



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚĂRINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Radu Lupșa

**Clasa a IX-a**

**Problema 1 – planete**

**100p**

Astronomii au descoperit că steaua *cmt2018* are un sistem de planete ce se rotesc în jurul ei. Ei pot determina momentele în care cel puțin una dintre planete trece prin fața stelei (fenomenul se numește *tranzit*). De notat că ei pot determina doar că la un anumit moment s-a produs un tranzit, dar nu pot determina care sau câte planete au participat la acel tranzit.

În urma măsurărilor, avem o astfel de secvență de momente de tranzit. În plus, mai știm că:

- la momentul zero, toate planetele tranzitează simultan prin fața stelei;
- durata secvenței este mai mare decât cea mai lungă dintre perioadele mișcărilor de revoluție ale planetelor în jurul stelei (altfel spus, pe durata măsurărilor, fiecare planetă tranzitează cel puțin de două ori);
- între momentul primului tranzit din secvență și momentul ultimului, nu există nici un tranzit care să nu fi fost observat (toate tranziturile fac parte din secvență).

**Cerință**

Să se afle numărul minim de planete ce explică tranziturile observate, precum și perioadele mișcărilor de revoluție ale acestora.

**Date de intrare**

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

- pe prima linie, numărul  $L$  de momente în care cel puțin o planetă tranzitează prin fața stelei;
- pe a doua linie,  $L$  numere reprezentând momentele tranziturilor. Numerele sunt ordonate crescător, iar primul număr este 0.

**Date de ieșire**

În fișierul de ieșire **planete.out** se va scrie una dintre posibilitățile sistemului planetar:

- pe prima linie se va scrie numărul  $N$  de planete;
- pe a doua linie se vor scrie  $N$  numere reprezentând perioadele mișcărilor de revoluție ale planetelor.

**Restricții și precizări**

- $2 \leq L \leq 1000000$
- Toți timpii sunt mai mici decât  $2^{63}$ .
- Se garantează că există cel puțin o soluție.
- Se acordă 10 puncte din oficiu.

**Exemplu**

<b>planete.in</b>	<b>planete.out</b>
6	2
0 4 8 10 12 16 20	4 10

Timp maxim de execuție/test: 2 secunde.

Total memorie disponibilă: 10 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚARINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Florin Miron

**Clasa a IX-a**

**Problema 2 – distincte**

**100p**

Gigel este preocupat, mai nou, de consultarea șirurilor formate din numere naturale. Mai exact, el analizează diversele secvențe ale unui șir și încearcă să afle câte dintre acestea sunt formate din elemente distincte. Se știe că o secvență a unui șir este formată din elemente ale acestuia aflate pe poziții alăturate.

**Cerință**

Cunoscându-se numărul de elemente ale șirului și elementele acestuia, să se determine numărul de secvențe din șir care sunt formate din valori distincte.

**Date de intrare**

Fișierul de intrare **distincte.in** conține pe prima linie un număr natural  $n$ , reprezentând numărul de elemente ale șirului.

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

**Date de ieșire**

Fișierul de ieșire **distincte.out** va conține o singură linie pe care va fi scrisă o singură valoare, reprezentând numărul de secvențe ale șirului formate din valori distincte.

**Restricții și precizări**

- $1 \leq n \leq 100000$
- Fiecare element al șirului este un număr natural cu cel mult 6 cifre.
- O secvență poate fi formată dintr-un singur element.
- Se acordă 10 puncte din oficiu.

**Exemplu**

<b>distincte.in</b>	<b>distincte.out</b>	<b>Explicație</b>
5 1 2 1 3 2	11	Sunt 11 secvențe formate din valori distincte: (1) (1 2) (2) (2 1) (2 1 3) (1) (1 3) (1 3 2) (3) (3 2) (2).

Timpi maxim de execuție/test: 0.2 secunde.

Total memorie disponibilă: 5 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚĂRINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Radu Lupșa

**Clasa a IX-a**

**Problema 3 – suma**

**100p**

Costel este pasionat de șiruri de numere. Ori de câte ori găsește undeva un șir de numere, începe să calculeze sumele numerelor din subsecvențe ale șirului.

