

100. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 100$ ) și apoi elementele unui tablou bidimensional cu  $n$  linii și  $n$  coloane, care memorează numere naturale cu cel mult 9 cifre fiecare; programul afișează pe ecran acele valori din tablou care sunt strict mai mici decât toate elementele cu care se învecinează direct (aflate pe aceeași linie dar pe o coloană alăturată sau pe aceeași coloană dar pe o linie alăturată), ca în exemplu. Numerele afișate vor fi separate prin câte un spațiu. **Exemplu:** pentru  $n=4$  și tabloul alăturat se afișează numerele: 2 0 (2 se învecinează direct cu 4, 3, 6 și 9, și este mai mic decât acestea, iar 0 se învecinează direct cu 6, 9 și 1 și este mai mic decât acestea).

```
5 4 7 9
6 2 3 4
0 9 8 5
1 3 8 6
```

99. Scrieți un program C/C++ care citește de la tastatură un număr natural nenul  $n$  ( $n \leq 24$ ) și construiește în memorie un tablou bidimensional cu  $n$  linii și  $n$  coloane care să conțină primele  $n$  numere naturale nenule. Prima linie a tabloului va conține, în această ordine, valorile 1, 2, ...,  $n$ ; a doua linie va conține, în ordine, valorile 2, 2, 3, ...,  $n$ ; a treia linie va conține, în ordine, valorile 3, 3, 3, 4, ...,  $n$ , iar ultima linie va conține valorile  $n$ ,  $n$ , ...,  $n$ . Programul afișează pe ecran matricea construită, câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind despărțite prin câte un spațiu.

**Exemplu:** pentru  $n=5$  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 5 5 5 5
```

98. Scrieți un program C/C++ care citește de la tastatură un număr natural nenul  $n$  ( $n \leq 24$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane care să conțină primele  $n^2$  numere naturale pare. Prima linie a matricei va conține, în ordine crescătoare, valorile 0, 2, ...,  $2n-2$ ; a doua linie va conține, în ordine, valorile  $2n$ ,  $2n+2$ , ...,  $4n-2$ ; a treia linie va conține, în ordine, valorile  $4n$ ,  $4n+2$ , ...,  $6n-2$ , iar ultima linie va conține, în ordine, valorile  $2n^2-2n$ ,  $2n^2-2n+2$ , ...,  $2n^2-2$ . Programul afișează pe ecran matricea construită, câte o linie a matricei pe câte o linie a ecranului, elementele de pe aceeași linie fiind despărțite prin câte un spațiu.

**Exemplu:** pentru  $n=3$  se va afișa matricea alăturată.

```
0 2 4
6 8 10
12 14 16
```

97. Se consideră un tablou bidimensional cu  $n$  linii și  $m$  coloane ( $1 \leq n \leq 24$ ,  $1 \leq m \leq 24$ ) ce memorează numere întregi cu cel mult două cifre fiecare. Scrieți un program în limbajul C/C++ care citește de la tastatură valorile  $n$ ,  $m$  și elementele tabloului, și care inversează ordinea elementelor în cadrul fiecărei coloane, ca în exemplu. Programul va afișa pe ecran, pe  $n$  linii, matricea obținută după inversare, elementele fiecărei linii fiind separate prin câte un spațiu.

**Exemplu:** pentru  $n=4$ ,  $m=3$  și matricea:

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

Pe ecran se va afișa:

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

92. Scrieți un program C/C++ care citește de la tastatură numerele întregi  $m$  și  $n$  ( $1 \leq m \leq 24$ ,  $1 \leq n \leq 24$ ) și elementele unui tablou bidimensional cu  $m$  linii și  $n$  coloane, numere întregi distincte de cel mult 4 cifre fiecare, și elimină din tablou, la nivelul memoriei, linia și coloana corespunzătoare elementului de valoare minimă. Programul va afișa tabloul obținut pe ecran pe  $m-1$  linii, elementele fiecărei linii fiind separate prin câte un spațiu.

**Exemplu:** pentru  $m=3$  și  $n=4$  și tabloul de mai jos

```
2 7 1 4
14 6 12 3
9 22 8 5
```

Pe ecran se va afișa:

