



SOCIETATEA DE ȘTIINȚE MATEMATICE DIN ROMÂNIA - FILIALA CLUJ



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XV-a, 8–9 mai 2015



## Clasa a IX-a

### Problema 1 – PĂTRATE

Pe o foaie cu pătrățele se stabilește un sistem de coordonate în care o intersecție primește coordonatele  $(0,0)$ , astfel încât fiecare intersecție a caroiajului are coordonate numere întregi.

Pe acest caroiaj se desenează un pavaj cu dreptunghiuri, în care fiecare dreptunghi are o lățime  $L$  și o înălțime  $H$  date, iar punctul de coordonate  $(0,0)$  este un colț de dreptunghiuri. În acest mod, fiecare intersecție a pavajului are coordonate de forma  $(L*i, H*j)$ , cu  $i$  și  $j$  întregi.

Se mai dă o pereche de întregi  $x$  și  $y$  și se consideră segmentul de dreaptă ce unește punctul de coordonate  $(0,0)$  cu punctul de coordonate  $(x,y)$ .

#### Cerință

Se cere să se determine câte dreptunghiuri ale pavajului sunt intersectate de segmentul considerat. Un dreptunghi se consideră intersectat de segment dacă are cel puțin un punct interior comun. Cu alte cuvinte, dacă segmentul doar atinge colțul unui dreptunghi, nu se consideră că îl taie.

#### Date de intrare

Fișierul `patrate.in` conține, pe o singură linie, 4 numere separate prin spații:  $L H X Y$ .

#### Date de ieșire

În fișierul `patrate.out` se va scrie un singur număr întreg reprezentând numărul de dreptunghiuri tăiate de segment.

#### Restricție

- $1 \leq L, H, X, Y \leq 200000000$

#### Exemplu

<code>patrate.in</code>	<code>patrate.out</code>
4 2 6 3	2

Timp maxim de execuție/test: 0.5 secunde.

Total memorie disponibilă: 10 MB.

Dimensiunea maximă a sursei: 5 KB.

SOCIETATEA DE ȘTIINȚE MATEMATICE DIN ROMÂNIA - FILIALA CLUJ



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XV-a, 8–9 mai 2015



### Clasa a IX-a

### Problema 2 – PARITATE

Gigel tocmai a învățat noțiunea de paritate a unui număr natural. Vrând să-și testeze atenția, Gigel s-a gândit la următorul joc: el scrie un șir de numere naturale distincte din care, mai apoi, elimină o parte dintre elemente, păstrând cele mai mari valori din șir în ordinea în care acestea apăreau în șirul inițial și, în final, se întreabă câte perechi de elemente succesive de parități diferite sunt în șirul astfel obținut.

#### Cerință

Cunoscându-se dimensiunea și elementele unui șir de numere naturale distincte, să se răspundă la  $t$  întrebări de forma „Câte perechi de elemente succesive de parități diferite sunt în șirul obținut prin păstrarea și nemodificarea ordinii celor mai mari  $k$  numere din șirul dat?”.

#### Date de intrare

Fișierul de intrare **paritate.in** conține pe prima linie, separate prin câte un spațiu, două numere naturale  $n$  și  $t$ , reprezentând, în ordine, dimensiunea șirului și numărul de întrebări.

Pe cea de-a doua linie a fișierului se găsesc, separate prin câte un spațiu,  $n$  numere naturale distincte reprezentând elementele șirului.

Pe fiecare dintre următoarele  $t$  linii ale fișierului se găsește câte o valoare  $k$  cu semnificația din întrebare.

#### Date de ieșire

În fișierul de ieșire **paritate.out** se va scrie, pe fiecare dintre cele  $t$  linii ale sale, câte o valoare reprezentând, în ordine, răspunsul la întrebarea corespunzătoare din fișierul de intrare.

#### Restricții și precizări

- $1 \leq n, t \leq 10000$
- $2 \leq k \leq n$
- Niciun element al șirului nu depășește valoarea **1000000**.
- Punctajul corespunzător unui test se acordă doar dacă s-a răspuns corect la toate întrebările din testul respectiv.

