

CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XVII-a, 7–8 aprilie 2017



**Clasa a IX-a**

**Problema 1 – teren**

**100p**

Un dezvoltator imobiliar are la dispoziție un teren, format din  $N$  fâșii dreptunghiulare late de 1m și de diferite lungimi. Toate fâșiile au una dintre laturile scurte lipite de stradă, strada fiind dreaptă și de  $N$  metri lungime. Pe latura lungă, fiecare fâșie este lipită de următoarea și de precedenta (dar, evident, doar pe partea comună a lungimii).

De asemenea, fiecare fâșie are o anumită valoare pe metru pătrat a terenului, cunoscută.

Identificați o parcelă dreptunghiulară, cu o latura lipită de stradă, și având valoarea cea mai mare posibilă.

**Date de intrare**

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

- pe prima linie, numărul  $N$  de fâșii;
- pe a doua linie,  $N$  numere pozitive reprezentând lungimile celor  $N$  fâșii;
- pe a treia linie,  $N$  numere pozitive reprezentând valoarea unui metru pătrat de teren din fiecare fâșie.

**Date de ieșire**

În fișierul de ieșire **teren.out** se va scrie, pe o singură linie, numărul de ordine a primei fâșii și numărul de ordine a ultimei fâșii care vor intra în compoziția parcelei. Fâșiile se consideră numerotate de la 0.

**Restricție**

- $1 \leq N \leq 100000$

**Exemplu**

**teren.in**

**teren.out**

*Explicație*

6  
10 2 6 5 1 10  
3 1 5 5 4 4

2 3

Terenul cuprinde fâșiile 2 și 3 și are adâncimea 5 de la stradă, astfel având o valoare de  $5*5+5*5=50$ .

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 16 MB.

Dimensiunea maximă a sursei: 5 KB.

CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XVII-a, 7–8 aprilie 2017



Clasa a IX-a

Problema 2 – roți

100p

Gigel a construit un angrenaj format din două roți dințate. El pune o picătură de ulei pe un dinte al primei roți, și apoi începe să rotească mecanismul. De fiecare dată când dintele uns ajunge în contact cu roata a doua, o parte din ulei este transferat în intervalul dintre cei doi dinți ai roții a doua între care intră dintele uns. Mai departe, de fiecare dată când un interval uns vine în contact cu un dinte al primei roți, dintele primei roți este uns, și așa mai departe. După o vreme însă, uleiul ajunge să se răspândească pe toți dinții, respectiv în toate intervalele, la care poate ajunge.

Se cere să se determine:

- câți dinți ai primei roți, respectiv intervale între dinții celei de-a doua roți, ajung să fie unse;
- după câtă vreme se ajunge ca toți dinții ce pot fi unși să fie unși.

Date de intrare

Fișierul de intrare `roti.in` conține, pe o singură linie, două numere pozitive separate printr-un spațiu, reprezentând numerele de dinți ale celor două roți.

Date de ieșire

În fișierul de ieșire `roti.out` se va scrie:

- pe prima linie, două numere separate printr-un spațiu, reprezentând numărul de dinți ai primei roți, și numărul de intervale ale celei de-a doua, care ajung să fie unse;
- pe a doua linie, un singur număr, reprezentând cu câți dinți trebuie să se rotească roțile din momentul în care dintele uns al primei roți ajunge prima dată în contact cu a doua roată și până în momentul în care toți dinții ce ajung să fie unși sunt efectiv unși.

Restricții

- Prima roată are cel mult 1000 de dinți.
- A doua roată are cel mult 1000 de intervale.

Exemple

<code>roti.in</code>	<code>roti.out</code>	Explicație
5 10	1 2 5	Numerotăm dinții primei roți, 0 reprezentând dintele uns, și intervalele celei de-a doua roți, 0 reprezentând primul interval uns. Ajunge uns doar dintele 0 și intervalele 0 și 5. Intervalul 5 ajunge uns la o rotație completă (așadar, 5 dinți) a primei roți.
<code>roti.in</code>	<code>roti.out</code>	Explicație
12 16	3 4 36	Cu aceeași numerotare, se ung: - la momentul 0, dintele 0 și spațiul 0 - la momentul 12, spațiul 12 - la momentul 16, dintele 4 - la momentul 24, spațiul 8 - la momentul 32, dintele 8 - la momentul 36, spațiul 4 După momentul 36, nu mai vine niciodată în contact un dinte uns cu un spațiu încă ne-uns, sau un spațiu uns cu un dinte ne-uns.

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 1 MB.

Dimensiunea maximă a sursei: 5 KB.

CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XVII-a, 7–8 aprilie 2017



**Clasa a X-a**

**Problema 1 – planete**

**100p**

Cercetătorii studiază nou-descoperitul sistem planetar exosolar Aadrut. Ei au determinat, pentru fiecare planetă, timpul necesar unei revoluții complete în jurul stelei centrale a sistemului. Ei doresc acum să studieze ce se întâmplă atunci când două sau mai multe planete se aliniază (adică steaua și două sau mai multe planete sunt coliniare).

