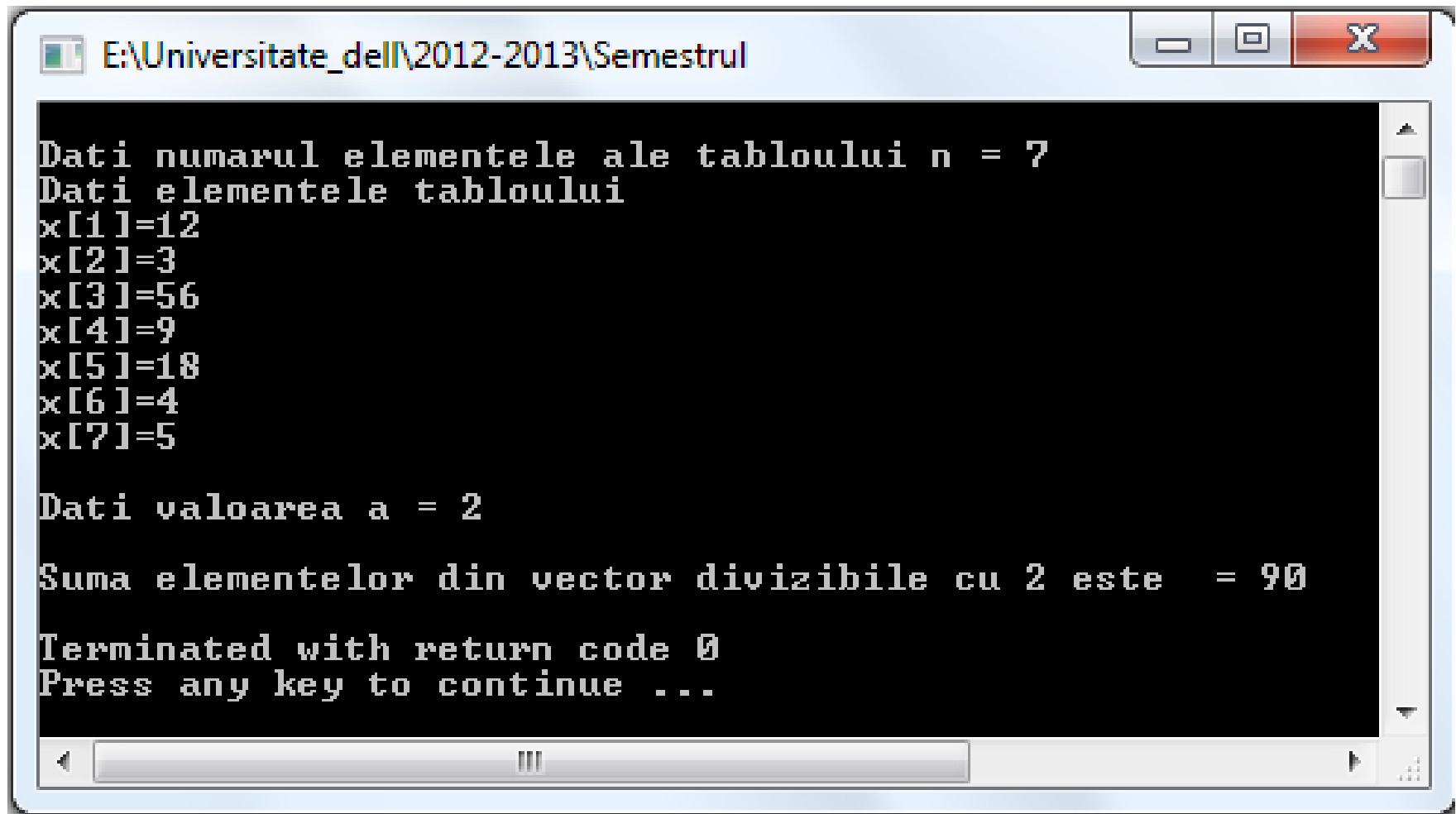
Citirea valorii a



Probleme rezolvate cu vectori

```
s = 0;    // suma elementelor care indeplinesc cerinta problemei
for(i = 0; i < n; i++)
    if(x[i] % a == 0)
        s = s + x[i];
cout<<"\nSuma elementelor din vector
divizibile cu "<<a<<" este = "<<s;
}
```

Executia programului pe o serie de date de test:



```
E:\Universitate_dell\2012-2013\Semestrul

Dati numarul elementele ale tabloului n = 7
Dati elementele tabloului
x[1]=12
x[2]=3
x[3]=56
x[4]=9
x[5]=18
x[6]=4
x[7]=5

Dati valoarea a = 2

Suma elementelor din vector divizibile cu 2 este = 90

Terminated with return code 0
Press any key to continue ...
```

Probleme rezolvate cu vectori

Problema 3:

Enunț:

Să se scrie un program C++ care să realizeze
inversarea unui vector:

- a) în același vector și fără a utiliza un vector suplimentar
- b) într-un alt vector

Probleme rezolvate cu vectori

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

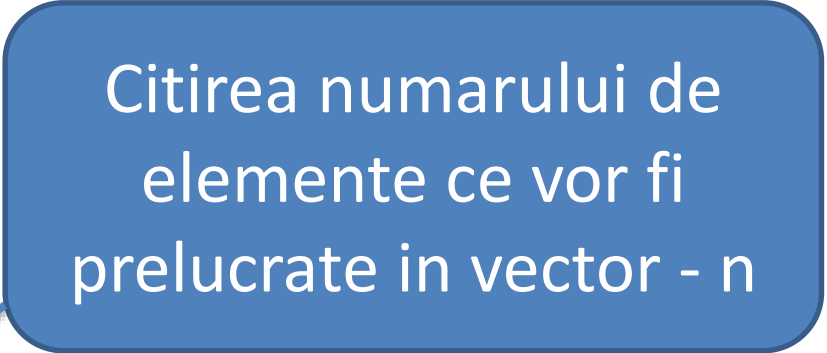
```
int main(void)  
{
```

```
    unsigned int x[100], y[100], n, i, j, aux;
```

```
    cout<<"Dati numarul de elemente ale tabloului n  
=";
```

```
    cin>>n;
```


Citirea numarului de
elemente ce vor fi
prelucrate in vector - n



Probleme rezolvate cu vectori

```
cout<<"Dati elementele tabloului\n";  
for(i = 0; i < n; i++)  
{  
    cout<<"x["<<i+1<<"]="";  
    cin>>x[i];  
    y[i]=x[i]; // pastram o copie a vectorului initial  
}
```

Citirea
elementelor si
memorarea
lor in vectorul
x



Probleme rezolvate cu vectori

```
cout<<"\n Punctul a ) \n";  
cout<<"\n Elementele vectorului inainte de  
inversare sunt : \n";  
for(i = 0; i < n; i++)  
{  
    cout.width(6);  
    cout<<x[i];  
}
```


Probleme rezolvate cu vectori

```
for(i = 0; i < n/2; i++)  
{  
    aux = x[i];  
    x[i] = x[n-i-1];  
    x[n-i-1]=aux;  
}
```

Probleme rezolvate cu vectori

```
cout<<"\nElementele vectorului dupa inversare  
sunt : ";  
for(i = 0; i < n; i++)  
{  
    cout.width(6);  
    cout<<x[i];  
}
```

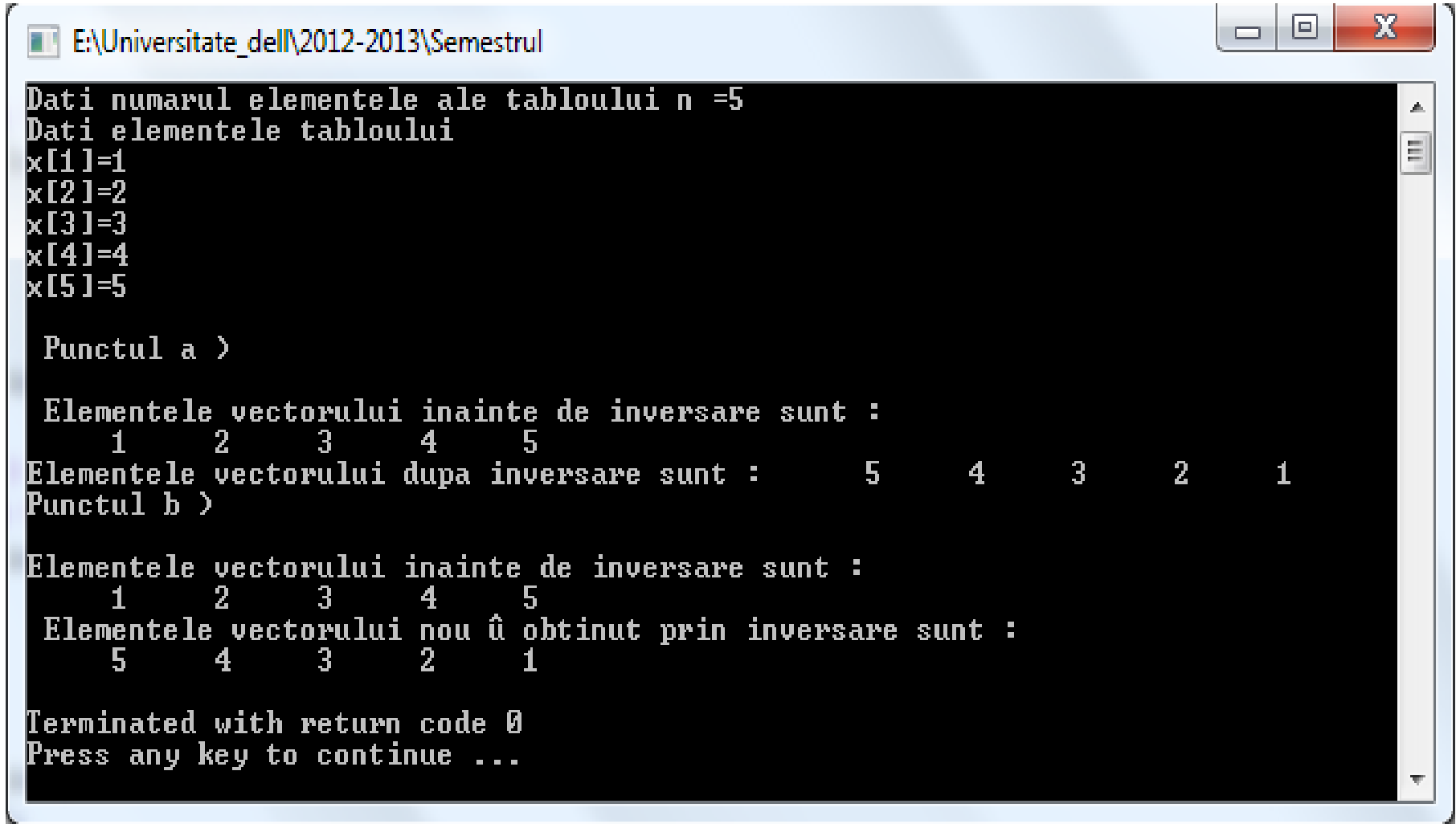
Probleme rezolvate cu vectori

```
for(i = 0; i < n; i++)  
    x[i] = y[i];                                //revenim la vectorul neinversat  
cout<<"\nPunctul b ) \n";  
cout<<"\nElementele vectorului inainte de inversare  
sunt : \n";  
for(i = 0; i < n; i++)  
{  
    cout.width(6);  
    cout<<x[i];  
}
```