**Cerință**

Dându-se un șir de numere întregi (pozitive și negative), se cere să se determine subsecvența de sumă maximă (o subsecvență este un șir format din numere aflate pe poziții consecutive în șirul dat).

**Date de intrare**

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

- pe prima linie, numărul  $N$  de numere din șirul de intrare;
- pe a doua linie,  $N$  numere întregi, reprezentând șirul propriu-zis.

**Date de ieșire**

În fișierul de ieșire **suma.out** se vor scrie două numere naturale reprezentând indecșii primului și respectiv ultimului element din subsecvența de sumă maximă. Indexarea se consideră de la 1.

**Restricții și precizări**

- $1 \leq N \leq 1000000$
- Numerele din șir sunt mai mici în, valoare absolută, decât  $2^{31}$ .
- Se garantează că există cel puțin un număr strict pozitiv în șirul dat.
- Dacă există mai multe subsecvențe care produc aceeași sumă maximă, se va scrie oricare dintre ele.
- Se acordă 10 puncte din oficiu.

**Exemplu**

<b>suma.in</b>	<b>suma.out</b>	<b>Explicație</b>
8	3 7	Subsecvența de sumă maximă este
2 -3 10 -5 0 12 2 -3	10 -5 0 12 2	

Timp maxim de execuție/test: 2 secunde.

Total memorie disponibilă: 10 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚĂRINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Radu Lupșa

Clasa a X-a

Problema 1 – dicționar

100p

O echipă de cercetători au descoperit niște mesaje de origine extraterestră. Pentru a înțelege mesajele, ei au început să studieze cuvintele limbii extraterestre. Astfel, ei au constatat că toate cuvintele limbii extraterestrelor sunt formate din același număr  $K$  de silabe, iar fiecare silabă este formată dintr-o singură vocală, urmată de zero, una sau două consoane. Ei au numerotat vocalele de la 0 la  $V-1$  și consoanele de la  $V$  la  $V+C-1$ .

**Cerință**

Dându-se un cuvânt extraterestru, se cere să se afle numărul său de ordine (numerotat de la 0) în lista alfabetică a tuturor cuvintelor extraterestre.

**Date de intrare**

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

- pe prima linie cele trei numere  $K$ ,  $V$  și  $C$ ;
- pe a doua linie, lungimea  $N$  a unui cuvânt extraterestru;
- pe cea de-a treia linie,  $N$  numere reprezentând literele cuvântului extraterestru.

**Date de ieșire**

În fișierul de ieșire **dicționar.out** se va scrie, pe o singură linie, numărul de ordine al cuvântului de la intrare, modulo 1000001.

**Restricții și precizări**

- $1 \leq K, V, C \leq 50$
- Se acordă 10 puncte din oficiu.

**Exemplu**

<b>dicționar.in</b>	<b>dicționar.out</b>	<b>Explicație</b>
1 2 2 3 1 2 2	9	Cuvintele extraterestre sunt, în ordine: (0), (0 2), (0 2 2), (0 2 3), (0 3), (0 3 2), (0 3 3), (1), (1 2), (1 2 2), (1 2 3), (1 3), (1 3 2), (1 3 3)

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 5 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚĂRINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Florin Miron

Clasa a X-a

Problema 2 – imagine

100p

Gigel a descoperit o nouă aplicație informatică care îi permite să creeze diverse fișiere de tip imagine. Fiind la primele încercări de utilizare a aplicației, el se axează pe exersarea unor operații elementare. Astfel, la deschiderea unui fișier nou, el precizează în număr de pixeli dimensiunile imaginii de formă dreptunghiulară cu care va lucra. Apoi, aplică succesiv operații care implică selectarea unei zone de formă dreptunghiulară și cu laturile paralele cu marginile imaginii și colorarea uniformă a acestei zone.

Cerință

Cunoscându-se dimensiunile imaginii, numărul de zone care vor fi colorate, pozițiile acestora în cadrul imaginii și codurile de culoare corespunzătoare, să se determine numărul de pixeli care rămân necolorați și care sunt codurile de culoare cele mai utilizate în imaginea rezultată după colorările succesiv realizate.

Date de intrare

Fișierul de intrare **image.in** conține pe prima linie un număr **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 printr-un spațiu, două numere naturale **n** și **m**, reprezentând dimensiunile imaginii exprimate în număr de pixeli pe verticală și pe orizontală.