```
14 6 3
9 22 5
```

90. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 20$ ), elementele unei matrice cu  $n$  linii și  $n$  coloane, numere întregi din intervalul  $[-100, 100]$  și afișează pe ecran diferența  $m1 - m2$ , unde  $m1$  este media aritmetică a elementelor strict pozitive ale matricei, situate deasupra diagonalei principale, iar  $m2$  este media aritmetică a elementelor strict pozitive ale matricei, situate sub diagonala principală, ca în exemplu. Cele două medii se consideră egale cu 0 dacă nu există valori strict pozitive în zonele corespunzătoare.

**Exemplu:** pentru  $n=4$  și matricea alăturată se afișează valoarea 0.25

( $m1=2.75$ , calculată din elementele aflate deasupra diagonalei principale, marcate cu chenar, și  $m2=2.5$ , calculată din elementele subliniate). (10p.)

```
-1 2 -4 5
0 6 3 1
2 4 2 0
3 -5 1 -3
```

89. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 6$ ) apoi construiește în memorie o matrice cu  $n$  linii și  $n$  coloane, astfel încât parcurgând liniile matricei de sus în jos și de la stânga la dreapta se obțin, în prima linie primele  $n$  numere ale șirului Fibonacci în ordine **crescătoare**, în linia a doua următoarele  $n$  numere ale șirului Fibonacci în ordine **descrescătoare**, în linia a treia următoarele  $n$  numere ale acestui șir în ordine **crescătoare**, și așa mai departe, ca în exemplu. Elementele șirului Fibonacci se obțin astfel: primul element este 0, al doilea este 1, iar elementele următoare se obțin însumând cele două elemente care preced elementul curent. Astfel, primele 16 elemente ale acestui șir sunt: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610.

Programul afișează pe ecran matricea obținută, 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 obține matricea alăturată. (10p.)

```
0 1 1 2
13 8 5 3
21 34 55 89
```

88. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 20$ ), elementele unei matrice cu  $n$  linii și  $n$  coloane, numere întregi din intervalul  $[-100, 100]$  și afișează pe ecran media aritmetică a elementelor strict pozitive ale matricei, care sunt situate deasupra diagonalei principale, ca în exemplu. Dacă nu există elemente strict pozitive situate deasupra diagonalei principale, programul va afișa mesajul „NU EXISTA”.

**Exemplu:** pentru  $n=4$  și matricea alăturată se afișează valoarea 2.75 (sunt luate în considerare doar elementele marcate).

```
-1 2 -4 5
0 6 3 1
2 4 2 0
3 -5 1 -3
```

87. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 20$ ), elementele unei matrice cu  $n$  linii și  $n$  coloane, numere întregi din intervalul  $[-100, 100]$  și afișează pe ecran media aritmetică a elementelor strict pozitive ale matricei, care sunt situate sub diagonala principală, ca în exemplu. Dacă nu există elemente strict pozitive situate sub diagonala principală, programul va afișa mesajul „NU EXISTA”.

**Exemplu:** pentru  $n=4$  și matricea alăturată se afișează valoarea 2.5 (sunt luate în considerare doar elementele marcate). (10p.)

```
-1 2 4 5
0 6 3 1
2 4 2 0
3 -5 1 -3
```

86. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 10$ ) apoi construiește în memorie o matrice cu  $2*n$  linii și  $2*n$  coloane, numerotate de la 1 la  $2*n$ , astfel încât parcurgând doar liniile impare ale matricei de sus în jos și fiecare linie impară de la stânga la dreapta se obțin în ordine strict crescătoare toate numerele impare cuprinse în intervalul  $[1, 4*n_2]$ , iar parcurgând doar liniile pare ale matricei de sus în jos și fiecare linie pară de la dreapta la stânga se obțin în ordine strict crescătoare toate numerele pare cuprinse în intervalul  $[1, 4*n_2]$ , ca în exemplu.

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

**Exemplu:** pentru  $n=2$  se obține matricea alăturată. (10p.)

