

# Teme Divide et Impera

1. Pe o tablă de joc se află piese aparținând celor doi jucători. Tabla este memorată ca o matrice în care zonele corespunzătoare fundalului sunt notate cu 0, iar cele corespunzătoare pieselor cu 1 sau 2, în funcție de jucătorul cărora le aparțin. Știind că jucătorul cu cele mai multe piese câștigă jocul, să se determine care este acesta. Se consideră că două zone (elemente ale matricei) aparțin aceluiași obiect dacă ele sunt vecine și sunt situate pe aceeași linie sau pe aceeași coloană.

**Date de intrare:** fișierul date.in conține

-pe prima linie două numere naturale,  $m$  și  $n$ ;

-pe următoarele linii elementele unei matrice cu  $m$  linii și  $n$  coloane, cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține numărul corespunzător jucătorului câștigător.

**Exemplu:**

date.in	date.out
4 6 0 0 0 1 1 2 2 0 0 0 1 0 2 0 1 0 0 0 0 0 1 0 0 2	2

2. O tablă are dimensiunile  $2^n * 2^n$  și are o gaură la poziția (lg, cg). Se cere să se acopere tabla complet, cu excepția găurii, cu piese de forma dată mai jos, care se pot roti cu  $90^\circ$ ,  $180^\circ$  sau  $270^\circ$ .



**Date de intrare:** fișierul date.in conține trei numere naturale,  $n$ , lg și cg, cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține pe fiecare linie câte 6 numere naturale, separate prin câte un spațiu, reprezentând coordonatele pătratelor din care este formată fiecare piesă așezată pe tablă.

**Exemplu:**

date.in	date.out
2 4 3	2 2 2 3 3 2 1 1 1 2 2 1 1 3 1 4 2 4 3 1 4 1 4 2 3 3 3 4 4 4

3. Pe o tablă de joc se află piese aparținând celor doi jucători. Tabla este memorată ca o matrice în care zonele corespunzătoare fundalului sunt notate cu 0, iar cele corespunzătoare pieselor cu 1 sau 2, în funcție de jucătorul cărora le aparțin. Să se elimine un număr minim de piese (oricare dintre cele aflate pe tablă), astfel încât cei doi jucători să aibă pe tablă același număr de piese. Se consideră că două zone (elemente ale matricei) aparțin aceluiași obiect dacă ele sunt vecine și sunt situate pe aceeași linie sau pe aceeași coloană.

**Date de intrare:** fișierul date.in conține

-pe prima linie două numere naturale,  $m$  și  $n$ ;

- pe următoarele linii elementele unei matrice cu  $m$  linii și  $n$  coloane, cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține tabla de joc, în urma eliminării pieselor conform cerinței

**Exemplu:**

date.in	date.out
4 6	0 0 0 0 0 1
0 0 0 1 1 1	2 0 0 0 0 0
2 0 0 0 1 0	2 0 1 0 0 0
2 0 1 0 0 0	0 0 1 0 0 2
0 0 1 0 0 2	

4. Pe o suprafață este trasată o curbă închisă. Suprafața este memorată ca o matrice cu  $M$  linii și  $N$  coloane, cu valori 0, iar punctele aparținând curbei sunt notate cu 1. Dându-se un punct din interiorul zonei marginite de curbă, se cere să se umple cu 1 această zonă.

**Date de intrare:** fișierul date.in conține

-pe prima linie două numere naturale,  $m$  și  $n$ ;

- pe următoarele  $m$  linii elementele unei matrice cu  $m$  linii și  $n$  coloane, cu semnificația din enunț.

-pe ultima linie două numere  $l0, c0$ , reprezentând linia, respectiv coloana pe care se află punctul.