Pe cea de-a treia linie a fișierului se găsește un număr natural **nr**, reprezentând numărul de zone care vor fi colorate.

Pe fiecare dintre următoarele **nr** linii ale fișierului se găsesc, separate prin câte un spațiu, câte cinci numere naturale **a b c d e**, care descriu fiecare operație de colorare: perechea **a b** reprezintă coordonatele verticală/orizontală ale pixelului din colțul stânga sus al zonei, perechea **c d** reprezintă coordonatele verticală/orizontală ale pixelului din colțul dreapta jos al zonei, iar **e** reprezintă codul de culoare utilizată pentru colorarea zonei.

Date de ieșire

Fișierul de ieșire **image.out** va conține o singură linie pe care vor fi scrise următoarele valori:

- rezultatul primei cerințe dacă valoarea lui **p** este **1** (numărul de pixeli care rămân necolorați);
- rezultatul celei de-a doua cerințe dacă valoarea lui **p** este **2** (codurile de culoare cele mai utilizate în imaginea finală); dacă sunt mai multe coduri de culoare cu această proprietate, acestea vor fi scrise în ordine crescătoare, separate prin câte un spațiu.

Restricții și precizări

- $1 \leq n, m \leq 1000000$
- $1 \leq nr \leq 500$
- $1 \leq e \leq 1000$
- $1 \leq a \leq c \leq n$
- $1 \leq b \leq d \leq m$
- Colorarea unei zone implică colorarea fiecărui pixel care se găsește în zona respectivă, indiferent dacă acesta era colorat anterior sau nu.
- Se consideră că toți pixelii sunt inițial necolorați și au codul de culoare **0**.
- Coordonatele unui pixel sunt date în raport cu colțul stânga sus al imaginii.
- Coordonatele pixelului stânga sus sunt (1,1).
- Fiecare dintre cele două cerințe reprezintă **50%** din punctajul total.
- Se acordă **10** puncte din oficiu.

## Exemple

`image.in`

```
1
5 5
3
1 1 2 2 1
2 2 3 3 2
4 4 5 5 3
```

`image.out`

```
14
```

### *Explicație*

În final rămân 14 pixeli necolorați.

`image.in`

```
2
5 5
3
1 1 2 2 1
2 2 3 3 2
4 4 5 5 3
```

`image.out`

```
2 3
```

### *Explicație*

În final sunt 4 pixeli colorați cu culoarea 2 și 4 pixeli colorați cu culoarea 3. Sunt doar 3 pixeli colorați cu culoarea 1.

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 5 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚĂRINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Gabriel Mircea

**Clasa a X-a**

**Problema 3 – vacanta**

**100p**

Un site de turism dorește să își premieze cei mai fideli clienți cu o vacanță de Paști în doi. Se dorește ca fericiții câștigători să fie cât mai apropiați ca vârstă și ca putere de cumpărare pentru a se reduce eventualele incompatibilități. Site-ul dispune de toate informațiile legate de vârsta  $V_i$  și puterea de cumpărare  $P_i$  a fiecărui client  $i$ . Specialiștii în marketing ai firmei au realizat un grafic prin care reprezintă relațiile de apropiere prin intermediul distanței euclidiene dintre doi clienți într-un reper cartezian în care abscisa definește vârsta iar ordonata, puterea de cumpărare. Se dorește realizarea unui algoritm care, folosind reprezentarea oferită de specialiști, să optimizeze găsirea celor mai compatibili doi clienți care să câștige vacanța de Paști în doi promisă.

**Intrarea**

Fișierul **vacanta.in** conține

- pe prima linie, numărul  $L$  definind numărul de clienți ai firmei;
- pe următoarele  $L$  linii:  $V_i P_i$  două numere întregi pozitive separate prin spațiu care definesc respectiv vârsta și puterea sa de cumpărare.

**Ieșirea**

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

- pe prima linie valorile pozițiilor din lista clienților ale celor mai compatibili doi clienți separate printr-un spațiu.

**Restricții și precizări**

- $1 \leq n \leq 100000$
- $1 \leq V_i \leq 100000000$
- $1 \leq P_i \leq 100000000$
- Indexarea se consideră de la 1.
- Se acordă 10 puncte din oficiu.

