



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ  
ȘI INFORMATICĂ „MARIAN ȚARINĂ”  
Ediția a XVI-a, 25–26 MARTIE 2016



Clasa a IX-a

Problema 1 – teren

100p

Gigel dorește să-și facă un teren de fotbal în curtea casei. În curte există pomi, însă terenul de fotbal nu este permis să aibă pomi în interiorul său. În plus, laturile terenului trebuie să fie paralele cu laturile curții. Ajutați-l pe Gigel să găsească o zonă cu suprafața cât mai mare pe care s-o transforme în teren de fotbal.

**Cerință**

Cunoscându-se dimensiunea curții și numărul de pomi din curte, să se determine coordonatele unei zone cu suprafață maximă pentru terenul de fotbal.

**Date de intrare**

Fișierul de intrare **teren.in** conține:

- pe prima linie, trei numere  $N$ ,  $X$ ,  $Y$ , reprezentând numărul de pomi, lățimea și, respectiv, lungimea curții;
- pe fiecare dintre următoarele  $N$  linii, câte două numere  $X_i$ ,  $Y_i$ , reprezentând coordonatele câte unui pom, raportate la un colț al curții.

**Date de ieșire**

În fișierul **teren.out** se vor scrie, pe o singură linie, patru numere,  $X_a$ ,  $Y_a$ ,  $X_b$ ,  $Y_b$ , reprezentând coordonatele a două colțuri opuse ale terenului de fotbal.

**Restricții și precizări**

- $0 \leq N \leq 1000$
- $0 < X, Y \leq 1000000$
- $0 \leq X_i \leq X$ ;  $0 \leq Y_i \leq Y$
- Terenul găsit este permis să aibă pomi pe margine sau să fie lipit de gardul curții.

**Exemplu**

<b>teren.in</b>	<b>teren.out</b>
2 10 10	0 0 5 10
5 5	
5 8	

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 2 MB.

Dimensiunea maximă a sursei: 5 KB.

CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ  
ȘI INFORMATICĂ „MARIAN ȚARINĂ”  
Ediția a XVI-a, 25–26 MARTIE 2016

Clasa a IX-a

Problema 2 – pătrate

100p

O foaie de formă dreptunghiulară cu dimensiunile  $a \times b$  este împărțită în pătrățele identice cu laturile paralele cu laturile foii. Numărul de pătrățele de pe această foaie este  $a \times b$ . Unele dintre aceste pătrățele sunt colorate.

## Cerință

Cunoscându-se dimensiunile foii și culorile pătrățelelor, să se determine:

1. numărul de culori utilizate la colorarea aceluiași număr maxim de pătrățele;
2. latura maximă a unei zone pătrate formată doar din pătrățele necolorate;
3. numărul de zone pătrate formate doar din pătrățele colorate în același mod.

## Date de intrare

Fișierul de intrare `patrate.in` conține pe prima linie un număr natural  $p$ , pentru toate testele de intrare, numărul  $p$  putând avea doar valoarea 1, valoarea 2 sau valoarea 3.

Pe cea de-a doua linie a fișierului se găsesc, separate printr-un spațiu, două numere naturale  $a$  și  $b$ , reprezentând dimensiunile foii.

Pe fiecare dintre următoarele  $a$  linii ale fișierului se găsesc, separate prin câte un spațiu, câte  $b$  numere naturale, reprezentând codurile culorilor corespunzătoare pătrățelelor.

## Date de ieșire

Fișierul de ieșire `patrate.out` va conține o singură linie pe care va fi scrisă o singură valoare reprezentând:

- rezultatul primei cerințe dacă valoarea lui  $p$  este 1 (numărul de culori utilizate la colorarea aceluiași număr maxim de pătrățele) sau
- rezultatul celei de-a doua cerințe dacă valoarea lui  $p$  este 2 (latura maximă a unei zone pătrate formată doar din pătrățele necolorate) sau
- rezultatul celei de-a treia cerințe dacă valoarea lui  $p$  este 3 (numărul de zone pătrate formate doar din pătrățele colorate în același mod).

## Restricții și precizări

- $1 \leq a, b \leq 1000$
- Codurile culorilor sunt numere naturale nenule cu maxim 3 cifre.
- Codul 0 este codul asociat unui pătrățel necolorat.
- Prima cerință reprezintă 20% din punctajul total, iar fiecare dintre celelalte două cerințe reprezintă 40% din punctajul total.