**Date de ieșire:** fișierul date.out conține matricea transformată.

**Exemplu:**

date.in	date.out
10 8	0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0	0 0 1 1 1 1 0 0
0 0 1 1 1 1 0 0	0 1 1 1 1 1 0 0
0 1 1 0 0 1 0 0	1 1 1 1 1 1 1 0
1 1 0 0 1 1 0 0	1 1 1 1 1 1 1 0
1 0 0 0 0 0 1 0	1 1 1 1 1 1 1 0
1 1 0 1 1 1 1 0	0 1 1 1 0 0 0 0
0 1 0 1 0 0 0 0	0 1 1 1 0 0 0 0
0 1 1 1 0 0 0 0	0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0	
5 4	

5. O fotografie alb-negru este memorată ca o matrice în care zonele corespunzătoare obiectelor sunt notate cu 1, iar cele de fundal cu 0. Să se determine numărul obiectelor figurate în fotografie. Se considera că două puncte aparțin aceluiași obiect dacă ele sunt vecine și situate pe aceeași linie sau aceeași coloană.

**Date de intrare:** fișierul date.in conține

-pe prima linie două numere naturale,  $m$  și  $n$ ;

- pe următoarele linii elementele unei matrice cu  $m$  linii și  $n$  coloane, cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține numărul cerut.

**Exemplu:**

date.in	date.out
4 6	4
0 0 0 1 1 0	
1 0 0 0 1 0	
1 0 1 0 0 0	
0 0 1 0 0 1	

6. Calul lui Făt Frumos se află în partea de nord a unui teren cuprins pe alocuri de flăcări. Terenul este memorat ca o matrice în care zonele libere sunt notate cu 0, iar cele cuprinse de flăcări cu 1. Calul va evita zonele cu foc și se va deplasa doar pe căsuțele vecine (sus, jos, stânga și dreapta). Știind că el poate porni din oricare dintre punctele aflate pe marginea de nord a terenului și că trebuie să ajungă la Făt Frumos, care se află la sud de acest teren, să se determine punctele de pe marginea de sud a terenului în care el poate ajunge.

**Date de intrare:** fișierul date.in conține

-pe prima linie două numere naturale,  $m$  și  $n$ ;

- pe următoarele linii elementele unei matrice cu  $m$  linii și  $n$  coloane, cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține, în ordine crescătoare, numerele de ordine ale coloanelor în care poate ajunge, conform cerinței.

Linii și coloanele matricei sunt numerotate începând cu 1

**Exemplu:**

date.in	date.out
6 7	1 6 7
0 1 0 0 1 0 0	
0 1 0 0 1 0 0	
0 0 1 1 1 0 0	
0 0 1 0 1 0 0	
0 1 0 1 0 1 0	
0 1 0 0 1 0 0	

7. O fotografie alb-negru este memorată ca o matrice în care zonele corespunzătoare obiectelor sunt notate cu 1, iar cele de fundal cu 0. Să se coloreze fiecare obiect cu o culoare diferită. Se considera ca două puncte aparțin aceluiași obiect dacă ele sunt vecine și situate pe aceeași linie sau aceeași coloană.

**Date de intrare:** fișierul date.in conține

-pe prima linie două numere naturale,  $m$  și  $n$ ;

- pe următoarele linii elementele unei matrice cu  $m$  linii și  $n$  coloane, cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține matricea transformată conform cerinței.

Se cere o singură soluție.

**Exemplu:**

date.in	date.out
4 6	0 0 0 1 1 0
0 0 0 1 1 0	2 0 0 0 1 0
1 0 0 0 1 0	2 0 3 0 0 0
1 0 1 0 0 0	0 0 3 0 0 4
0 0 1 0 0 1	

8. Romeo și Julieta se află într-un labirint memorat ca o matrice în care culorile sunt notate cu 0, iar zidurile cu 1. Să se verifice dacă Romeo poate ajunge la Julieta, știind că el se poate deplasa doar în sus, jos, stânga și dreapta.