```
1 3 5 7
8 6 4 2
9 11 13 15
16 14 12 10
```

82. Scrieți un program C/C++ care citește de la tastatură trei valori naturale nenule  $k$ ,  $n$ ,  $m$  ( $n \leq 10$ ,  $m \leq 10$ ,  $k \leq 32000$ ) și apoi  $n*m$  numere întregi, fiecare având cel mult 4 cifre, reprezentând elementele unui tablou bidimensional cu liniile numerotate de la 1 la  $n$  și

coloanele numerotate de la 1 la  $m$ . Programul determină și afișează pe ecran produsul numerelor de ordine ale coloanelor care conțin cel puțin o dată valoarea  $k$ . Dacă nu există nicio coloană care să conțină cel puțin o dată valoarea  $k$ , programul va afișa pe ecran mesajul **NU EXISTA**.

**Exemplu:** pentru  $k=3$ ,  $n=5$ ,  $m=4$  și matricea alăturată se va afișa 6, deoarece coloanele 2 și 3 conțin numărul  $k=3$ . **(10p.)**

```
2 4 5 -8
5 3 7 9
6 -2 3 10
7 4 2 37
8 7 3 13
```

81. Scrieți un program C/C++ care citește de la tastatură trei valori naturale nenule  $k$ ,  $n$ ,  $m$  ( $n \leq 10$ ,  $m \leq 10$ ,  $k \leq 32000$ ) și apoi  $n \cdot m$  numere întregi, fiecare având cel mult 4 cifre, reprezentând elementele unui tablou bidimensional cu liniile numerotate de la 1 la  $n$  și coloanele numerotate de la 1 la  $m$ . Programul determină și afișează pe ecran suma numerelor de ordine ale coloanelor care conțin cel puțin o dată valoarea  $k$ . Dacă nu există nicio coloană care să conțină cel puțin o dată valoarea  $k$ , programul va afișa pe ecran mesajul **NU EXISTA**. **Exemplu:** pentru  $k=3$ ,  $n=5$ ,  $m=4$  și matricea alăturată se va afișa 5, deoarece coloanele 2 și 3 conțin numărul  $k=3$ . **(10p.)**

```
2 4 5 -8
1 3 7 9
4 -2 3 10
5 4 2 37
6 7 3 13
```

75. Într-o matrice  $a$ , cu  $n$  linii și  $n$  coloane, numerotate de la 1 la  $n$ , numim "semidiagonală de indice  $k$ " mulțimea formată din elementele  $a_{i,j}$  ale matricei pentru care relația:  $i+j=k+1$  este adevărată ( $1 \leq k < 2 \cdot n$ ,  $1 \leq i \leq n$ ,  $1 \leq j \leq n$ ,  $i \in \mathbb{N}$ ,  $j \in \mathbb{N}$ ,  $k \in \mathbb{N}$ ).

Scrieți programul C/C++ care citește de la tastatură o valoare naturală  $n$  ( $2 \leq n \leq 24$ ), construiește în memorie și afișează pe ecran o matrice  $a$  cu  $n$  linii și  $n$  coloane, numerotate de la 1 la  $n$ , în care elementele fiecărei semidiagonale de indice  $k$  sunt egale cu  $(k+1)^2$ , ca în exemplu. Matricea se va afișa pe ecran, câte o linie a matricei pe o linie a ecranului, elementele unei linii fiind separate între ele prin câte un spațiu.

**Exemplu:** pentru  $n = 4$  se va obține matricea alăturată. **(10p.)**

```
4 9 16 25
9 16 25 36
16 25 36 49
25 36 49 64
```

74. Se consideră șirul lui Fibonacci, definit astfel:  $f_0 = 0$ ,  $f_1 = 1$ ,  $f_n = f_{n-1} + f_{n-2}$ , dacă  $n > 1$ . Scrieți un program C/C++ care citește de la tastatură o valoare naturală  $n$  ( $2 \leq n \leq 24$ ), construiește în memorie și afișează pe ecran o matrice  $a$  cu  $n$  linii și  $n$  coloane ale cărei elemente sunt numere naturale, fiecare reprezentând ultima cifră a câte unui termen al șirului lui Fibonacci, începând de la termenul de indice 1 și până la termenul de indice  $n^2$ , în ordine, linie cu linie de sus în jos, și în cadrul fiecărei linii de la stânga la dreapta, ca în exemplu.

Matricea se va afișa pe ecran, câte o linie a matricei pe câte o linie a ecranului, elementele de pe aceeași linie fiind separate între ele prin câte un spațiu.

**Exemplu:** pentru  $n = 4$  se va obține matricea alăturată. (10p.)