**Exemplu**

<b>vacanta.in</b>	<b>vacanta.out</b>	<b>Explicație</b>
10	5 8	Clienții cu numerele de ordine 5 și 8 sunt cei mai apropiați ca distanță euclidiană în planul cartezian propus de specialiști.
26 77		
12 37		
14 18		
19 96		
71 95		
91 9		
98 43		
66 77		
2 75		
94 91		

Timp maxim de execuție/test: 0.4 secunde.

Total memorie disponibilă: 5 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚĂRINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Radu Lupșa

**Clasele XI-XII**

**Problema 1 – robot**

**100p**

Un robot se deplasează printre niște obstacole. Robotul are forma unui pătrat de dimensiune impară. Obstacolele sunt pătrate de dimensiune 1. Robotul se poate deplasa doar paralel cu una sau cealaltă dintre axele de coordonate.

**Cerință**

Găsiți traseul cel mai scurt pe care trebuie să-l parcurgă robotul pentru a ajunge din poziția sa inițială în poziția finală dorită.

**Date de intrare**

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

- pe prima linie, șase numere întregi,  $N$   $P$   $X_s$   $Y_s$   $X_d$   $Y_d$ , reprezentând numărul de obstacole, dimensiunea robotului, coordonatele punctului de plecare și coordonatele punctului destinație;
- pe următoarele  $N$  linii, câte două numere  $X_k$   $Y_k$ , reprezentând coordonatele câte unui obstacol.

**Date de ieșire**

În fișierul de ieșire **robot.out** se va scrie, pe o singură linie, secvența de mutări care duce robotul de la sursă la destinație, folosind un număr minim de mutări. Secvența se va scrie ca un șir de caractere format din litere  $N$   $S$   $E$  sau  $V$ . Mutările reprezintă:  $N$  – coordonata  $Y$  crește cu o unitate;  $S$  -  $Y$  scade cu o unitate;  $E$  -  $X$  crește cu o unitate,  $V$  -  $X$  scade cu o unitate.

**Restricții și precizări**

- Toate coordonatele reprezintă centrele pătratelor (robotului sau unui obstacol).
- Toate coordonatele sunt cuprinse între -500 și 500.
- $0 \leq N \leq 10000$
- $P$  e impar și  $0 \leq P \leq 999$ .
- Se garantează existența unei soluții.
- Se acordă 10 puncte din oficiu.

**Exemplu**

<b>robot.in</b>	<b>robot.out</b>
2 3 -1 0 4 1	NNNEEEEEESS
1 1	
2 -2	

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 64 MB.

Dimensiunea maximă a sursei: 5 KB.





CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚĂRINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Florin Miron

Clasele XI-XII

Problema 2 – echilibru

100p

Un taler de forma unui disc omogen se sprijină în centrul său pe o tijă verticală și se găsește inițial în echilibru. Se dorește amplasarea unor obiecte de același tip în puncte de coordonate cunoscute în raport cu centrul talerului, astfel încât talerul să rămână în echilibru.

Cerință

Cunoscându-se numărul de puncte în care pot fi amplasate obiectele și coordonatele acestora, să se determine număr de variante distincte de amplasare a obiectelor modulo 10007, astfel încât talerul să fie în echilibru.

Date de intrare

Fișierul de intrare **echilibru.in** conține pe prima linie un număr natural  $n$ , reprezentând numărul de puncte în care pot fi amplasate obiecte.

Pe fiecare dintre următoarele  $n$  linii ale fișierului se găsesc, separate printr-un spațiu, câte două numere întregi  $x$  și  $y$ , reprezentând coordonatele unui punct în care poate fi amplasat un obiect.

Date de ieșire

Fișierul de ieșire **echilibru.out** va conține o singură linie pe care va fi scris numărul de variante distincte de amplasare a obiectelor modulo 10007, astfel încât talerul să fie în echilibru.

Restricții și precizări

- $1 \leq n \leq 50$
- $-10 \leq x, y \leq 10$
- Toate punctele în care pot fi amplasate obiectele sunt distincte între ele.
- Într-un punct poate fi amplasat cel mult un obiect.
- Două variante de amplasare a obiectelor se consideră distincte dacă diferă fie prin numărul de obiecte amplasate, fie prin punctele în care s-au amplasat obiecte.
- Trebuie amplasat pe taler cel puțin un obiect.
- Numărul de obiecte avute la dispoziție coincide cu numărul de puncte în care pot fi amplasate obiecte.
- În 30% dintre teste  $n \leq 25$ .
- Se acordă 10 puncte din oficiu.