Probleme rezolvate cu vectori

```
for(i = 0; i < n; i++)  
    y[i] = x[n-i-1];  
cout<<"\n Elementele vectorului nou – obtinut  
prin inversare sunt : \n";  
for(i = 0; i < n; i++)  
{  
    cout.width(6);  
    cout<<y[i];  
}  
}
```

Executia programului pe o serie de date de test:



```
E:\Universitate_dell\2012-2013\Semestrul
Dati numarul elementele ale tabloului n =5
Dati elementele tabloului
x[1]=1
x[2]=2
x[3]=3
x[4]=4
x[5]=5

Punctul a >

Elementele vectorului inainte de inversare sunt :
    1    2    3    4    5
Elementele vectorului dupa inversare sunt :      5    4    3    2    1
Punctul b >

Elementele vectorului inainte de inversare sunt :
    1    2    3    4    5
Elementele vectorului nou ũ obtinut prin inversare sunt :
    5    4    3    2    1

Terminated with return code 0
Press any key to continue ...
```

Capitolul 7. Tablouri

7.5. Probleme cu vectori

7.6. Probleme cu matrici

Probleme rezolvate cu matrici

Problema 1:

Enunț:

Se consideră o matrice $A_{n \times n}$. Să se calculeze:

1. suma elementelor de pe **diagonala principală**
2. produsul elementelor de pe **diagonala secundară**
3. minimele din elementele aflate deasupra, respectiv sub **diagonala principală**
4. maximele din elementele aflate deasupra, respectiv sub **diagonala principală**

Probleme rezolvate cu matrici

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

```
int main(void)
```

```
{
```

```
int n, m, i, j, a[30][30], min1, min2, max1, max2, suma,  
produs;
```

```
cout<<"Dati dimensiunile matricei \n";
```

```
cout<<"Dati numarul de linii si de coloane n = ";
```

```
cin>>n;
```

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

```
for(j=0; j<n; j++) {
```

```
cout<<"a["<<i+1<<"]["<<j+1<<"]="<<";
```

```
cin>>a[i][j];
```

Citirea numarului de
linii ale matricei - n

Citirea elementelor
si memorarea lor in
matricea a

Probleme rezolvate cu matrici

```
cout<<"Elementele matricei A sunt: \n";  
for(i=0; i<n; i++)  
{  
    for(j=0; j<n; j++)  
    {  
        cout.width(4);  
        cout<<a[i][j];  
    }  
    cout<<"\n";  
}
```

Afisarea
elementelor din
matricea a



Probleme rezolvate cu matrici

// == suma elementelor de pe diagonala principala ==

```
suma=0;
```

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

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

```
        if( i == j ) suma = suma + a[i][j];
```

```
cout<<"\nSuma elementelor de pe diagonala  
principala este "<<suma;
```