#### Exemplu

<b>paritate.in</b>	<b>paritate.out</b>	<i>Explicație</i>
7 3	1	Prin păstrarea celor mai mari 3 elemente, șirul obținut este <b>5 8 10</b> . Există o singură pereche de elemente succesive de paritate diferită.
2 5 1 4 8 10 3	0	
3	2	
2		
5		

Timp maxim de execuție/test: **0.5** secunde.

Total memorie disponibilă: **10** MB.

Dimensiunea maximă a sursei: **5** KB.

SOCIETATEA DE ȘTIINȚE MATEMATICE DIN ROMÂNIA - FILIALA CLUJ



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XV-a, 8–9 mai 2015



**Clasa a X-a**

**Problema 1 – DOMINO**

Un joc de domino constă în niște piese dreptunghiulare, pe fiecare piesă fiind înscrise două numere, câte unul la fiecare capăt al piesei. Piesele se pot înșira una după alta doar dacă numărul înscris pe capătul din dreapta al fiecărei piese este egal cu numărul înscris pe capătul din stânga al piesei următoare.

**Cerință**

Dându-se numerele înscrise pe fiecare piesă de domino, se cere să se determine un aranjament în care să se înșire în linie toate piesele. Fiecare piesă se poate folosi exact o dată și poate fi plasată în două moduri, prin eventuala inversare a numerelor din stânga cu cel din dreapta prin rotirea piesei cu 180 de grade. Se garantează existența unei soluții.

**Date de intrare**

Datele se citesc din fișierul **domino.in** având următorul format:

- pe prima linie, numărul  $N$  de piese de domino
- pe fiecare din următoarele  $N$  linii, câte două numere,  $S_i$  și  $D_i$ , reprezentând numerele înscrise pe piesa de domino corespunzătoare.

**Date de ieșire**

Rezultatele se vor scrie în fișierul **domino.out** care va avea  $N$  linii, descriind piesele din șir de la stânga spre dreapta. Pe fiecare linie, se vor afla două numere,  $P_i$  și  $R_i$ .  $P_i$  reprezintă numărul de ordine, din fișierul de intrare, a piesei ce se va plasa pe poziția  $i$ , iar  $R_i$  va fi 0 dacă piesa se așează normal și 1 dacă piesa se așează rotită 180 de grade. Piesele din fișierul de intrare sunt numerotate de la 1 la  $N$ .

**Restricții**

- $0 \leq S_i, D_i \leq 1000000$
- $1 \leq N \leq 10000$

**Exemplu**

<b>domino.in</b>	<b>domino.out</b>
6	1 0
1 10	2 1
1 10	6 0
2 4	3 0
6 5	5 1
5 4	4 1
1 2	

Timp maxim de execuție/test: 0.5 secunde.

Total memorie disponibilă: 10 MB.

Dimensiunea maximă a sursei: 5 KB.

SOCIETATEA DE ȘTIINȚE MATEMATICE DIN ROMÂNIA - FILIALA CLUJ



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XV-a, 8–9 mai 2015



**Clasa a X-a**

**Problema 2 – POMI**

Bunicul unuia dintre elevii clasei a X-a i-a invitat pe colegii nepotului său la țară. Pentru a doua zi bunicul le-a promis copiilor că-i va lăsa în livadă la cules de fructe. Livada bunicului este de formă dreptunghiulară și este parcelată sub forma unui caroiaj, fiecare parcelă de dimensiuni elementare având un anumit număr de pomi fructiferi.

Dorind să testeze istețimea copiilor și, totodată, să nu fie persoane nemulțumite, bunicul le-a promis acestora că îi va lăsa să păstreze fructele culese doar dacă identifică o zonă a livezii de formă dreptunghiulară care să conțină un număr de pomi care să poată fi împărțit în totalitate și în mod echitabil tuturor copiilor. În plus, bunicul ar vrea să afle de la copii și în câte modalități poate fi stabilită o astfel de zonă.

La rândul lor, copiii, încântați de provocarea bunicului, doresc să afle care este numărul maxim de pomi care se găsesc într-o zonă care respectă condițiile fixate de bunic.

**Cerință**

Cunoscându-se dimensiunile livezii, numărul de pomi specific fiecărei parcele și numărul de copii, să se determine numărul maxim de pomi de pe care vor culege copiii fructe și în câte moduri poate fi stabilită o zonă care respectă condițiile fixate de bunic.