Exemplu

<b>echilibru.in</b>	<b>echilibru.out</b>	<b>Explicație</b>
5	3	Sunt 3 variante de dispunere a obiectelor.
1 2		1. (1 2) (-1 -1) (0 -1)
-2 0		2. (-2 0) (2 0)
-1 -1		3. (1 2) (-2 0) (-1 -1) (0 -1) (2 0)
0 -1		
2 0		

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 5 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ  
„MARIAN ȚĂRINĂ”  
Ediția a XVIII-a, 30-31 MARTIE 2018

Gabriel Mircea

**Clasele XI-XII**

**Problema 3 – artist**

**100p**

Un artist abstract are un mod foarte pragmatic prin care își realizează și evaluează material lucrările. Plecând de la o pânză imensă dreptunghiulară albă de lungime  $L$  el realizează dreptunghiuri de înălțime egală cu dimensiunea pânzei de anumite culori (fiecare culoare are asociată o valoare numerică) care pot să se suprapună.

Arta lui abstractă constă în crearea unor zone de suprapunere a două dreptunghiuri colorate astfel: atunci când un dreptunghi se suprapune cu un alt dreptunghi existent plecând de la poziția  $a$  la poziția  $b$  culoarea obținută în zonă se modifică conform formulei: pentru fiecare poziție  $k$  începând de la poziția  $a$  la poziția  $b$ , codul culorii de la poziția  $k$  ( $a \leq k \leq b$ ) va deveni **partea întreagă a mediei aritmetice** dintre *culoarea existentă la poziția  $k$  dacă aceasta nu e culoarea albă și culoarea care este aplicată peste zona care a fost deja colorată, iar dacă culoarea existentă este albă* codul culorii de la poziția  $k$  va lua doar **valoarea noii culori aplicate**.

La final el selectează un indice de start  $X$  și unul de stop  $Y$  pentru a tăia o anumită bucată îndeajuns de mică pentru a putea fi valorificată.

Pentru a-și putea recupera investiția, el vrea să calculeze **valoarea bucății obținute** ca fiind suma asignată culorilor fiecărei poziții întregi din intervalul  $[X, Y]$ .

**Intrarea**

Fișierul **artist.in** conține:

- pe prima linie, numărul întreg  $L$  care definește lungimea pânzei;
- pe a doua linie,  $N$  - numărul dreptunghiurilor pe care le desenează artistul;
- pe următoarele  $N$  linii:  $A_n$ ,  $B_n$ ,  $C_n$ , indicii stânga și dreapta ai dreptunghiurilor trasate de artist și codul  $C_n$  al culorii folosite;
- pe ultima linie:  $X$  și  $Y$ , indicii stânga și dreapta ai bucății dorite de artist.

**Ieșirea**

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

- pe o singură linie valoarea corespunzătoare bucății decupate de artist.

**Restricții și precizări**

- $1 \leq L \leq 100000$
- $1 \leq N \leq 100000$
- $1 \leq A_n \leq B_n \leq N$
- $1 \leq X \leq Y \leq N$
- $0 \leq C_n \leq 10000000$
- Se acordă **10** puncte din oficiu.

## Exemplu

artist.in

artist.out

### *Explicație*

Straturile aplicate de artist pot fi ilustrate ca mai jos  
elementele de pe pozițiile 5,6 vor fi înlocuite conform  
regulii cu  $(3+7)/2 = 5$  iar cea de pe poziția 8 cu  $(7+5)/2$   
 $=6$

10  
3  
3 6 3  
5 8 7  
8 9 5  
4 8

26

0 0 3 3 3 3 0 0 0 0

0 0 0 0 7 7 7 7 0 0

-----  
0 0 3 3 5 5 7 7 0 0

0 0 0 0 0 0 5 5 0

-----  
0 0 3 **3 5 5 7 6** 5 0

=> rezultatul este suma subșirului dorit (cu bold) **26**

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

Total memorie disponibilă: **5** MB.

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