Probleme rezolvate cu matrici

// == produsul elementelor de pe diagonala secundara ==

```
produs=1;
for(i=0; i<n; i++)
    for(j=0; j<n; j++)
        if( i + j == n-1 ) produs = produs * a[i][j];

cout<<"\nProdusul elementelor de pe diagonala
secundara este "<<produs;
```

Probleme rezolvate cu matrici

// minimul elementelor de deasupra diagonalei principale

min1=10000;

for(i=0; i<n; i++)

for(j=0; j<n; j++)

if(i < j)

if(a[i][j] < min1) min1 = a[i][j];

cout<<"\nMinimul de deasupra diag.
principale este "<<min1;

Probleme rezolvate cu matrici

// minimul elementelor de sub diagonala principala

min2=10000;

for(i=0; i<n; i++)

for(j=0; j<n; j++)

if(i > j)

if(a[i][j] < min2) min2 = a[i][j];

**cout<<"\nMinimul de sub diag. principala este
"<<min2;**

Probleme rezolvate cu matrici

```
// maximul elementelor de deasupra diagonalei principale  
max1=-10000;  
for(i=0; i<n; i++)  
    for(j=0; j<n; j++)  
        if( i < j )  
            if( a[i][j] > max1 ) max1 = a[i][j];  
cout<<"\nMaximul de deasupra diag. principale  
este "<<max1;
```

Probleme rezolvate cu matrici

// maximul elementelor de sub diagonala principala

```
max2=-10000;
```

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

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

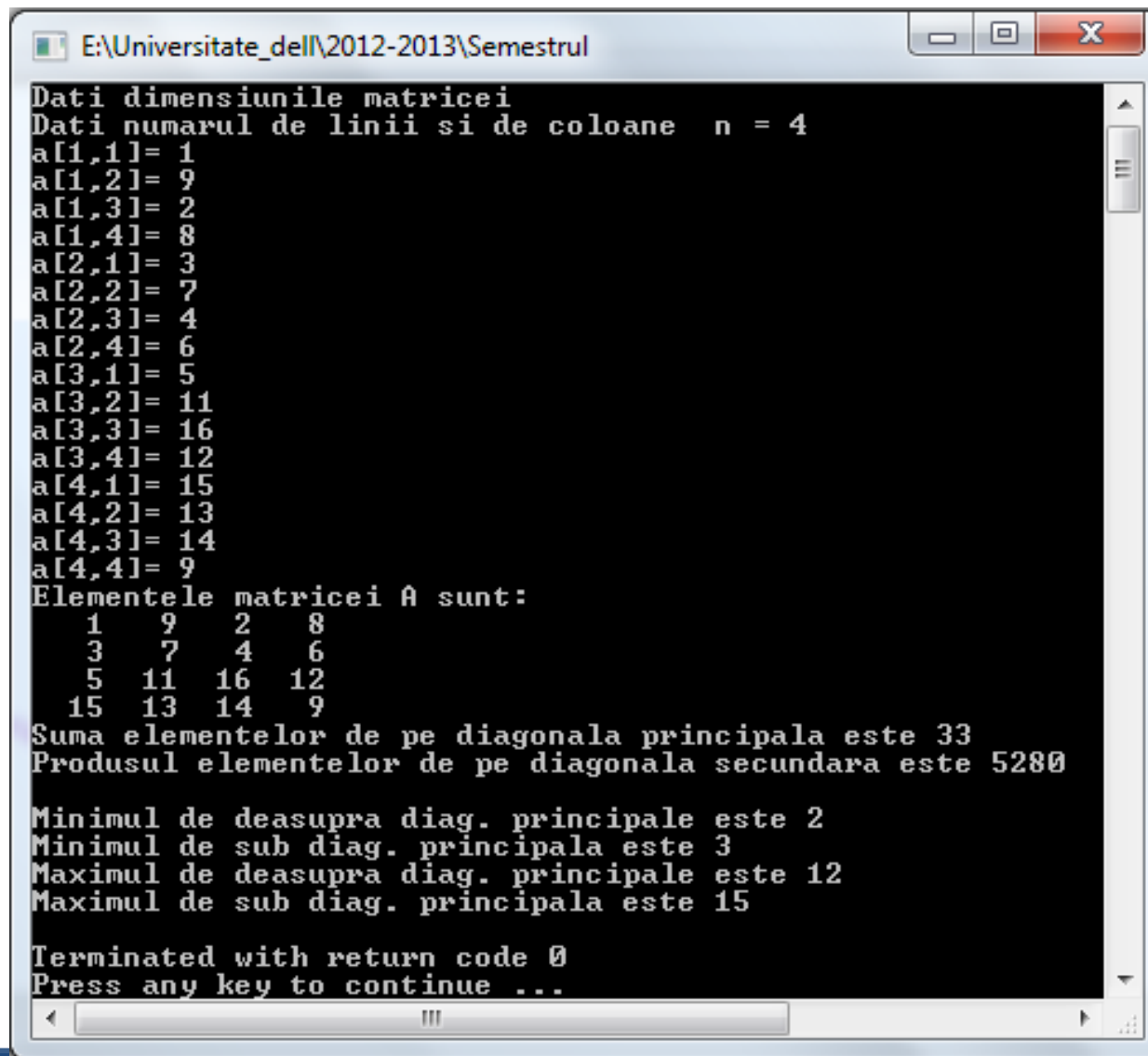
```
        if( i > j )
```

```
            if( a[i][j] > max2 ) max2 = a[i][j];
```

```
cout<<"\nMaximul de sub diag. principala este  
"<<max2;
```

```
}
```