**Date de intrare:** fișierul date.in conține

-pe prima linie două numere naturale,  $m$  și  $n$ ;

-pe a doua linie două numere naturale, LR și CR, reprezentând linia și coloana din labirint pe care se află Romeo

-pe a treia linie două numere naturale, LJ și CJ, reprezentând linia și coloana din labirint pe care se află Julieta

- pe următoarele  $m$  linii elementele unei matrice cu  $m$  linii și  $n$  coloane, cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține mesajul DA, dacă răspunsul este afirmativ, sau mesajul NU în caz contrar.

Linii și coloanele matricei sunt numerotate începând cu 1

**Exemplu:**

date.in	date.out
6 7	NU
3 1	
5 5	
0 1 0 0 1 0 0	
0 1 0 0 1 0 0	
0 0 1 1 1 0 0	
0 0 1 0 1 0 0	
0 1 0 1 0 1 0	
0 1 0 0 1 0 0	

9. Să se construiască o matrice pătratică, având  $2^n$  linii și  $2^n$  coloane, cu elemente din mulțimea  $\{0,1\}$  urmând următorul procedeu:

-se împarte tabloul în patru subtablouri de latură  $2^{n-1}$

-se plasează valori 1 în subtabloul dreapta-sus și valori nule în celelalte subtablouri

- se reia procedeu pentru fiecare dintre celelalte trei subtablouri, până la obținerea unor subtablouri de latură 1, care vor avea valori nule.

**Date de intrare:** fișierul date.in conține numărul  $n$ , cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține matricea construită.

**Restricții și precizări:**

$n < 10$

**Exemplu:**

date.in	date.out
2	0 1 1 1 0 0 1 1 0 1 0 1 0 0 0 0

10. Să se construiască o matrice pătratică, având  $3^n$  linii și  $3^n$  coloane, cu elemente din mulțimea  $\{0,1\}$  urmând următorul procedeu:

-se împarte tabloul în nouă subtablouri de latura  $3^{n-1}$

-se plasează valori 1 în subtabloul din mijloc

- se plasează valori 2 în subtablourile de sud și nord și valori 3 în subtablourile de est și vest

- în etapa următoare se reia procedeu pentru fiecare dintre celelalte patru subtablouri

Procedeu se realizează în  $k$  etape sau până când se obțin subtablouri formate din câte un singur element, care vor avea valoare nulă.

**Date de intrare:** fișierul date.in conține numerele  $n$  și  $k$ , în această ordine, cu semnificația din enunț.

**Date de ieșire:** fișierul date.out conține matricea construită.

**Restricții și precizări:**

$n < 10, k < 10$

**Exemplu:**

date.in	date.out
2 2	0 2 0 2 2 2 0 2 0 3 1 3 2 2 2 3 1 3 0 2 0 2 2 2 0 2 0 3 3 3 1 1 1 3 3 3 3 3 3 1 1 1 3 3 3 3 3 3 1 1 1 3 3 3 0 2 0 2 2 2 0 2 0 3 1 3 2 2 2 3 1 3 0 2 0 2 2 2 0 2 0

11. Dată o mulțime de puncte în plan (prin coordonatele lor), să de determine cea mai apropiată pereche de puncte (se vor afișa distanța și punctele) <http://infoarena.ro/problema/cmap> - pentru detalii.

12. Se dă un arbore binar prin parcurgerea în preordine în care apar și valorile null (semnificând lipsa unui fiu). Sa se construiască în memorie arborele (alocat dinamic) și să se verifice dacă este arbore binar de căutare  $O(n)$ , unde  $n$ =numărul de vârfuri

**Date de intrare :**

Date.in	Date.out
4 1 null 3 null null 7 6 null null null	da

13. Se dau  $n$  numere naturale. Să se afișeze al  $k$ -ulea lea cel mai mic element din șir.

**Date de intrare:** fisierul date.in contine

- pe prima linie un numar natural  $n$  si un numar natural  $k$  s
- pe a doua linie, un șir de  $n$  numere naturale, separate prin câte un spatiu.

**Date de ieșire:** fisierul date.out conține valoarea ceruta.