Pentru studiul acesta, putem considera ca planetele urmează o orbită perfect circulară, având steaua în centru, și că toate orbitele se găsesc în același plan. De asemenea, durata unei revoluții complete în jurul stelei este un număr întreg de zile (terestre).

În momentul începerii studiului, toate planetele sunt aliniată. Cercetătorii doresc să afle:

- peste câtă vreme toate planetele se vor alinia din nou și în aceeași poziție (altfel spus, toate planetele vor reveni în pozițiile lor inițiale);
- până în acel moment, câte momente în care 2 sau mai multe planete sunt alineate vor exista.

**Date de intrare**

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

- pe prima linie, numărul  $N$  de planete;
- pe a doua linie,  $N$  numere pozitive reprezentând perioadele de revoluție ale planetelor.

**Date de ieșire**

În fișierul **planete.out** se va scrie, pe o singură linie, două numere: timpul între două alinieri ale planetelor în aceeași poziție și, respectiv, numărul de alinieri de 2 sau mai multe planete, incluzând alinierea inițială și excluzând alinierea finală.

**Restricții și precizări**

- $2 \leq N \leq 10$
- Se garantează că timpul între două alinieri pe direcția originală a tuturor planetelor este mai mic sau egal cu 2000000000 ( $2 \cdot 10^9$ ).

**Exemple**

**planete.in**                      **planete.out**

2                                      12 11  
1 12

**planete.in**                      **planete.out**

3                                      9 12  
1 3 9

**Explicație**

Prima și a doua planetă se aliniază de 6 ori, la momentele 0, 1.5, 3, 4.5, 6 și 7.5.

Prima și a treia planetă se aliniază de 8 ori, la momentele 0, 1.125, 2.25, 3.375, 4.5, 5.625, 6.75 și 7.875.

A doua și a treia planetă se aliniază de 2 ori, la momentele 0 și 4.5. Alinierile comune se numără o singură dată, astfel că sunt 12 momente distincte.

Timp maxim de execuție/test: **0.1** secunde.

Total memorie disponibilă: **1** MB.

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

CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XVII-a, 7–8 aprilie 2017



**Clasa a X-a**

**Problema 2 – obiective**

**100p**

Aflat într-una dintre desele lui excursii, Gigel a ajuns într-un nou oraș. Chiar la intrarea în acest oraș, Gigel se informează și află că, în mod ciudat, toate cele  $n$  obiective care pot fi vizitate din acest oraș sunt dispuse, succesiv, de-a lungul unui traseu liniar și sunt identificate, în mod unic, printr-un număr de ordine cuprins între  $1$  și  $n$ . Astfel, mergând pe jos, o persoană care se găsește la un obiectiv poate trece, în orice moment la obiectivul cu numărul de ordine strict mai mare (dacă există un astfel de obiectiv) sau la obiectivul cu numărul de ordine strict mai mic (dacă există un astfel de obiectiv). În plus, există și curse speciale care asigură transportul fără oprire într-un sens între două obiective. Fiecare trecere de la un obiectiv la altul, indiferent că deplasarea se face apelând la o cursă sau nu, se realizează într-o unitate de timp.

**Cerință**

Cunoscându-se numărul de obiective, numărul de curse și perechile de obiective între care există astfel de curse, să se determine timpul minim în care Gigel, plecând de la primul obiectiv, ajunge la ultimul obiectiv.

**Date de intrare**

Fișierul de intrare **obiective.in** conține pe prima linie, separate printr-un spațiu, două numere naturale  $n$  și  $m$ , reprezentând numărul obiectivelor și, respectiv, numărul curselor.

Pe următoarele  $m$  linii ale fișierului se găsesc, separate printr-un spațiu, câte două numere naturale distincte  $a$  și  $b$ , reprezentând numerele de ordine ale obiectivelor între care există o cursă.

**Date de ieșire**

Fișierul de ieșire **obiective.out** va conține o singură linie pe care va fi scris timpul minim de deplasare de la primul la ultimul obiectiv.

**Restricții și precizări**

- $1 \leq m \leq n \leq 100000$
- $1 \leq a, b \leq n$
- Nu există două curse care să lege aceleași obiective.
- Dintr-un obiectiv poate pleca cel mult o cursă.
- La un obiectiv pot ajunge mai multe curse.
- Timpul de vizitare al unui obiectiv din oraș nu este relevant pentru problemă.
- În 50% dintre teste, pentru fiecare cursă  $a < b$ .

