



OLIMPIADA NAȚIONALĂ DE INFORMATICĂ BRĂILA 26 APRILIE – 03 MAI 2002

Clasa a IX-a

Sursa: `becuri.pas`, `becuri.c`, `becuri.cpp`

Intrare: `becuri.in`

Ieșire: `becuri.out`

Problema 4

Becuri

Un panou publicitar, de formă dreptunghiulară conține becuri, unul lângă altul, aliniate pe linii și coloane.

Fiecare linie și fiecare coloană are un comutator care schimbă starea tuturor becurilor de pe acea linie sau coloană, din starea în care se află în starea opusă (stins/aprins, aprins/stins). Asupra unui bec nu se poate acționa individual. Inițial panoul are toate becurile stinse.

Cerință:

Să se realizeze un program care acționând asupra unui număr minim de linii și coloane aduce panoul din starea inițială, la o configurație dată, dacă acest lucru este posibil.

Datele de intrare: se vor citi din fișierul **BECURI . IN**, care are următoarea configurație:

- pe prima linie două numere naturale, **m** și **n**, separate prin spațiu, reprezentând numărul de linii și de coloane ale panoului;
- pe următoarele **m** linii câte **n** coloane, formate din elementele **0** sau **1**, separate prin spațiu, reprezentând configurația finală a panoului. **1** este codificarea unui bec aprins iar **0** a unui bec stins.

Datele de ieșire : se va afișa o soluție în fișierul **BECURI . OUT** astfel:

- pe prima linie a fișierului indicii coloanelor asupra cărora s-a acționat, separați prin spațiu.
- pe a doua linie a fișierului indicii liniilor asupra cărora s-a acționat, separați prin spațiu.

Dacă nu se acționează asupra liniilor sau coloanelor se va afișa pe linia corespunzătoare 0.

Numerotarea liniilor, respectiv coloanelor începe de la 1.

Dacă problema nu are soluție, pe prima linie a fișierului de ieșire se va afișa -1.

Restricții:

- $m \leq 100$, $n \leq 100$

Exemplu:

BECURI . IN						
5	6					
1	0	1	1	0	1	
1	0	1	1	0	1	
1	0	1	1	0	1	
0	1	0	0	1	0	
0	1	0	0	1	0	

BECURI . OUT		
2	5	
1	2	3

Timp maxim de execuție/test: 1 secundă.