```
1 1 2 3
5 8 3 1
4 5 9 4
3 7 0 7
```

72. Scrieți programul C/C++ care citește de la tastatură o valoare naturală  $n$  ( $2 \leq n \leq 24$ ), construiește în memorie și apoi afișează pe ecran o matrice  $a$ , cu  $n$  linii și  $n$  coloane, **simetrică** față de diagonală secundară. Elemente matricei situate deasupra diagonalei secundare, inclusiv diagonală secundară, sunt toate numerele naturale de la 1 la  $2n(n+1)$ . Elementele fiecărei linii, de la stânga la dreapta și ale fiecărei coloane, de sus în jos, situate deasupra diagonalei secundare, inclusiv cele de pe diagonală secundară, sunt în ordine strict crescătoare, ca în exemplu. Matricea se va afișa pe ecran, câte o linie a matricei pe o linie a ecranului, elementele fiecărei linii fiind separate între ele prin câte un spațiu.

**Exemplu:** pentru  $n = 4$ , atunci o soluție posibilă este matricea alăturată.

(10p.)

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

70. Scrieți programul C/C++ care citește de la tastatură două numere naturale  $m$  și  $n$  ( $1 \leq m \leq 24$ ,  $1 \leq n \leq 24$ ), un număr natural  $x$  ( $1 \leq x \leq m$ ) și apoi  $m \cdot n$  numere naturale de cel mult 5 cifre ce reprezintă elementele unui tablou bidimensional  $a$ , cu  $m$  linii, numerotate de la 1 la  $m$ , și  $n$  coloane, numerotate de la 1 la  $n$ . Programul va determina eliminarea liniei cu numărul de ordine  $x$  din matrice, modificarea corespunzătoare a numărului de linii din matrice și afișarea matricei obținute în următorul format: câte o linie a matricei pe câte o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu. (10p.)

**Exemplu:** pentru  $m=3$ ,  $n=4$ ,

$x=2$  și matricea alăturată

```
11 21 31 41 se va afișa matricea
51 61 71 81
91 11 21 31
11 21 31 41
91 11 21 31
```

69. Scrieți programul C/C++ care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 50$ ) și  $n \cdot n$  numere naturale de cel mult 5 cifre ce reprezintă elementele unui tablou bidimensional  $a$ , cu  $n$  linii și  $n$  coloane, și verifică dacă matricea este triunghiulară superior. Programul va afișa pe ecran mesajul corespunzător: „Este triunghiulară superior” respectiv „Nu este triunghiulară superior”. O matrice se numește triunghiulară superior dacă toate elementele aflate sub diagonală principală a ei sunt nule. (10p.)

**Exemplu:** pentru  $n=3$  și matricea alăturată se va afișa mesajul:

Este triunghiulară superior

```
1 2 3
0 5 6
0 0 9
```

68. Scrieți programul C/C++ care citește de la tastatură două numere naturale  $m$  și  $n$  ( $1 \leq m \leq 50$ ,  $1 \leq n \leq 50$ ) și  $m \cdot n$  numere întregi de cel mult 5 cifre ce reprezintă elementele unui tablou bidimensional, și afișează pe ecran ultima cifră a produsului elementelor pozitive aflate pe linii cu numere de ordine pare și coloane cu numere de ordine impare.

Numerotarea liniilor, respectiv a coloanelor, se va face începând cu valoarea 1. Dacă nu există elemente pozitive aflate pe linii cu numere de ordine pare și coloane cu numere de ordine impare, se va afișa mesajul **NU EXISTA.** (10p.)

**Exemplu:** pentru  $m=4$ ,  $n=4$  și matricea alăturată se va afișa 5 (care reprezintă ultima cifră a valorii  $355=5 \cdot 71$ ).