## Exemple

<code>patrate.in</code>	<code>patrate.out</code>	Explicație
1	2	Culorile cu codurile 2 și 3 sunt cele mai utilizate în colorarea pătrățelelor.
5 6		
0 2 2 4 4 0		
2 2 2 3 3 4		
2 2 2 3 3 4		
1 2 2 3 3 3		
1 1 1 3 3 3		

```
patrate.in
2
5 6
0 1 7 0 0 0
0 1 2 6 0 0
0 3 4 2 0 0
0 0 0 0 0 0
2 0 0 0 1 2
```

```
patrate.out
2
```

### *Explicație*

Latura maximă a unei zone pătrate formată doar din pătrățele necolorate este 2.

```
patrate.in
3
5 6
1 1 1 1 1 1
1 1 2 1 1 1
0 1 1 1 1 0
1 1 3 3 1 1
1 1 3 3 1 1
```

```
patrate.out
35
```

### *Explicație*

Sunt 28 de zone pătrate cu latura 1 și 7 zone pătrate cu latura 2.

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

Total memorie disponibilă: **10** MB.

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



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ  
ȘI INFORMATICĂ „MARIAN ȚARINĂ”  
Ediția a XVI-a, 25–26 MARTIE 2016



Clasa a X-a

100p

Problema 1 – cuvinte

Limba extraterestrilor de pe planeta **adruT 6102** constă din cuvinte alcătuite după următoarea regulă:

- fiecare cuvânt este alcătuit din exact  $S$  silabe;
- fiecare silabă constă fie dintr-o consoană urmată de o vocală, fie dintr-o vocală urmată de o consoană.

În plus, cercetătorii au descoperit care este ordinea literelor în alfabet, precum și, despre fiecare literă dacă este vocală sau consoană.

Toate șirurile de litere formate după regulile de mai sus sunt cuvinte valide.

**Cerintă**

Dându-se un cuvânt, se cere să se determine numărul său de ordine între cuvintele valide (în ordine lexicografică).

**Date de intrare**

Fișierul **cuvinte.in** conține pe prima linie, două numere,  $N$  și  $S$ , reprezentând numărul de litere din alfabet și numărul de silabe din cuvintele valide.

Pe a doua linie,  $N$  numere, fiecare descriind o literă din alfabet; numărul este 0 dacă litera corespunzătoare este vocală sau 1 dacă e consoană.

Pe a treia linie,  $2 \cdot S$  numere cuprinse între 0 și  $N-1$  fiecare, reprezentând literele succesive ale unui cuvânt valid.

**Date de ieșire**

În fișierul **cuvinte.out** se va scrie, pe o singură linie, numărul de ordine al cuvântului dat la intrare modulo 54321.

**Restricții și precizări**

- $1 \leq N < 65535$
- $1 \leq S \leq 200000$
- Literele alfabetului sunt numerotate de la 0 la  $N-1$ .
- Cuvintele sunt numerotate de la 0.

**Exemplu**

<b>cuvinte.in</b>	<b>cuvinte.out</b>	<b>Explicație</b>
3 2	12	Cuvintele posibile sunt:
1 1 0		0: 0 2 0 2      6: 1 2 2 0      12: 2 1 0 2
2 1 0 2		1: 0 2 1 2      7: 1 2 2 1      13: 2 1 1 2
		2: 0 2 2 0      8: 2 0 0 2      14: 2 1 2 0
		3: 0 2 2 1      9: 2 0 1 2      15: 2 1 2 1
		4: 1 2 0 2      10: 2 0 2 0
		5: 1 2 1 2      11: 2 0 2 1

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 4 MB.

Dimensiunea maximă a sursei: 5 KB.

CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ  
ȘI INFORMATICĂ „MARIAN ȚARINĂ”  
Ediția a XVI-a, 25–26 MARTIE 2016

Clasa a X-a

Problema 2 – intervale

100p

Noua preocupare a lui Gigel este prelucrarea intervalelor. Astfel, el încearcă, plecând de la mai multe intervale închise fixate, să afle câte numere întregi aparțin unui număr maxim de intervale și numărul de zerouri cu care se termină produsul tuturor numerelor întregi care aparțin cel puțin unuia dintre aceste intervale. Gigel vrea să verifice dacă numerele determinate de el sunt corecte și apelează la ajutorul vostru.

**Cerință**

Cunoscându-se numărul de intervale închise cu capete numere întregi și aceste capete, să se determine:

1. numărul de numere întregi care aparțin unui număr maxim de intervale;
2. numărul de zerouri cu care se termină produsul tuturor numerelor întregi care aparțin cel puțin unuia dintre intervalele date.

**Date de intrare**

Fișierul de intrare **intervale.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 intervale.

Pe fiecare dintre următoarele **n** linii ale fișierului se găsesc, separate prin câte un spațiu, câte două numere întregi **a b**, reprezentând capetele unui interval închis.

