

Tema 36

În parcul din orașul meu se află o alee dreptunghiulară construită din N dale pătratice identice numerotate de la 1 la N :

Într-o zi, la un capăt al aleei, în fața dalei 1, se află greierașul Cri iar la celălalt capăt, după dala N , se află broscuța Oac. Mare le-a fost bucuria întâlnirii! Încă o dată se vor întrece în parcurgerea aleei. Oare cine va ajunge cel mai repede la capătul opus al aleei, în locul celuilalt? Știm că Oac și Cri se deplasează de-a lungul aleei în salturi. Cri sare din X în X dale iar Oac din Y în Y dale. Este posibil ca în deplasarea lor, Cri și Oac să ajungă în același moment pe aceeași dală, motiv pentru a face o mică pauză de “reîmpărsărire”.

Cerințe

Scrieți un program care să citească numerele naturale N , X și Y și care să determine: **a)** numărul minim de salturi S necesare pentru parcurgerea aleei de către Cri sau Oac; **b)** numărul dalei D pe care ajung în același moment Cri și Oac.

Date de intrare

Fișierul de intrare `crioac.in` conține pe prima linie cele trei numere naturale N , X și Y , separate prin câte un spațiu.

Date de ieșire

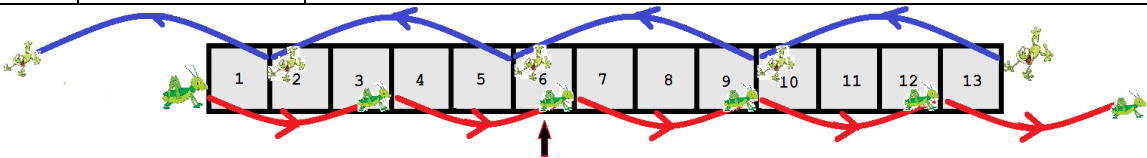
Fișierul de ieșire `crioac.out` va conține

- pe prima linie numărul natural S reprezentând numărul minim de salturi necesare ca unul dintre cei doi (Cri și Oac) să parcurgă aleea;
- pe a doua linie, un număr natural D reprezentând numărul dalei pe care ajung în același moment Cri și Oac; dacă cei doi nu ajung în același moment pe o aceeași dală în timpul parcurgerii aleei, atunci fișierul va conține pe linia a doua valoarea 0 (zero).

Restricții și precizări:

- $1 \leq N, X, Y \leq 30\,000$
- Pentru cerința **a)** se acordă 30% din punctaj iar pentru cerința **b)** se acordă 70% din punctaj;

Exemple:

	<code>crioac.in</code>	<code>crioac.out</code>	<i>Explicații</i>
1)	13 3 4	4 6	Cri sare din 3 în 3 dale, ajungând pe dalele 3,6,9,12 (în această ordine). El va face 5 salturi pentru a parcurge aleea. Oac sare din 4 în 4 dale ajungând pe dalele: 10,6,2 în această ordine. Oac va face 4 salturi pentru parcurgerea aleei. Cei doi se întâlnesc pe dala 6 (după al doilea salt).
			
2)	5 3 2	2 0	Cri sare din 3 în 3 dale, ajungând pe dala 3. El va face 2 salturi pentru a parcurge aleea. Oac sare din 2 în 2 dale ajungând pe dalele: 4,2 în această ordine. Oac va face 3 salturi pentru parcurgerea aleei. Cei doi nu vor ajunge pe aceeași dală, deci nu vor face pauză.
			
3)	4 2 4	2 0	Cri sare din 2 în 2 dale, ajungând pe dalele 2 și 4 în această ordine, El va face 3 salturi pentru a parcurge aleea. Oac sare din 4 în 4 dale ajungând pe dala 1. Oac va face 2 salturi pentru parcurgerea aleei. Cei doi nu vor ajunge pe aceeași dală, deci nu vor face pauză.
			

Timp maxim de executare: 0.2 secunde/test

Limite de memorie: total memorie disponibilă 2 Mb, din care pentru stivă maxim: 2 Mb.

Dimensiunea maximă a sursei 5 KB

Tema 37

Gobu este un apărător cosmic. El dorește să stabilească echilibrul pe care l-a distrus Vrăjitorul Negru între lumea reală și lumea digitală. Pentru aceasta el trebuie să descopere un număr magic. Zilele trecute a primit de la prietenul său Ank un bilețel care conține o înșiruire de cifre binare **0** și **1**. Acesta i-a spus:

”Transformă înșiruirea de cifre binare într-un număr natural zecimal **N**, apoi găsește două numere distincte **A** și **B** formate din cel mult două cifre aflate pe poziții consecutive în scrierea zecimală a numărului **N**, citindu-l pe **N** de la stânga la dreapta și de la dreapta la stânga, astfel încât produsul acestor două numere să fie maxim. Numărul magic (**P**), cel pe care îl cauți, este cel mai mare pătrat perfect la care se împarte produsul celor două numere **A** și **B**”.

Gobu vă roagă să-l ajutați să descopere numărul magic **P** de care are nevoie pentru a-l învinge pe Vrăjitorul Negru.

Cerința

Scrieți un program care să determine numărul magic **P**.

Date de intrare

Fișierul de intrare *magic.in* conține pe prima linie un număr natural **K** iar pe a doua linie un șir de **K** cifre binare **0** și **1**, separate prin câte un spațiu având prima cifră 1.

Date de ieșire

În fișierul de ieșire *magic.out* se va scrie un număr natural reprezentând numărul magic **P** căutat.

Restricții și precizări

$$7 \leq K \leq 31$$

Exemplu

<i>magic.in</i>	<i>magic.out</i>
31 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1 0 1 1 1	1296
Explicații	
N=2004318071 A=80 B=81 A*B=81*80=81*16*5 Cel mai mare patrat este P=1296	
8 1 1 1 0 1 0 0 1	16
Explicații	
N=233 A=33 B=32 A*B=33*32=33*16*2 Cel mai mare patrat este P=16	

Timp maxim de executare: 0.2 secunde/test

Limite de memorie: total memorie disponibilă **2 Mb**, din care pentru stivă maxim: **2 Mb**.

Dimensiunea maximă a sursei 5 KB