```
11 -21 31 41
5 -61 71 -81
91 11 21 31
-11 31 -41 0
```

65. Scrieți un program în limbajul C/C++ care citește de la tastatură două valori naturale  $n$  și  $m$

( $1 \leq n \leq 24$ ,  $1 \leq m \leq 24$ ) și construiește în memorie un tablou bidimensional cu  $n$  linii și  $m$  coloane format din toate numerele naturale de la 1 la  $n \cdot m$ , ca în exemplu. Programul va afișa pe ecran, pe  $n$  linii, tabloul obținut, elementele fiecărei linii fiind separate prin câte un spațiu.

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

```
20 19 18 17 16
11 12 13 14 15
10 9 8 7 6
1 2 3 4 5
```

63. Scrieți un program în limbajul C/C++ care citește de la tastatură două valori naturale  $n$  și  $m$  ( $1 \leq n \leq 24$ ,  $1 \leq m \leq 24$ ) și construiește în memorie un tablou bidimensional cu  $n$  linii și  $m$  coloane format din toate numerele naturale de la 1 la  $n \cdot m$ , ca în exemplu. Programul va afișa pe ecran, pe  $n$  linii, tabloul obținut, elementele fiecărei linii fiind separate prin câte un spațiu.

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

```
5 10 15 20
4 9 14 19
3 8 13 18
2 7 12 17
1 6 11 16
```

61. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $2 < n \leq 15$ ) și construiește în memorie o matrice pătrată cu  $n$  linii și  $n$  coloane în care ultima linie conține, în ordine, numerele  $1, 2, 3, \dots, n$ , elementele situate deasupra diagonalei principale sunt nule și oricare alt element este obținut prin însumarea elementelor vecine cu el, aflate pe linia imediat următoare, pe aceeași coloană cu el sau pe una din coloanele alăturate. Programul va afișa pe ecran matricea obținută pe  $n$  linii, elementele fiecărei linii fiind separate prin câte un spațiu.

**Exemplu:** pentru  $n=4$  pe ecran se va afișa:

(10p.)

```
27 0 0 0
9 18 0 0
3 6 9 0
1 2 3 4
```

57. Scrieți un program C/C++ care citește de la tastatură trei numere naturale  $n$ ,  $a$  și  $b$

( $2 < n < 25$ ,  $0 < a < n$ ,  $0 < b < n$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane numerotate de la 1 la  $n$ , formată numai din valori 0, 1, 2, 3 și 4 astfel încât: elementele aflate pe linia  $a$  sau coloana  $b$  sunt egale cu 0, cele aflate deasupra liniei  $a$  și la stânga coloanei  $b$  sunt egale cu 1, cele aflate deasupra liniei  $a$  și la dreapta coloanei  $b$  sunt egale cu 2, cele aflate sub linia  $a$  și la stânga coloanei  $b$  sunt egale cu 3, iar elementele aflate sub linia  $a$  și la dreapta coloanei  $b$  sunt egale cu 4 ca în exemplul de mai jos.

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

**Exemplu:** pentru  $n=5$ ,  $a=4$ ,  $b=3$  se construiește în memorie și se afișează matricea alăturată: (10p.)

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

56. Scrieți un program C/C++ care citește de la tastatură două numere naturale  $n$  și  $k$  ( $2 < n < 25$ ,  $0 < k < n$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane formată numai din valori 1, 2, 3 și 4 astfel încât: elementele aflate la intersecția primelor  $k$  linii cu primele  $k$  coloane sunt egale cu 1, elementele aflate la intersecția primelor  $k$  linii cu

ultimele  $n-k$  coloane sunt egale cu 2, elementele aflate la intersecția ultimelor  $n-k$  linii cu primele  $k$  coloane sunt egale cu 3, elementele aflate la intersecția ultimelor  $n-k$  linii cu ultimele  $n-k$  coloane sunt egale cu 4 ca în exemplul de mai jos.

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

**Exemplu:** pentru  $n=5$ ,  $k=3$  se construiește în memorie și se afișează matricea alăturată. (10p.)

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

54. Scrieți un program C/C++ care citește de la tastatură un număr natural **par**,  $n$  ( $4 \leq n \leq 10$ ), și un număr natural  $x$ , cu exact 3 cifre, și care construiește în memorie un tablou

bidimensional cu  $n$  linii și  $n$  coloane, ce are elementele de pe diagonala principală egale cu prima cifră a numărului  $x$ , elementele de pe diagonala secundară egale cu ultima cifră a numărului  $x$ , iar restul elementelor egale cu cifra din mijloc a numărului  $x$ .

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

**Exemplu:** dacă se citesc de la tastatură  $n=4$  și  $x=123$  atunci se afișează tabloul alăturat. (10p.)

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

48. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $2 < n < 10$ ) și care construiește în memorie și afișează pe ecran un tablou bidimensional cu  $n$  linii și  $n$  coloane astfel încât, parcurgându-l linie cu linie de sus în jos și fiecare linie de la stânga la dreapta, se obțin primele  $n^2$  numere pare nenule, în ordine strict crescătoare, ca în exemplu.

**Exemplu:** pentru  $n=4$ , se construiește și se afișează tabloul alăturat. (10p.)