**Date de ieșire**

Fișierul de ieșire **intervale.out** va conține o singură linie pe care va fi scrisă o singură valoare reprezentând:

- rezultatul primei cerințe dacă valoarea lui **p** este **1** (numărul de numere întregi care aparțin unui număr maxim de intervale) sau
- rezultatul celei de-a doua cerințe dacă valoarea lui **p** este **2** (numărul de zerouri cu care se termină produsul tuturor numerelor întregi care aparțin cel puțin unuia dintre aceste intervale).

**Restricții și precizări**

- $1 \leq n \leq 100000$
- $1 \leq a \leq b \leq 1000000000$
- Fiecare dintre cele două cerințe reprezintă **50%** din punctajul total.
- Pentru **30%** din teste  $n \leq 1000$  și  $b-a \leq 1000$ .

**Exemple**

<b>intervale.in</b>	<b>intervale.out</b>	<b>Explicație</b>
1	2	Sunt 2 numere (5 și 6) care aparțin unui număr maxim de intervale (3).
4		
3 10		
2 6		
10 20		
5 7		

<b>intervale.in</b>	<b>intervale.out</b>	<b>Explicație</b>
2	4	Produsul numerelor care apar în cel puțin unul dintre intervale (numerele de la 2 la 20) se termină cu 4 zerouri.
4		
3 10		
2 6		
10 20		
5 7		

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

Total memorie disponibilă: **5** MB.

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



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ  
ȘI INFORMATICĂ „MARIAN ȚARINĂ”  
Ediția a XVI-a, 25–26 MARTIE 2016



Clasele XI-XII

Problema 1 – avioane

100p

Un turist dorește să ajungă într-un anumit oraș. El dispune de lista tuturor zborurilor care l-ar putea ajuta să ajungă la destinația sa, direct sau prin aeroporturi intermediare.

Întrucât turistul nostru dispune de timp, dar nu și de prea mulți bani, dorește varianta cea mai ieftină. La reducerea prețului ar putea contribui și ofertele promoționale ale companiilor aeriene: astfel, la 2 sau mai multe zboruri consecutive cu aceeași companie, se oferă 20% reducere pe fiecare dintre zborurile respective.

**Cerință**

Cunoscându-se datele despre toate zborurile posibile, aeroportul de plecare și aeroportul destinație, să se determine o secvență de zboruri cu cost total minim.

**Date de intrare**

Fișierul **avioane.in** va conține pe prima linie, 3 numere,  $N$ ,  $S$ ,  $D$ , reprezentând numărul de zboruri, aeroportul de plecare și aeroportul destinație.

Pe următoarele  $N$  linii, câte 6 numere,  $S_i$ ,  $TS_i$ ,  $D_i$ ,  $TD_i$ ,  $A_i$ ,  $C_i$  reprezentând aeroportul și ora plecării, aeroportul și ora sosirii, compania și prețul fără reducere al zborului.

**Date de ieșire**

Fișierul **avioane.out** va conține, pe o singură linie, numărul  $M$  de zboruri din soluția optimă, urmat de  $M$  numere între 0 și  $N-1$  reprezentând zborurile alese.

**Restricții și precizări**

- Dacă un zbor ajunge la un anumit timp pe un anumit aeroport, orice zbor care pleacă de pe acel aeroport la un timp mai mare sau egal cu acel timp poate fi utilizat pentru continuarea călătoriei.
- $1 \leq N \leq 1000$
- Aeroporturile și companiile sunt identificate prin numere nenegative mai mici de 1000000.
- Timpii sunt numere pozitive mai mici de 1000000.
- Costul fiecărui zbor este pozitiv, mai mic de 100000 și este divizibil cu 5.
- În 40% dintre teste, toate zborurile sunt efectuate de companii distincte.
- Se garantează că există cel puțin o soluție.

**Exemplu**

<b>avioane.in</b>	<b>avioane.out</b>	<b>Explicație</b>
5 0 1	2 0 3	Soluția costă $0.8 \cdot (100 + 300) = 320$ ambele zboruri fiind operate de aceeași companie.
0 0 2 10 1 100		Zborul direct (4) costă 350.
2 5 1 10 1 100		Zborul 0 urmat de 2 costă $100 + 280 = 380$ (fiind companii diferite, nu avem reducere).
2 15 1 20 2 280		Zborul 0 nu poate fi urmat de 1 deoarece zborul 1 pleacă înainte ca zborul 0 să aterizeze.
2 15 1 20 1 300		
0 0 1 10 1 350		

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 4 MB.

Dimensiunea maximă a sursei: 5 KB.

CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ  
ȘI INFORMATICĂ „MARIAN ȚARINĂ”  
Ediția a XVI-a, 25–26 MARTIE 2016

## Clasele XI-XII

## Problema 2 – camere

100p

Într-o piață de formă dreptunghiulară sunt amplasate la diverse înălțimi și în diverse poziții mai multe camere de supraveghere care monitorizează diverse obiective. Pentru protejarea acestor camere, firma care se ocupă cu întreținerea lor, dorește să monteze un paravan orizontal deasupra camerelor. De la un alt amplasament, firma are deja un paravan de formă dreptunghiulară, însă este dispusă să construiască unul nou de forma unui poligon convex cu condiția încadrării în limita unei sume care depinde direct proporțional de suprafața paravanului. Un paravan poate fi amplasat la o anumită înălțime doar dacă protejează toate camerele care se găsesc la o înălțime cel mult egală cu cea de amplasare a paravanului.

## Cerință

Cunoscându-se coordonatele punctelor în care sunt amplasate camerele de supraveghere, dimensiunile paravanului existent și suprafața maximă a paravanului care poate fi construit, să se determine:

1. numărul maxim de camere amplasate la aceeași înălțime;
2. numărul maxim de camere care pot fi protejate prin amplasarea paravanului existent, știind că acesta va fi amplasat cu laturile paralele cu laturile pieței;
3. numărul maxim de camere care ar putea fi protejate prin amplasarea paravanului nou cu suprafața maximă dată.

## Date de intrare

Fișierul de intrare **camere.in** conține pe prima linie un număr natural  $p$ , pentru toate testele de intrare, numărul  $p$  putând avea doar valoarea 1, valoarea 2 sau valoarea 3.

Pe cea de-a doua linie a fișierului se găsesc, separate prin câte un spațiu, patru numere naturale  $n a b s$ , reprezentând, în ordine, numărul de camere, dimensiunile paravanului existent și suprafața maximă a paravanului care poate fi construit.

Pe fiecare dintre următoarele  $n$  linii ale fișierului se găsesc, separate prin câte un spațiu, câte trei numere naturale  $x y h$ , reprezentând coordonatele  $(x, y)$  în raport cu unul dintre colțurile pieței, respectiv, înălțimea la care este amplasată o cameră.

## Date de ieșire

Fișierul de ieșire **camere.out** va conține o singură linie pe care va fi scrisă o singură valoare reprezentând:

- rezultatul primei cerințe dacă valoarea lui  $p$  este 1 (numărul maxim de camere amplasate la aceeași înălțime) sau
- rezultatul celei de-a doua cerințe dacă valoarea lui  $p$  este 2 (numărul maxim de camere care pot fi protejate prin amplasarea paravanului existent) sau
- rezultatul celei de-a treia cerințe dacă valoarea lui  $p$  este 3 (numărul maxim de camere care ar putea fi protejate prin amplasarea paravanului nou).

## Restricții și precizări

- $1 \leq n \leq 100000$
- $1 \leq a, b, x, y, h \leq 1000$
- $1 \leq s \leq 1000000$
- Se consideră că dimensiunile unei camere de supraveghere sunt neglijabile.



- O cameră de supraveghere este protejată de un paravan dacă proiecția poziției camerei pe suprafața pieței se găsește în interiorul, pe una dintre laturile sau într-unul dintre colțurile poligonului determinat de proiecția paravanului pe suprafața pieței și înălțimea la care este amplasată camera este cel mult egală cu cea la care este amplasat paravanul.
- Dacă proiecția paravanului pe suprafața pieței este un segment sau un punct, se consideră că suprafața lui este 0.
- Fiecare dintre primele două cerințe reprezintă 20% din punctajul total, iar cea de-a treia cerință reprezintă 60% din punctajul total.

### Exemple

<b>camere.in</b>	<b>camere.out</b>	<b>Explicație</b>
1 7 3 3 6 2 3 4 1 2 5 3 1 6 2 2 2 3 4 4 5 1 3 2 1 5	2	Numărul maxim de camere de supraveghere amplasate la aceeași înălțime este 2 (camera 1 și camera 5).
<b>camere.in</b>	<b>camere.out</b>	<b>Explicație</b>
2 7 3 3 6 2 3 4 1 2 5 3 1 6 2 2 2 3 4 4 5 1 3 2 1 5	4	Prin amplasarea paravanului existent la înălțimea 4 sunt protejate 4 camere (cele cu numerele de ordine 1, 4, 5 și 6).
<b>camere.in</b>	<b>camere.out</b>	<b>Explicație</b>
3 7 3 3 6 2 3 4 1 2 5 3 1 6 2 2 2 3 4 4 5 1 3 2 1 5	4	Prin amplasarea unui paravan nou cu o suprafață minimă de 4 la înălțimea 4 ar fi protejate 4 camere (cele cu numerele de ordine 1, 4, 5 și 6). Pentru protejarea tuturor camerelor, suprafața minimă a noului paravan este de 6.50.

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 5 MB.

Dimensiunea maximă a sursei: 10 KB.