

### Problema 3 – roboți

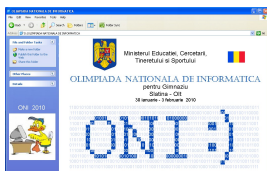
**100 puncte**

Într-o zonă dreptunghiulară cu  $p$  linii și  $q$  coloane se află  $n$  roboți. Celula din stânga sus se află pe linia 1 și coloana 1. Pentru fiecare robot se cunoaște linia și coloana pe care se află, precum și orientarea lui. Un robot poate fi orientat în una din cele patru direcții: nord, sud, est sau vest, codificate cu caracterele **N**, **S**, **E**, respectiv **V**. Fiecare robot execută  $m$  comenzi. O comandă este codificată printr-un caracter **L**, **R** sau **F**. La o comandă de tip **L**, robotul se întoarce cu 90 de grade spre stânga, în sensul invers acelor de ceasornic. La o comandă de tip **R**, robotul se întoarce cu 90 de grade spre dreapta, în sensul acelor de ceasornic. La o comandă de tip **F**, robotul se deplasează cu o poziție în sensul în care este orientat robotul.

Roboții execută simultan prima comandă din șirul lor de comenzi, apoi a doua comandă, etc.

Dacă la un moment dat, doi sau mai mulți roboți ajung în aceeași poziție, aceasta va conduce la dispariția lor, iar celula din care dispar se consideră traversată de toți cei care au dispărut.

Dacă un robot se deplasează în afara suprafeței la execuția unei comenzi, robotul dispare.



**Clasa a VII-a**

---

Se consideră trecere printr-o celulă vizitarea rezultată în urma executării unei comenzi de tip **F**. Dacă un robot trece de mai multe ori printr-o celulă, se contorizează fiecare trecere a sa. Celula din care pleacă fiecare robot se consideră trecere pentru robotul respectiv.

**Cerință**

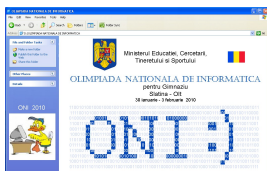
Scrieți un program care să determine:

- numărul de roboți rămași după execuția celor **m** comenzi,
- poziția celulei din zonă prin care s-a trecut cel mai des (dacă există mai multe celule, atunci se va preciza cea cu indicele de linie cel mai mic, iar dacă există mai multe cu acest indice, cea cu indicele de coloană cel mai mic) și numărul de treceri prin această celulă.

**Date de intrare**

Fișierul **roboti.in** conține:

- pe prima linie: **p** și **q**, valori separate printr-un spațiu și reprezentând numărul de linii și de coloane ale zonei;



**Clasa a VII-a**

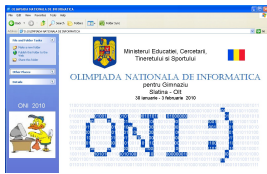
---

- pe a doua linie: valoarea  $n$ , reprezentând numărul de roboți;
- pe următoarele  $n$  linii: câte trei valori separate prin câte un spațiu, reprezentând linia, coloana și orientarea fiecărui robot;
- pe următoarea linie: valoarea  $m$  reprezentând numărul de comenzi de executat de către fiecare robot;
- pe următoarele  $n$  linii se află câte  $m$  caractere reprezentând comenzile pentru fiecare robot (mai întâi cele  $m$  comenzi pentru primul robot, apoi cele  $m$  comenzi pentru al doilea robot, ș.a.m.d.). Între caracterele unei comenzi nu există nici un spațiu.

**Date de ieșire**

Fișierul **roboti.out** va conține:

- pe prima linie numărul de roboți rămași în final



**Clasa a VII-a**

---

- pe linia a doua poziția celulei (linie și coloană) prin care s-a trecut cel mai des (dacă există mai multe celule, atunci se va preciza cea cu indicele de linie cel mai mic, iar dacă există mai multe cu acest indice, cea cu indicele de coloană cel mai mic) și numărul de treceri prin această celulă, trei valori separate prin câte un spațiu

**Restricții și precizări**

$1 \leq p, q, m \leq 50$

$2 \leq n \leq 50$

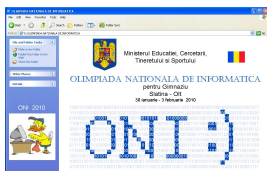
Caracterele cu care se codifică orientările pot fi doar N, S, E, V, iar cele pentru comenzi L, R sau F (litere mari).

Inițial, nu există doi sau mai mulți roboți în aceeași celulă.

Se acordă 50% din punctaj pentru rezolvarea cerinței a) și 50% pentru cerința b).

**Exemplu:**

<code>roboti.in</code>	<code>roboti.out</code>	Explicație
------------------------	-------------------------	------------



**Clasa a VII-a**

4 4 3 1 1 E 1 3 V 4 4 S 2 FL FF FF	0 1 2 2	După prima comandă, roboții 1 și 2 ar trebui să ajungă în aceeași celulă, deci dispar. Al treilea robot conform primei deplasări va părăsi zona, deci dispăre. În final vor fi 0 roboți. Doi roboți au ajuns în celula de pe linia 1 și coloana 2, după care au dispărut. Prin celelalte celule s-a trecut maxim o singură dată.
--	------------	--

**Timpe maxim de execuție/test:** 1 secundă