```
2 4 6 8
10 12 14 16
18 20 22 24
26 28 30 32
```

47. Scrieți un program C/C++ care citește de la tastatură două numere naturale  $m$  și  $n$  ( $0 < m \leq 10$ ,  $0 < n \leq 10$ ) și elementele unui tablou bidimensional cu  $m$  linii și  $n$  coloane care reține numere naturale mai mici decât 100; programul va construi și va afișa pe ecran un nou tablou bidimensional cu  $n$  linii și  $m$  coloane, obținut din primul prin rotirea acestuia în sens trigonometric cu  $90^\circ$ , ca în exemplu. (10p.)

**Exemplu:** pentru  $m=4$ ,

$n=5$  și tabloul:

```
21 22 23 24 25
26 27 28 29 30
31 32 33 34 35
36 37 38 39 40
```

se va afișa:

```
25 30 35 40
24 29 34 39
23 28 33 38
22 27 32 37
21 26 31 36
```

44. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $0 < n \leq 23$ ) și apoi construiește în memorie o matrice cu  $n$  linii și  $n$  coloane, formată din numere naturale nenule, mai mici sau egale cu  $n$ , astfel încât să **nu** existe două linii cu aceeași sumă a elementelor și nici două coloane cu aceeași sumă a elementelor. Programul va afișa matricea pe ecran, câte o linie a matricei pe o linie a ecranului, cu un spațiu între elementele fiecărei linii.

**Exemplu:** dacă  $n=3$  atunci o soluție posibilă este următoarea matrice:

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

43. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $0 < n \leq 23$ ) și apoi construiește în memorie o matrice cu  $n$  linii și  $n$  coloane astfel încât elementele situate pe diagonala principală să fie egale cu 2, cele situate deasupra diagonalei principale să fie egale cu 1, iar cele situate sub diagonala principală să fie egale cu 3.

Programul va afișa matricea pe ecran, câte o linie a matricei pe o linie a ecranului, cu câte un spațiu între elementele fiecărei linii.

**Exemplu:** dacă  $n$  este 4 atunci programul va construi și va afișa matricea alăturată. (10p.)

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



41. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $1 \leq n \leq 23$ ) și apoi construiește în memorie o matrice cu  $n$  linii și  $n$  coloane, numerotate de la 1 la  $n$ , astfel încât fiecare element situat pe o linie  $i$  ( $1 \leq i \leq n$ ) și pe o coloană  $j$  ( $1 \leq j \leq n$ ) va fi egal cu suma dintre  $i$  și  $j$ . Programul va afișa matricea pe ecran, câte o linie a matricei pe o linie a ecranului, elementele de pe aceeași linie fiind separate prin câte un spațiu.

**Exemplu:** dacă  $n=4$ , se va afișa matricea alăturată. (10p.)

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

38. Se consideră o matrice pătratică cu  $n$  linii și  $n$  coloane ( $1 \leq n \leq 30$ ), ce memorează numere întregi nenule de cel mult două cifre fiecare.

Scrieți un program C/C++ care citește de la tastatură valoarea  $n$  și elementele matricei și care afișează pe ecran, ultima cifră a produsului acelor elemente de pe diagonala secundară ce au proprietatea că sunt valori minime pe coloanele lor. Dacă nu există astfel de elemente în

```
3 4 90 10
25 2 7 9
18 3 10 4
3 7 20 3
```

matrice, se va afișa mesajul NU EXISTA.

**Exemplu:** pentru  $n=4$  și matricea alăturată se va afișa pe ecran valoarea 1 ( $3 \cdot 7 = 21$ ). (

36. Se consideră o matrice cu  $n$  linii și  $m$  coloane ( $1 \leq n \leq 30$ ,  $1 \leq m \leq 30$ ), ce memorează numere întregi de cel mult 4 cifre fiecare. Scrieți un program C/C++ care citește de la tastatură valorile  $n$ ,  $m$  și elementele matricei și care afișează pe ecran, separate prin câte un spațiu, valorile minime de pe fiecare coloană, în ordine de la ultima la prima coloană.