**Date de intrare**

Fișierul de intrare **pomi.in** conține pe prima linie un număr natural **p**, pentru toate testele de intrare, numărul **p** putând avea doar valoarea **1** sau valoarea **2**.

Pe cea de-a doua linie a fișierului se găsesc, separate prin câte un spațiu, trei numere naturale **n**, **m** și **k**, reprezentând, în ordine, dimensiunile livezii și numărul de copii.

Pe fiecare dintre următoarele **n** linii ale fișierului se găsesc, separate prin câte un spațiu, câte **m** valori reprezentând numărul de pomi care se găsesc în parcelele livezii.

**Date de ieșire**

Fișierul de ieșire **pomi.out** va conține o singură linie pe care va fi scris rezultatul primei cerințe dacă valoarea lui **p** este **1** (numărul maxim de pomi de pe care vor culege copiii fructe) sau rezultatul celei de-a doua cerințe dacă valoarea lui **p** este **2** (numărul de modalități în care poate fi stabilită o zonă care respectă condițiile fixate de bunic).

**Restricții și precizări**

- $1 \leq n, m \leq 400$
- $2 \leq k \leq 1000$
- Numărul de pomi fructiferi existenți într-o parcelă este nenul și maxim 10.
- Fiecare zonă a livezii este de formă dreptunghiulară și are laturile paralele cu laturile acesteia.
- Se garantează că există cel puțin o modalitate de stabilire a zonei și că numărul de modalități de stabilire a zonei nu depășește valoarea 1000000000.
- Prima cerință reprezintă 30% din punctaj, iar cea de-a doua cerință reprezintă 70% din punctaj.

## Exemple

<code>pomi.in</code>	<code>pomi.out</code>	<i>Explicație</i>
1	12	În zona dreptunghiulară pusă în evidență sunt <b>12</b> pomi, maxim posibil.
3 3 3		
2 5 1		
4 3 2		
1 1 1		

<code>pomi.in</code>	<code>pomi.out</code>	<i>Explicație</i>
2	19	Sunt <b>19</b> variante de stabilire a unei zone a livezii care respectă condițiile impuse.
3 4 3		
2 5 1 3		
4 3 2 5		
2 1 6 2		

Timp maxim de execuție/test: **1** secundă.

Total memorie disponibilă: **2** MB.

Dimensiunea maximă a sursei: **5** KB.

SOCIETATEA DE ȘTIINȚE MATEMATICE DIN ROMÂNIA - FILIALA CLUJ



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XV-a, 8–9 mai 2015



## Clasele XI-XII

### Problema 1 – ȘOSEA

Proiectarea unei șosele într-o zonă montană necesită găsirea unui compromis între costul realizării șoselei și costul utilizării ei. Astfel, cu cât pantele sunt mai puține, cu atât consumul de combustibil al automobilelor este mai mic, dar costul lucrărilor crește prin realizarea de tuneluri sau viaducte.

Considerăm cunoscut profilul terenului, astfel, pe fiecare unitate de lungime, terenul poate urca cu o unitate de înălțime, poate coborî cu o unitate de înălțime sau poate fi orizontal. Șoseaua poate urma suprafața terenului, poate trece printr-un tunel sau peste un viaduct. Tunelurile și viaductele pot fi doar orizontale. Astfel, un tunel poate începe doar la începutul unei porțiuni de teren care urcă și se poate termina doar la sfârșitul primei porțiuni care coboară înapoi la aceeași altitudine. Similar, un viaduct poate începe doar la începutul unei porțiuni de teren în coborâre și termină acolo unde terenul urcă înapoi la aceeași altitudine.

Costul unui tunel sau viaduct este egal cu lungimea tunelului sau viaductului înmulțită cu un coeficient fix. De asemenea, fiecare unitate de lungime de șosea în urcare sau coborâre are un cost fix, care reflectă consumul de combustibil al utilizatorilor.

### Cerință

Se cere alegerea amplasamentelor tunelurilor și viaductelor pentru a minimiza costul total.

### Date de intrare

Fișierul **sosea.in** are următorul format:

- pe prima linie, trei numere  $N$   $T$   $P$ , reprezentând lungimea totală a șoselei, costul unei unități de lungime de tunel sau viaduct și, respectiv, costul unei unități de lungime de șosea în pantă (urcare sau coborâre)
- pe a doua linie, un șir de  $N$  litere din mulțimea  $\{U, H, C\}$ , fără spații între ele, reprezentând faptul că, pe porțiunea respectivă terenul urcă, este orizontal sau, respectiv coboară.