Executia programului pe o serie de date de test:



```
E:\Universitate_dell\2012-2013\Semestrul
Dati dimensiunile matricei
Dati numarul de linii si de coloane  n = 4
a[1,1]= 1
a[1,2]= 9
a[1,3]= 2
a[1,4]= 8
a[2,1]= 3
a[2,2]= 7
a[2,3]= 4
a[2,4]= 6
a[3,1]= 5
a[3,2]= 11
a[3,3]= 16
a[3,4]= 12
a[4,1]= 15
a[4,2]= 13
a[4,3]= 14
a[4,4]= 9
Elementele matricei A sunt:
  1   9   2   8
  3   7   4   6
  5  11  16  12
 15  13  14   9
Suma elementelor de pe diagonala principala este 33
Produsul elementelor de pe diagonala secundara este 5280

Minimul de deasupra diag. principale este 2
Minimul de sub diag. principala este 3
Maximul de deasupra diag. principale este 12
Maximul de sub diag. principala este 15

Terminated with return code 0
Press any key to continue ...
```


Probleme rezolvate cu matrici

Problema 2:

Enunț:

Se consideră o matrice $A_{n \times m}$ ($1 \leq n, m \leq 30$) având componente numere întregi.

Să se determine **maximul fiecărei coloane** și **minimul fiecărei linii**.

Probleme rezolvate cu matrici

```
#include <iostream.h>
int main(void)
{
    int n, m, i, j, a[30][30], min, max;
    cout<<"Dati dimensiunile matricei \n";
    cout<<"Dati numarul de linii n = ";
    cin>>n;
    cout<<"Dati numarul de linii m = ";
    cin>>m;
    for(i=0; i<n; i++)
        for(j=0; j<m; j++) {
            cout<<"a["<<i+1<<"]["<<j+1<<"]=" ";
            cin>>a[i][j];
        }
}
```

Citirea numarului
de linii ale
matricei – n si a
numarului de
coloane - m

Citirea elementelor
si memorarea lor in
matricea a

Probleme rezolvate cu matrici

```
cout<<"Elementele matricei A sunt: \n";  
for(i=0; i<n; i++)  
{  
    for(j=0; j<m; j++)  
    {  
        cout.width(4);  
        cout<<a[i][j];  
    }  
    cout<<"\n";  
}
```



Afisarea
elementelor din
matricea a

Probleme rezolvate cu matrici

// calculam minimele de pe fiecare linie

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

```
{
```

```
    min = a[i][0];
```

```
    for(j=1; j<m; j++)
```

```
        if( a[i][j] < min )
```

```
            min = a[i][j];
```

```
    cout<<"Minimul de pe linia "<<i<<" este  
    "<<min<<"\n";
```

```
}
```

Probleme rezolvate cu matrici

// calculam maximele de pe fiecare coloana

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

```
{
```

```
    max = a[0][j];
```

```
    for(i=1; i<n; i++)
```

```
        if( a[i][j] > max )
```

