



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XIV-a, 16– 17 MAI 2014



Clasa a IX-a

Problema 1 – reprezentare Fibonacci

Suntem obișnuiți cu scrierea numerelor într-o bază de numerație B , unde reprezentarea $a_{n-1}a_{n-2}\dots a_1a_0$ corespunde valorii $a_0 + B \cdot a_1 + \dots + B^{n-1} \cdot a_{n-1}$.

Este posibil însă ca factorii cu care se înmulțesc cifrele să nu fie puterile unui număr fix.

Un astfel de sistem de reprezentare este sistemul Fibonacci. În acest sistem, reprezentarea $a_{n-1}a_{n-2}\dots a_1$ corespunde numărului $F_1 \cdot a_1 + \dots + F_{n-1} \cdot a_{n-1}$, unde F_1, \dots, F_{n-1} sunt numerele lui Fibonacci ($F_1=1, F_2=2$ și, pentru $n \geq 3$, avem $F_n = F_{n-1} + F_{n-2}$). În această schemă de reprezentare, orice număr natural poate fi reprezentat utilizând doar cifre 0 și 1, însă această reprezentare nu este unică. De exemplu, numărul 8 are următoarele reprezentări echivalente: $10000_{\text{FIB}}=8$, $1100_{\text{FIB}}=3+5$, $1011_{\text{FIB}}=1+2+5$.

Cerință

Dându-se două reprezentări în sistemul Fibonacci, se cere să se afle dacă acestea reprezintă sau nu același număr.

Intrarea

Fișierul **fib.in** va conține:

- pe prima linie, numărul N de perechi de reprezentări de comparat
- pe următoarele $2 \cdot N$ linii, câte un număr natural nenul, M_i , reprezentând numărul de cifre ale reprezentării Fibonacci, urmat de un șir de M_i numere 0 sau 1 reprezentând cifrele, toate separate prin spații.

Ieșirea

În fișierul **fib.out** se vor scrie N linii, astfel: dacă numerele de pe liniile $2 \cdot i$ și $2 \cdot i + 1$ din fișierul de intrare sunt egale, linia i va consta din cuvântul **DA**, altfel va consta din cuvântul **NU**.

Exemplu:

fib.in	fib.out
3	DA
5 1 0 0 0 0	NU
4 1 0 1 1	DA
5 1 1 0 0 0	
4 1 1 0 0	
5 1 1 1 1 1	
6 1 0 1 0 0 1	

Explicații:

$10000_{\text{FIB}} = 8$	$1100_{\text{FIB}} = 5+3 = 8$
$1011_{\text{FIB}} = 5+2+1 = 8$	$11111_{\text{FIB}} = 8+5+3+2+1 = 19$
$11000_{\text{FIB}} = 8+5 = 13$	$101001_{\text{FIB}} = 13+5+1 = 19$

Restricții și precizări:

- Fiecare număr are cel mult 10000 de cifre, iar prima cifră este 1
- $1 \leq N \leq 100$

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 XIV-a, 16– 17 MAI 2014



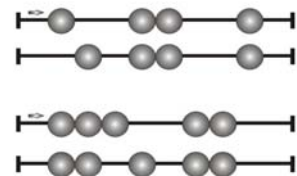
Clasa a IX-a

Problema 2 – bile

Gigel și-a confecționat pentru un experiment un dispozitiv care constă într-o tijă metalică pe care sunt dispuse mai multe bile cu caracteristici identice. Tija străpunge fiecare bilă printr-un canal și este prevăzută la fiecare capăt cu o piedică care nu permite căderea vreunei bile (vezi desenul alăturat). Fiecare bilă de pe această tijă se găsește pe o anumită poziție precizată relativ la capătul din stânga al tijei.



Gigel își imaginează următorul joc: la un moment dat poate fi lovită doar o bilă care se găsește la una din marginile tijei (fie prima bilă din stânga, fie ultima bilă din dreapta). Astfel când Gigel lovește prima bilă din stânga această bilă se deplasează spre dreapta cu o poziție dacă poziția din dreapta este liberă sau, dacă în dreapta bilei sunt mai multe bile succesive, efectul lovirii se propagă până la ultima bilă din succesiune, care se va deplasa spre dreapta cu o poziție (vezi desenul alăturat).



Analog, lovirea ultimei bile din dreapta are ca efect deplasarea unei bile spre stânga cu o poziție.

Cerință

Cunoscându-se numărul de bile și pozițiile acestora pe tijă, să se determine numărul minim de loviri care trebuie aplicate bilelor astfel încât toate bilele să ajungă în poziții succesive.