### Date de ieșire

În fișierul **sosea.out** se va scrie:

- pe prima linie, se vor scrie două numere,  $M$  și  $C$ , unde  $M$  este numărul total de tuneluri și viaducte (numărate împreună), iar  $C$  este costul total
- pe fiecare dintre următoarele  $M$  linii, câte două numere reprezentând distanța față de începutul drumului a punctului în care începe tunelul sau viaductul, respectiv a punctului unde se termină. Liniile trebuie să fie ordonate crescător după distanța față de început.

### Restricție

- $2 \leq N \leq 1000000$

## Exemplu

**sosea.in**                      **sosea.out**

**11 2 3**                         **2 18**

**UUUCCUHHHCU**               **1 5**

**9 11**

Timp maxim de execuție/test: **2** secunde.

Total memorie disponibilă: **30** MB.

Dimensiunea maximă a sursei: **5** KB.

SOCIETATEA DE ȘTIINȚE MATEMATICE DIN ROMÂNIA - FILIALA CLUJ



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XV-a, 8–9 mai 2015



## Clasele XI-XII

### Problema 2 – RESURSE

Satelitul S5102 a finalizat scanarea planetei P2015 în vederea descoperirii eventualelor resurse existente pe planetă. Din considerente tehnice, cercetătorii au programat satelitul să scaneze suprafața planetei pe zone de formă dreptunghiulară. La fiecare astfel de scanare se caută un anumit tip de resursă, iar, dacă rezultatul scanării este pozitiv, acesta se reține prin înregistrarea coordonatelor zonei scanare și a tipului de resursă descoperită.

#### Cerință

Cunoscându-se numărul de scanări pozitive efectuate și rezultatele acestora, să se determine resursele care se găsesc în cantitate maximă și aria suprafeței care conține resurse.

#### Date de intrare

Fișierul de intrare **resurse.in** conține pe prima linie un număr natural **p**, pentru toate testele de intrare, numărul **p** putând avea doar valoarea **1** sau valoarea **2**.

Pe cea de-a doua linie a fișierului se găsește un număr natural **n** reprezentând numărul de scanări pozitive efectuate.

Pe fiecare dintre următoarele **n** linii ale fișierului se găsesc, separate printr-un spațiu, câte cinci numere naturale **a b c d k** reprezentând coordonatele (**a,b**) și coordonatele (**c,d**) ale două colțuri opuse ale zonei scanate și tipul **k** de resursă descoperită.

#### Date de ieșire

Fișierul de ieșire **resurse.out** va conține o singură linie:

- dacă **p** este egal cu **1** se va scrie numărul de resurse care se găsesc în cantitate maximă, valoare urmată de tipurile de resurse corespunzătoare, toate valorile succesive scrise fiind separate printr-un spațiu și afișate în ordine crescătoare
- dacă **p** este egal cu **2** se va scrie aria suprafeței care conține resurse

#### Restricții și precizări

- $1 \leq n \leq 300$
- $0 \leq a < c \leq 1000000$
- $0 \leq b < d \leq 1000000$
- $1 \leq k \leq 1000000$
- Fiecare zonă scanată este de formă dreptunghiulară și are laturile paralele cu axele.
- O anumită zonă poate apărea în rezultatele mai multor scanări.
- O resursă poate fi descoperită în mai multe zone.
- Fiecare resursă descoperită într-o anumită zonă scanată se găsește într-o cantitate direct proporțională cu suprafața zonei scanate.
- Aria suprafeței care conține resurse este un număr care nu depășește valoarea  $10^{10}$ .
- Prima cerință reprezintă 50% din punctajul total, iar cea de-a doua cerință reprezintă 50% din punctajul total.



## Exemple

resurse.in

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

resurse.out

```
1 2
```

### *Explicație*

Există o singură resursă în cantitate maximă:  
resursa 2.

resurse.in

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

resurse.out

```
10
```

Timp maxim de execuție/test: **0.5** secunde.

Total memorie disponibilă: **4** MB.

Dimensiunea maximă a sursei: **5** KB.