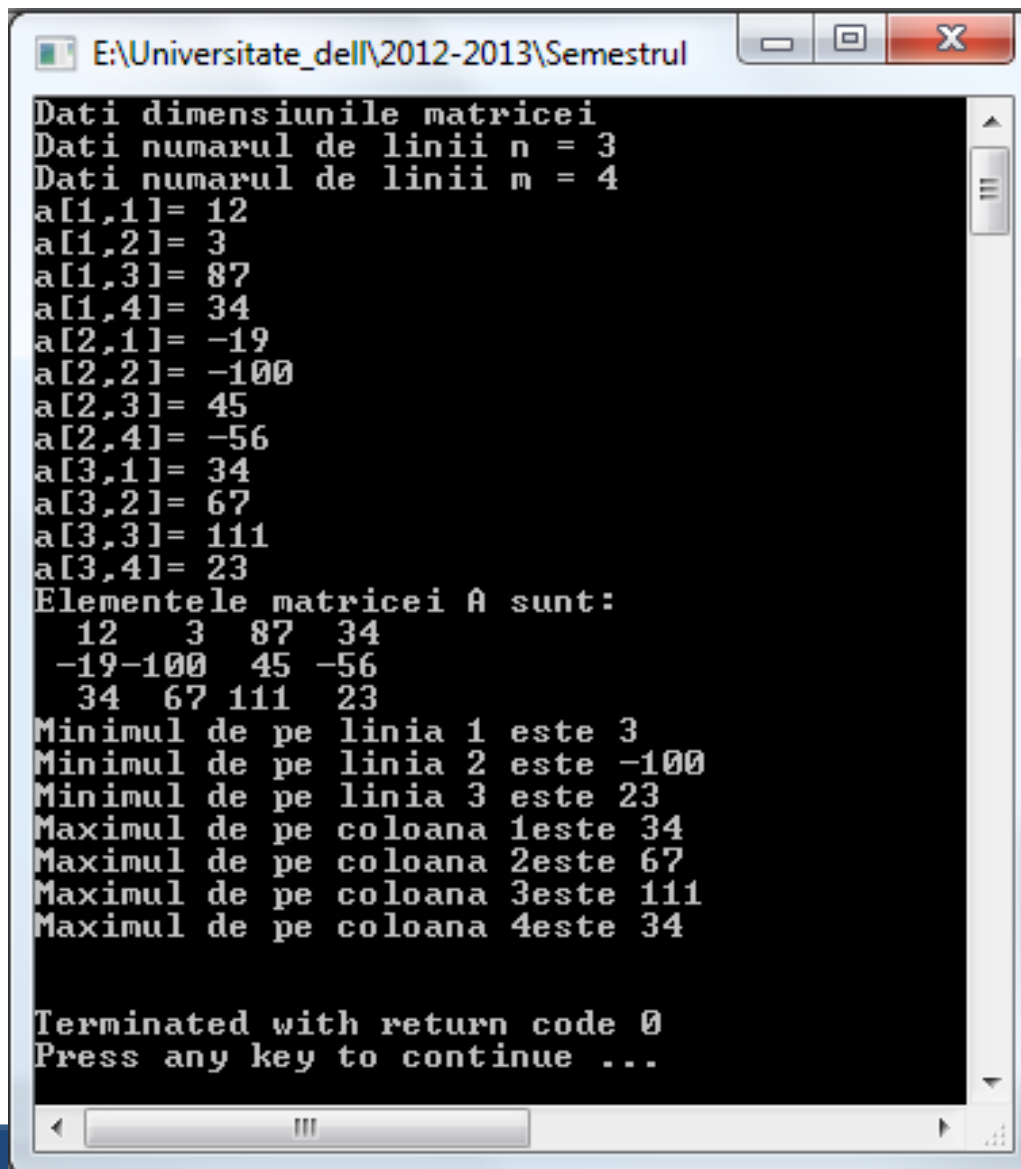
```
            max = a[i][j];
```

```
    cout<<"Maximul de pe coloana "<<j<<"este  
"<<max<<"\n";
```

```
}
```

```
}
```

Executia programului pe o serie de date de test:



```
E:\Universitate_dell\2012-2013\Semestrul
Dati dimensiunile matricei
Dati numarul de linii n = 3
Dati numarul de linii m = 4
a[1,1]= 12
a[1,2]= 3
a[1,3]= 87
a[1,4]= 34
a[2,1]= -19
a[2,2]= -100
a[2,3]= 45
a[2,4]= -56
a[3,1]= 34
a[3,2]= 67
a[3,3]= 111
a[3,4]= 23
Elementele matricei A sunt:
  12  3  87  34
 -19 -100 45 -56
  34  67 111  23
Minimul de pe linia 1 este 3
Minimul de pe linia 2 este -100
Minimul de pe linia 3 este 23
Maximul de pe coloana 1este 34
Maximul de pe coloana 2este 67
Maximul de pe coloana 3este 111
Maximul de pe coloana 4este 34

Terminated with return code 0
Press any key to continue ...
```

Probleme rezolvate cu matrici

Problema 3:

Enunț:


Se consideră o matrice $A_{n \times m}$ ($1 \leq n, m \leq 30$) având componente numere întregi.

Să se determine **suma elementelor de pe marginea (rama) matricei**.