**Exemple**

<b>obiective.in</b>	<b>obiective.out</b>	<b>Explicație</b>
7 2	4	Traseul care cuprinde obiectivele 1, 2, 3, 6 și 7 este cel mai rapid.
2 4		
3 6		

<b>obiective.in</b>	<b>obiective.out</b>	<b>Explicație</b>
9 4	3	Traseul care cuprinde obiectivele 1, 6, 4 și 9 este cel mai rapid.
4 9		
3 7		
1 6		
6 4		

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 XVII-a, 7–8 aprilie 2017



**Clasele XI-XII**

**Problema 1 – dale**

**100p**

Pentru construcția unei porțiuni de autostradă ce traversează un teren instabil, constructorul a decis așezarea unor dale de beton armat.

Constructorul dispune de câteva tipuri de dale; despre fiecare tip, cunoaștem lungimea dalei (în metri), rezistența mecanică și costul. De asemenea, constructorul dispune de un plan al terenului, în care traseul autostrăzii este împărțit în porțiuni de 1m lungime; despre fiecare astfel de porțiune, se cunoaște rezistența minimă a unei dale ce se poate așeza pe acea porțiune.

Se cere stabilirea unei înșiruii de dale care să acopere traseul autostrăzii și să aibă costul total minim.

**Date de intrare**

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

- pe prima linie două numere pozitive **L** și **N** reprezentând lungimea autostrăzii și numărul de tipuri de dale;
- pe a doua linie, **L** numere pozitive reprezentând rezistența minimă a dalei pentru porțiunea respectivă de teren de autostradă;
- pe următoarele **N** linii, câte **3** numere pozitive **L<sub>i</sub>** **R<sub>i</sub>** **C<sub>i</sub>** reprezentând lungimea dalei, rezistența și costul.

**Date de ieșire**

În fișierul de ieșire **dale.out** se va scrie:

- pe prima linie, numărul **K** de dale;
- pe a doua linie, **K** numere reprezentând tipurile de dale folosite, în ordinea în care ele vor fi plasate de-a lungul traseului autostrăzii.

**Restricții și precizări**

- $1 \leq N \leq 200$
- $1 \leq L \leq 10000$
- Tipurile de dale se numerotează de la 0.
- Dalele nu au voie să depășească lungimea autostrăzii, nu se pot suprapune și trebuie să fie adiacente.
- Se garantează existența unei soluții.

**Exemplu**

**dale.in**

```
6 4
1 10 2 3 2 2
3 3 9
1 2 4
2 10 15
1 12 10
```

**dale.out**

```
4
1 3 0 1
```

**Explicație**

Primul metru se acoperă cu dala de rezistență 2 (trebuie minim 1) și cost 4, al doilea cu dala de rezistență 12 (trebuie rezistență minim 10) și cost 10, următorii 3 metri se acoperă cu dala de rezistență 3 și cost 9, iar ultimul metru se acoperă cu o nouă dala de rezistență 2 și cost 4. Costul total este 27.

Timp maxim de execuție/test: 1 secundă.

Total memorie disponibilă: 1 MB.

Dimensiunea maximă a sursei: 5 KB.

CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XVII-a, 7–8 aprilie 2017



**Clasele XI-XII**

**Problema 2 – matrice**

**100p**

Fie o matrice binară cu dimensiunile  $m \times n$  căreia  $i$  se asociază două șiruri, numite în continuare șiruri caracteristice: unul de dimensiune  $m$ , conținând în ordine numerele de valori de  $1$  corespunzătoare tuturor liniilor din matrice și altul de dimensiune  $n$ , conținând în ordine numerele de valori de  $1$  corespunzătoare tuturor coloanelor din matrice.

**Cerință**

Cunoscându-se dimensiunile și valorile elementelor a două șiruri, să se determine o matrice binară cu dimensiunile precizate care are ca șiruri caracteristice șirurile date.

**Date de intrare**

Fișierul de intrare **matrice.in** conține pe prima linie, separate printr-un spațiu, două numere naturale  $m$  și  $n$ , reprezentând dimensiunile șirurilor.

Pe cea de-a doua linie a fișierului se găsesc, separate prin câte un spațiu,  $m$  numere naturale, reprezentând elementele primului șir.

Pe cea de-a treia linie a fișierului se găsesc, separate prin câte un spațiu,  $n$  numere naturale, reprezentând elementele celui de-al doilea șir.

**Date de ieșire**

Fișierul de ieșire **matrice.out** va conține  $m$  linii pe care vor fi scrise, fără spații între ele, câte  $n$  valori de  $0$  sau  $1$ , reprezentând elementele unei matrici binare cu specificațiile date.

**Restricții și precizări**

- $1 \leq m, n \leq 1000$
- Toate valorile din fișierul de intrare sunt numere naturale care nu depășesc valoarea  $1000$ .
- Se garantează existența a cel puțin unei matrici cu specificațiile date.
- Dacă există mai multe matrici cu specificațiile date, poate fi afișată oricare dintre ele.

**Exemplu**

<b>matrice.in</b>	<b>matrice.out</b>
5 7	0100011
3 4 4 3 4	0101101
2 3 3 3 3 1 3	1010101
	0011100
	1111000

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

Total memorie disponibilă: **10** MB.

Dimensiunea maximă a sursei: **10** KB.