```
3 4 90 10
25 2 7 9
18 3 10 4
```

**Exemplu:** pentru  $n=4$ ,  $m=4$  și matricea alăturată se vor afișa pe ecran valorile 3 7 2 3. (10p.)

35. Scrieți programul C/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. (10p.)

**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
```

34. Scrieți programul C/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 crescătoare a valorii, pe linii și coloane, primele  $n_2$  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.

(10p.)

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

30. 5. Scrieți un program C/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  $A_{i,j}$  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.

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

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

29. Scrieți un program C/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 nenule mai mici decât 100, și afișează pe ecran produsul numerelor “**pivot**” pentru matricea  $A$  dacă există astfel de numere, altfel va afișa mesajul **NU EXISTA**. Un număr natural  $x$  este “**pivot**” pentru matricea  $A$  dacă înmulțind fiecare element de pe prima coloană cu numărul  $x$ , se obțin, în aceeași ordine, elementele unei alte coloane din matrice.

2	7	4	8	4
1	1	2	4	2
3	12	6	12	3
1	22	2	4	2
5	10	10	20	8

**Exemplu:** pentru matricea din figura alăturată se afișează 8.

27. Scrieți un program C/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

```

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

```

linie fiind separate prin câte un spațiu.

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

26. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $2 \leq n \leq 9$ ) și elementele unui tablou bidimensional  $A$  cu  $n$  linii și  $n$  coloane, care memorează numere naturale mai mici decât 10, 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. Dacă nu există astfel de elemente, programul va afișa pe ecran mesajul **NU EXISTA**.

**Exemplu:** pentru matricea din figura alăturată se afișează, nu neapărat în

1	1	2	9
9	2	3	3
8	4	1	1
0	2	9	3

această ordine, valorile: 4 9 ( $4=1*2*2$ ;  $9=3*1*3$ ).

25. 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/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 numerează de la 1 la  $m$ , respectiv de la 1 la  $n$ . Numerele se vor afișa separate prin câte un spațiu.

**Exemplu:** pentru  $m=4$  și  $n=4$  și tabloul alăturat se va afișa: 1 2

```

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

```

23. Un tablou bidimensional  $A$  cu  $m$  linii și  $n$  coloane ( $1 \leq m \leq 100$ ,  $1 \leq n \leq 100$ ) conține pe prima linie numerele  $1, 2, \dots, n$ , iar pe prima coloană numerele  $1, 2, \dots, m$ . Celelalte elemente ale tabloului sunt date de relația:  $A_{i,j} = A_{i-1,j} + A_{i,j-1}$ . Scrieți un program C/C++ care citește de la tastatură numerele  $m$  și  $n$  și afișează pe ecran ultima cifră a elementului de pe ultima linie și ultima coloană a tabloului.

**Exemplu:** pentru  $m=3$  și  $n=4$  se va afișa 5 deoarece elementele tabloului  $A$  sunt:

(10p.)

```
1 2 3 4
2 4 7 11
3 7 14 25
```

22. Scrieți un program C/C++ care citește de la tastatură numerele naturale  $m$  și  $n$  din intervalul  $[1, 24]$ , 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.

**Exemplu:** pentru  $m=4$  și  $n=3$  se va construi și afișa tabloul alăturat. (10p.)

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

20. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $2 < n < 20$ ) și construiește în memorie o matrice cu  $n$  linii și  $n$  coloane, numerotate de la 1 la  $n$ , în care fiecare element aflat pe chenarul exterior al matricei este egal cu suma dintre indicii liniei și indicii coloanei pe care se află, iar fiecare dintre celelalte elemente este egal cu suma celor trei "vecini" situați în matrice pe linia anterioară. Două elemente din matrice se numesc vecine dacă se găsesc alături pe linie, coloană sau diagonală. Chenarul exterior al unei matrice este format din prima linie, ultima linie, prima coloană și ultima coloană.

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 obține matricea alăturată. (10p.)

```
2 3 4 5 6
3 9 12 15 7
4 24 36 34 8
5 64 94 78 9
6 7 8 9 10
```

19. Scrieți un program C/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 linie impară este egal cu suma dintre indicii liniei și coloanei pe care se află și fiecare element aflat pe o linie pară este egal cu cel mai mic dintre elementele aflate pe linia anterioară și pe aceeași coloană cu el sau pe linia anterioară și pe una dintre coloanele vecine cu cea pe care se află el.