Probleme rezolvate cu matrici

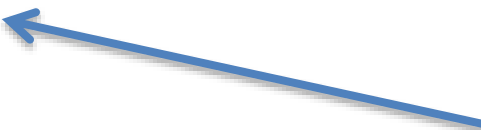
```
#include <iostream.h>
int main(void)
{
```

Citirea numarului de linii ale
matricei – n si a numarului de
coloane - m



```
    int n, m, i, j, a[30][30], suma;
    cout<<"Dati dimensiunile matricei \n";
    cout<<"Dati numarul de linii n = ";
    cout<<"Dati numarul de coloane m = ";
    for(i=0; i<n; i++)
        for(j=0; j<m; j++) {
            cout<<"a["<<i+1<<"]["<<j+1<<"]=" ";
            cin>>a[i][j];
        }
```

Citirea elementelor
si memorarea lor in
matricea a



Probleme rezolvate cu matrici

```
cout<<"Elementele matricei A sunt: \n";  
for(i=0; i<n; i++)  
{  
    for(j=0; j<m; j++)  
    {  
        cout.width(4);  
        cout<<a[i][j];  
    }  
    cout<<"\n";  
}
```



Afisarea
elementelor din
matricea a

Probleme rezolvate cu matrici

// calculam suma elementelor de pe linia 0

suma = 0;

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

suma = suma + a[0][j];

// calculam suma elementelor de pe linia n-1

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

suma = suma + a[n-1][j];

Probleme rezolvate cu matrici

// calculam suma elementelor de pe **coloana 0**

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

```
    suma = suma + a[i][0];
```

// calculam suma elementelor de pe **coloana m-1**

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

```
    suma = suma + a[i][m-1];
```

Probleme rezolvate cu matrici

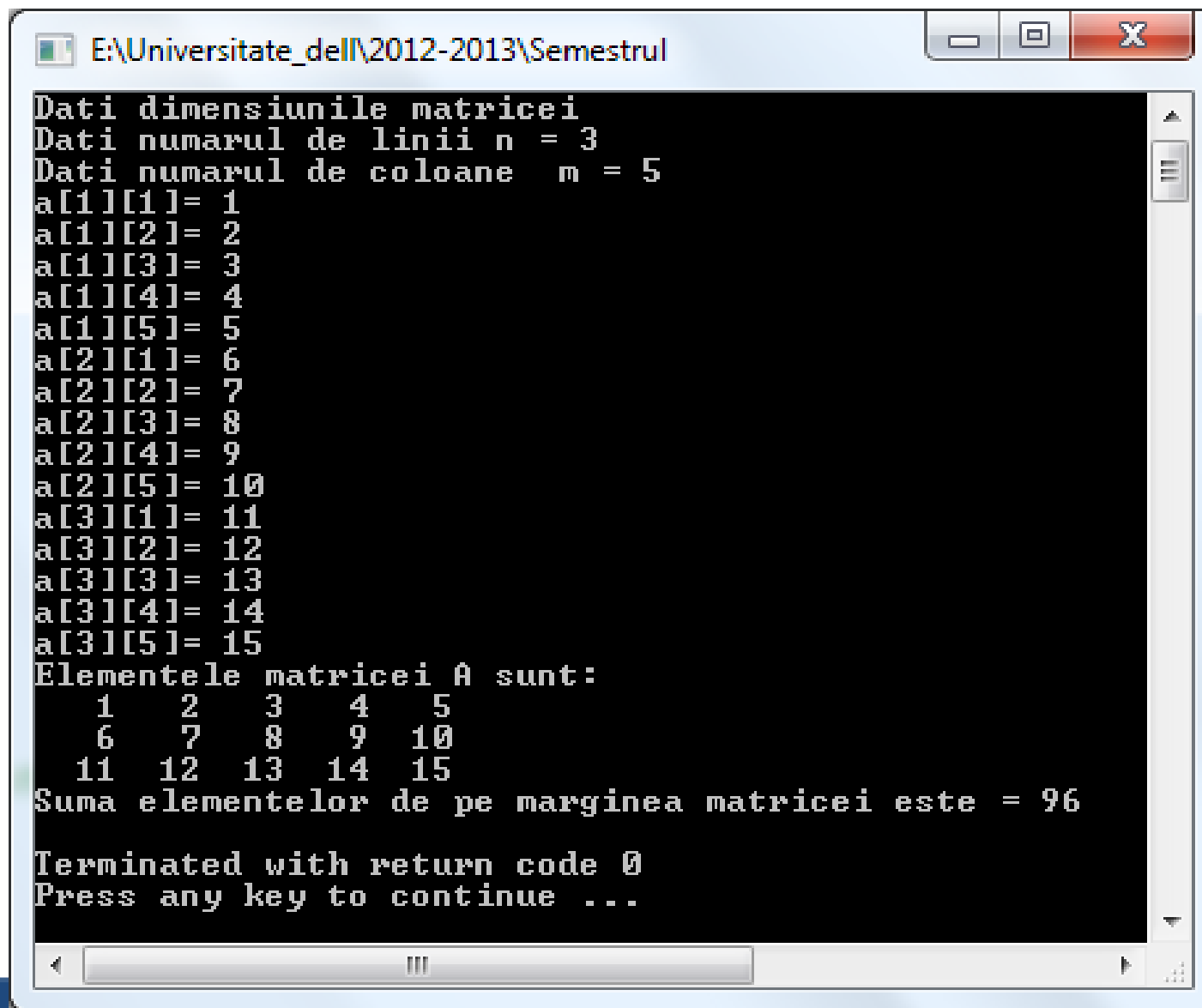
// scadem cele **4 colturi ale matricei**, deoarece le-am adunat de 2 ori pe fiecare

```
suma = suma - a[0][0] - a[0][m-1] - a[n-1][0] -  
a[n-1][m-1];
```

```
cout<<"\nSuma elementelor de pe marginea  
matricei este = "<<suma;
```

```
}
```