Date de intrare

Fișierul de intrare **bile.in** conține pe prima linie un număr natural n reprezentând numărul bilelor. Pe cea de-a doua linie a fișierului se găsesc, separate prin câte un spațiu, în ordine crescătoare, cele n poziții ale bilelor de pe tijă.

Date de ieșire

Fișierul de ieșire **bile.out** va conține o singură linie pe care va fi scris numărul minim de loviri necesare pentru ca toate bilele să ajungă în poziții succesive.

Restricții și precizări

- $2 \leq n \leq 50000$
- Poziția maximă a oricărei bile de pe tijă este cel mult 5000000.
- Pentru 50% din teste $n \leq 1000$ și poziția maximă a oricărei bile de pe tijă este cel mult 50000.

Exemplu

bile.in	bile.out	Explicație
4	7	Sunt necesare 7 loviri ale bilelor pentru ca acestea să ajungă, de exemplu, în pozițiile 3 4 5 6.
2 4 7 10		

Timp maxim de execuție/test: 0.3 secunde.

Total memorie disponibilă: 1 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XIV-a, 16– 17 MAI 2014



Clasa a X-a

Problema 1 – negativ

Suntem obișnuiți cu scrierea numerelor într-o bază de numerație B , unde reprezentarea $a_{n-1}a_{n-2}\dots a_1a_0$ corespunde valorii $a_0 + B \cdot a_1 + \dots + B^{n-1} \cdot a_{n-1}$, unde B este un număr natural mai mare sau egal cu 2 și fiecare cifră este cuprinsă între 0 și $B-1$ inclusiv.

Acest sistem se poate extinde în diverse moduri, unul dintre acestea este să permitem ca baza de numerație să fie un număr negativ. În acest caz, B este un număr întreg, $B \leq -2$. Cifrele rămân însă numere naturale, $0 \leq a_i < -B$. Avantajul unei astfel de scrieri este acela că putem reprezenta orice număr întreg (posibil negativ), folosind un număr corespunzător de cifre. Reprezentarea este unică dacă impunem condiția ca prima cifră a numărului să nu fie zero.

Cerință

Dându-se două numere în reprezentarea într-o aceeași bază negativă, se cere să se calculeze reprezentarea sumei lor în aceeași bază.

Intrarea

Fișierul **negativ.in** va conține:

- pe prima linie, baza B
- pe fiecare din următoarele 2 linii, numărul N_i de cifre ale unui număr, urmat de N_i numere, separate prin spații, reprezentând cifrele numărului.

Ieșirea

În fișierul **negativ.out** se va scrie, pe o singură linie, numărul N de cifre ale rezultatului, urmat de N numere reprezentând cifrele rezultatului, toate separate prin spații.

Exemplu

negativ.in

```
-3
4 1 1 0 2
3 2 1 0
```

negativ.out

```
2 1 2
```

Explicație

$$1102_{-3} = 1 \cdot (-27) + 1 \cdot 9 + 0 \cdot (-3) + 2 = -16$$

$$210_{-3} = 2 \cdot 9 + 1 \cdot (-3) + 0 = 15$$

$$\text{rezultatul este } -1 = 1 \cdot (-3) + 2 = 12_{-3}$$

Restricții și precizări

- $-2 \geq B \geq -10000$
- $1 \leq N_i \leq 30000$
- Atât în datele de intrare, cât și în cele de ieșire, prima cifră a fiecărui număr va fi diferită de zero. Numărul zero se va reprezenta ca un șir de zero cifre.

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 XIV-a, 16– 17 MAI 2014



Clasele a X-a

Problema 2 – paranteze

Dându-se un șir de paranteze, să se determine lungimea maximă a unei secvențe din șir în care parantezele se închid corect.

Date de intrare

Fișierul de intrare **paranteze.in** conține următoarele date:

- pe prima linie un număr natural n reprezentând numărul parantezelor din șir;
- pe cea de-a doua linie a fișierului se găsesc, fără spații între ele, cele n paranteze ale șirului.

Date de ieșire

Fișierul de ieșire **paranteze.out** va conține o singură linie pe care va fi scrisă o singură valoare reprezentând lungimea maximă a unei secvențe din șir în care parantezele se închid corect.

Restricții și precizări

- $1 \leq n \leq 1000000$
- În cadrul șirului pot apărea doar parantezele deschise (, [și {, precum și parantezele închise corespunzătoare),] și }.
- Pentru 30% din teste $n \leq 1000$.
- Pentru 50% din teste șirul va conține doar un tip de paranteză deschisă și perechea ei.