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. (10p.)

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

18. Scrieți un program C/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 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. (10p.)

```
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
```

17. Scrieți un program C/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. (10p.)

```
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
```

16. Scrieți un program C/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 0, elementele care se află deasupra ambelor diagonale sunt egale cu 1, elementele care se află sub ambele diagonale sunt egale cu 2, 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. (10p.)

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

15. Scrieți un program C/C++ care citește de la tastatură un număr natural cu exact 5 cifre și construiește în memorie o matrice cu 6 linii și 6 coloane, numerotate de la 1 la 6, formată astfel:

- elementele aflate pe diagonala principală sunt toate nule;
- elementele de pe linia 1, aflate deasupra diagonalei principale precum și elementele de pe coloana 1, aflate sub diagonala principală au toate valoarea egală cu cifra unităților numărului citit;
- elementele de pe linia 2, aflate deasupra diagonalei principale precum și elementele de pe coloana 2, aflate sub diagonala principală au toate valoarea egală cu cifra zecilor numărului citit, și așa mai departe, ca în exemplu.

Matricea astfel construită va fi afișată pe ecran, 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 scrisă alăturat. **(10p.)**

```
0 1 1 1 1 1
1 0 3 3 3 3
1 3 0 7 7 7
1 3 7 0 8 8
1 3 7 8 0 2
1 3 7 8 2 0
```

14. Scrieți un program C/C++ care citește de la tastatură un număr natural nenul, cu maximum

8 cifre, construiește în memorie și afișează apoi pe ecran o matrice pătratică, având numărul de linii și de coloane egal cu numărul de cifre ale numărului dat, completată cu cifrele numărului citit.

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, și așa mai departe, ca în exemplu.

$$\begin{pmatrix} 9 & 5 & 3 & 1 \\ 9 & 5 & 3 & 1 \\ 9 & 5 & 3 & 1 \\ 9 & 5 & 3 & 1 \end{pmatrix}$$

**Exemplu:** dacă se citește numărul 1359, matricea construită va fi cea alăturată. **(10p.)**

13. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $2 < n < 25$ ) 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 coloana  $i$  ( $1 \leq i \leq n$ ), aflate deasupra diagonalei secundare, au valoarea egală cu  $i$ ;
- elementele de pe linia  $n - i + 1$  ( $1 \leq i \leq n$ ), aflate sub diagonala secundară, au valoarea egală cu  $i$ .

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. **(10p.)**

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

12. Scrieți un program C/C++ care citește de la tastatură un număr natural  $n$  ( $2 < n < 25$ ) 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: elementul din linia  $i$  și coloana  $j$  primește ca valoare ultima cifră a produsului  $i*j$  ( $1 \leq i \leq n$  și  $1 \leq j \leq n$ ).

Programul va afișa matricea astfel construită pe ecran, câte o linie a matricei

pe o linie a ecranului, elementele fiecărei linii fiind separate prin câte un spațiu.

**Exemplu:** pentru  $n=4$  se va afișa matricea alăturată. (10p.)

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

11. Scrieți un program C/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 afișează pe ecran cea mai mare valoare dintre aceste minime.

**Exemplu:** pentru  $m=3$ ,  $n=5$  și matricea

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

, se afișează pe ecran valoarea 6

(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. Cea mai mare dintre aceste trei valori este 6).

10. Scrieți un program C/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 matricea 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 pare**, 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: (10p.)

```
0 4 16
36 64 100
```

9. Se consideră tabloul bidimensional cu  $n$  linii și  $n$  coloane ce conține numere naturale cu cel mult patru cifre fiecare. Scrieți programul C/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,

```
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
```

se va afișa:

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

(10p.)

8. Scrieți un program C/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  $n=2$ ,  $p=3$  se va afișa tabloul alăturat:

**(10p.)**

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

5. Scrieți un program C/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  $A_{ij}$  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. **(10p.)**

**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
```

4. Scrieți un program C/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. **(10p.)**

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

3. Scrieți un program C/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. (10p.)

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

1. Scrieți un program C/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  $A_{ij}$  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.

(10p.)

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