Executia programului pe o serie de date de test:



```
E:\Universitate_dell\2012-2013\Semestrul
Dati dimensiunile matricei
Dati numarul de linii n = 3
Dati numarul de coloane m = 5
a[1][1]= 1
a[1][2]= 2
a[1][3]= 3
a[1][4]= 4
a[1][5]= 5
a[2][1]= 6
a[2][2]= 7
a[2][3]= 8
a[2][4]= 9
a[2][5]= 10
a[3][1]= 11
a[3][2]= 12
a[3][3]= 13
a[3][4]= 14
a[3][5]= 15
Elementele matricei A sunt:
  1   2   3   4   5
  6   7   8   9  10
 11  12  13  14  15
Suma elementelor de pe marginea matricei este = 96
Terminated with return code 0
Press any key to continue ...
```

Probleme propuse spre rezolvate – tablouri unidimensionale si tablouri bidimensionale

1. Dintr-un șir de numere întregi, să se afișeze suma elementelor de pe pozițiile pare și suma celor de pe poziții impare.

Exemplu:

Date de intrare:

n=6 si numerele 3 1 5 4 6 7

Date de ieșire:

s_imp = 14 (am însumat elem. 3+5+6)

s_par = 12 (am însumat elem. 1+4+7)

Probleme propuse spre rezolvate – tablouri unidimensionale si tablouri bidimensionale

2. Se introduc în calculator și se memorează vârstele a n bărbați. Afișați câți au vârsta mai mare decât 50 și mai mică decât 70 de ani.

Exemplu:

Date de intrare:

$n=4$ si vârstele 56 45 76 65

Date de ieșire:

2 barbati

Probleme propuse spre rezolvate – tablouri unidimensionale si tablouri bidimensionale

3. Fiind date n numere naturale, câte sunt prime cu 7? Dar cu 10?

Exemplu:

Date de intrare:

$n=6$ si numerele 5 7 14 20 12 15

Date de ieşire:

prime cu 7: 4 numere (5,20,12,15)

prime cu 10: 1 numar (7)

Doua numere sunt prime între ele daca cmmdc al lor este 1.

Probleme propuse spre rezolvate – tablouri unidimensionale si tablouri bidimensionale

4. Se dă un tablou bidimensional cu n linii și m coloane, $1 \leq n, m \leq 50$, cu componente întregi și un număr întreg k . Se cere să se afișeze tabloul cu componentele mărite cu k .

Exemplu:

Date de intrare:

$n = 2, m = 3, k = 5$ si elementele:

1 2 3

6 7 8

Date de ieșire:

6 7 8

11 12 13

Probleme propuse spre rezolvate – tablouri unidimensionale si tablouri bidimensionale

5. Fiind dat un tablou bidimensional cu n linii și n coloane, afișați cea mai mare componentă și pozițiile pe care le ocupă.

Exemplu:

Date de intrare:

$n = 2$ si matricea:

4 7

5 7

Date de ieșire:

max = 7 se afla pe pozitiile 1,2 si 2,2

Probleme propuse spre rezolvate – tablouri unidimensionale si tablouri bidimensionale

6. Să se afișeze suma elementelor de pe coloana k a unei matrici cu n linii și m coloane, $k \leq m$.

Exemplu:

Date de intrare:

$n = 2, m = 4, k = 3$ si matricea

4 3 6 8

7 4 9 0

Date de ieșire:

$s = 15$

Pentru alte informații teoretice și aplicative legate de acest capitol se recomandă următoarele referințe bibliografice:

1. Adrian Runceanu, Mihaela Runceanu, ***Noțiuni de programare în limbajul C++***, Editura Academica Brâncuși, Târgu-Jiu, 2012 (www.utgjiu.ro/editura)
2. Adrian Runceanu, **Programarea și utilizarea calculatoarelor**, Editura Academica Brâncuși, Târgu-Jiu, 2003 (www.utgjiu.ro/editura)
3. Octavian Dogaru, **C++ - teorie și practică**, volumul I, Editura Mirton, Timișoara, 2004 (www.utgjiu.ro/editura)

Întrebări?