Exemple

paranteze.in	paranteze.out
10	6
((()())(())	

paranteze.in	paranteze.out
10	8
][(){}[]])	

Timp maxim de execuție/test: 0.2 secunde.

Total memorie disponibilă: 10 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XIV-a, 16– 17 MAI 2014



Clasele XI-XII

Problema 1 – angrenaj

Un angrenaj constă în niște osii pe care se găsesc roți dințate. Unele perechi de roți dințate sunt angrenate. Dacă pe axul a_1 se găsește o roată dințată cu n_1 dinți care este angrenată cu o roată cu n_2 dinți situată pe axul a_2 , atunci la fiecare rotație a axului a_1 în sens orar, axul a_2 efectuează n_1/n_2 rotații în sens antiorar.

Cerință

Să se afle ce roți se rotesc, în ce sens și cât, la fiecare rotație a axului 1.

Intrarea

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

- pe prima linie, două numere naturale, N și M , reprezentând numărul de osii și numărul de perechi de roți angrenate
- pe următoarele M linii, câte patru numere naturale, A, B, N_a, N_b , reprezentând respectiv numărul axului pe care se găsește prima roată, numărul axului pe care se află a doua roată, numărul de dinți de pe prima roată și numărul de dinți de pe a doua roată. Axele sunt numerotate de la 1 la N .

Ieșirea

În fișierul **angrenaj.out** se vor scrie N linii, câte una corespunzând fiecărui ax.

Pe fiecare linie se vor scrie două numere întregi, P și Q , cu semnificația că, pentru fiecare rotație a axului 1, axul corespunzător liniei curente face P/Q rotații. Numărul P este pozitiv dacă axul curent se rotește în același sens cu axul 1, negativ dacă axul curent se rotește în sens contrar axului 1 și 0 dacă axul curent nu se rotește deloc. Numărul Q trebuie să fie strict pozitiv. Dacă axul 1 nu se poate roti deloc, P va fi 0 pentru toate axele.

Exemple

angrenaj.in	angrenaj.out	angrenaj.in	angrenaj.out
5 4	1 1	5 4	0 1
1 2 4 5	-4 5	1 2 5 5	0 1
2 3 7 8	7 10	2 3 5 5	0 1
2 3 21 24	0 1	1 3 5 5	0 1
4 5 3 4	0 1	4 5 3 4	0 1

Restricții și precizări

- $1 \leq N \leq 200$
- $0 \leq M \leq 200$
- Pentru fiecare roată dințată, numărul de dinți este cuprins între 1 și 99.
- Dacă un ax nu are legătură nici direct nici indirect cu axul 1, el nu se rotește deloc.
- Numerele scrise în fișierul de ieșire nu este necesar să reprezinte fracții ireductibile.
- Pentru cel puțin 60% dintre teste, toate numerele P și Q au valorile absolute mai mici de 2^{31} .

Timp maxim de execuție/test: 5 secunde.

Total memorie disponibilă: 1 MB.

Dimensiunea maximă a sursei: 5 KB.



CONCURSUL INTERJUDEȚEAN DE MATEMATICĂ ȘI INFORMATICĂ

„MARIAN ȚARINĂ”

Ediția a XIV-a, 16– 17 MAI 2014



Clasele XI-XII

Problema 2 – șiruri

Dându-se un șir de litere mici ale alfabetului englez, să se determine:

- numărul subșirurilor crescătoare ale acestui șir modulo 41357;
- numărul minim de subșiruri crescătoare în care poate fi partiționat acest șir.

Date de intrare

Fișierul de intrare **siruri.in** conține următoarele date:

- pe prima linie un număr natural p , pentru toate testele de intrare, numărul p putând avea doar valoarea 1 sau valoarea 2;
- pe a doua linie un număr natural n reprezentând numărul literelor din șir;
- pe cea de-a treia linie a fișierului se găsesc, fără spații între ele, cele n litere ale șirului.

Date de ieșire

Fișierul de ieșire **siruri.out** va conține o singură linie pe care va fi scrisă o singură valoare reprezentând rezultatul cerinței a dacă valoarea lui p este 1 sau rezultatul cerinței b dacă valoarea lui p este 2.

Restricții și precizări

- $1 \leq n \leq 1000000$
- Se vor număra doar subșirurile nevide.
- Fiecare dintre cerințe reprezintă 50% din punctaj.
- Pentru 30% din teste $n \leq 10000$.

Exemple

siruri.in	siruri.out
1	41
7	
acabadc	

siruri.in	siruri.out
2	3
7	
acabadc	

Timp maxim de execuție/test: 0.2 secunde.

Total memorie disponibilă: 10 MB.

Dimensiunea maximă a sursei: 5 KB.