



Clasele V-VI

Sursa: ID1.cpp, ID1.c, ID1.pas

Problema 1 – platou

100 puncte

Fiind dat un șir de numere, denumim *secvență* a acestuia o parte dintre termenii șirului luați de pe poziții consecutive. Denumim *platou* al acestui șir o secvență formată din valori identice. *Lungimea* unui platou este egală cu numărul de elemente care îl formează.

De exemplu, în șirul de numere **1 1 1 7 7 3 4 4 4 7 7** avem:

- platourile **1 1 1** și **4 4 4** ambele având lungimea **3**;
- platourile **7 7** (cel care începe în poziția a patra) și **7 7** (cel care începe pe poziția a zecea), ambele având lungimea **2**;
- platoul **3** care are lungimea **1**.

În schimb nu avem platoul **7 7 7 7** deoarece cele patru elemente egale cu **7** nu sunt pe poziții consecutive!

Se dă un șir de **n** numere. Fiecare dintre aceste numere aparține intervalului **[0,1000000]**.

Asupra acestui șir se pot efectua o singură dată următoarele două operațiuni în această ordine:

1. se extrage un platou la alegere;
2. se inserează platoul extras la pasul anterior într-o poziție la alegere din șirul rezultat după extragere.

De exemplu, dacă avem următorul șir initial:

2 2 5 0 5 8 8 8 4 9 9 9 0 0 2 2 8

extragerem platoul **2 2** format din elementele aflate în penultima și antepenultima poziție și obținem șirul:

2 2 5 0 5 8 8 8 4 9 9 9 0 0 8

În șirul rezultat după rezultat inserăm platoul **2 2** (pe care l-am extras în pasul anterior) în poziția a doua și obținem șirul:

2 2 2 2 5 0 5 8 8 8 4 9 9 9 0 0 8

Cerințe

Să se scrie un program care pentru un șir dat determină:

1. Lungimea maximă a unui platou din șirul dat inițial precum și valoarea cea mai mare a elementelor care formează un platou de lungime maximă.
2. Lungimea maximă a unui platou care poate să apară în șir în urma efectuării celor două operațiuni precum și valoarea cea mai mare a elementelor care ar putea forma un astfel de platou.

Date de intrare

Fișierul **platou.in** conține:

- pe prima linie un număr natural **V** care poate avea valoarea **1** sau valoarea **2**;
- pe a doua linie un număr natural **n**, $1 \leq n \leq 1000000$;
- pe a treia linie un șir de **n** numere naturale separate prin câte un spațiu, reprezentând elementele șirului dat. Fiecare dintre aceste numere aparține intervalului **[0,1000000]**.

Date de ieșire

Fișierul **platou.out** va avea o singură linie care va avea următorul conținut:

- dacă **V=1**, atunci pe această linie se vor scrie cele două numere care reprezintă răspunsul la prima cerință.
- dacă **V=2**, atunci pe această linie se vor scrie cele două numere care reprezintă răspunsul la a doua cerință.

Restricții și precizări

- Numerele din șirul de pe a doua linie sunt numere naturale din intervalul **[0,1000000]**.
- Pentru rezolvarea corectă a primei cerințe se obțin 20 de puncte, iar pentru rezolvarea corectă a celei de a doua cerințe se obțin 80 puncte.

Exemple

platou.in	platou.out	Explicații
1 16 2 2 5 0 5 8 8 8 4 9 9 9 0 8 2 2	3 9	v=1 , deci se rezolvă NUMAI prima cerință. Lungimea maximă a unui platou din șirul dat este 3. Sunt două astfel de platouri: 8 8 8 și 9 9 9 . Valoarea cea mai mare din care este format unul dintre aceste platouri este 9 .
2 16 2 2 5 0 5 8 8 8 4 9 9 9 0 8 2 2	4 8	v=2 , deci se rezolvă NUMAI a doua cerință. În urma executării celor două operațiuni pot să apară platouri de lungime maximă 4 în două moduri: 2 2 2 2 respectiv 8 8 8 8 . Valoarea cea mai mare din care este format unul dintre aceste platouri este 8 .

Timp maxim de execuție/test: 1,5 secunde

Total memorie disponibilă 64 MB din care stiva 64 MB

Dimensiunea maximă a sursei: 20 KB.



Clasele V-VI

Sursa: ID2.cpp, ID2.c, ID2.pas

Problema 2 vali puncte

100

De Ziua Îndrăgostiților Vali a hotărât să organizeze o petrecere mare pe stadionul orașului. La petrecere pot participa numai și numai îndrăgostiți care și-au procurat biletele din timp.

Biletele oricărui cuplu sunt aproape identice. Ele au proprietatea că numărul de serie are prima cifră 1 pentru băieți și prima cifră 2 pentru fete, continuarea celor două numere este identică. De exemplu, dacă prietenul are biletul cu numărul 134, atunci prietena lui are biletul 234, iar dacă prietena are biletul 2234567890, atunci prietenul ei are biletul 1234567890.

Bineînțeles, că și Vali are bilet și speră să câștige un cadou. Biletul lui Vali are aceeași proprietate ca celelalte bilete, cu o singură excepție: Vali nu va participa cu pereche la acest eveniment.

Pe parcursul desfășurării petrecerii, biletele de intrare vor avea și rolul de bilet de tombolă. Acestea vor fi introduse într-o urnă și câteva dintre ele vor fi extrase, iar norocoșii proprietari ai biletelor vor primi cadouri valoroase. Se mai știe că există probabilitatea ca unii participanți să vină la petrecere cu bilete falsificate, având numere identice cu cele de pe un bilet original. Acest fapt se va depista cu ușurință în momentul extragerii.

Cerință

Cunoscând numerele de serie ale tuturor biletelor de intrare la acest eveniment, aflați:

- dacă pe Vali o cheamă Valentina sau îl cheamă Valentin;
- numărul biletului lui Vali.

Date de intrare

Fișierul `vali.in` are următoarea structură:

Pe prima linie un număr natural p . Pentru toate testele de intrare, numărul p poate avea doar valoarea 1 sau valoarea 2. Pe a doua linie un număr natural n , reprezentând numărul de participanți. Următoarele n linii conțin câte un număr de bilet.

Date de ieșire

Fișierul `vali.out` are următoarea structură:

Dacă valoarea lui p este 1, **se va rezolva numai punctul a)** din cerință. În acest caz, în fișierul de ieșire se va scrie un singur cuvânt: `valentin` sau `valentina` (totul cu litere mici!), în concordanță cu numele lui Vali din certificatul de naștere.

Dacă valoarea lui p este 2, **se va rezolva numai punctul b)** din cerință. În acest caz, în fișierul de ieșire se va scrie un număr natural, reprezentând numărul de pe biletul lui Vali.

Restricții și precizări

- $3 \leq n \leq 299999$;

- pentru 30% din teste avem $p=1$, iar pentru 70% din teste avem $p=2$;
- pentru 90% din teste numărul biletului nu depășește tipul long (C++) respectiv longint (Pascal), pentru 10% din teste există bilete cu numere până la 17 cifre.
- pentru 90% din teste biletul lui Vali nu este falsificat, pentru 10% din teste numărul biletului lui Vali coincide cu numărul unui alt bilet xeroxat.

Exemple

vali.in	vali.out	Explicație
1 5 1345 2649012 15480 2345 1649012	valentin	$p=1$, se rezolvă NUMAI cerința 1 Singurul bilet fără pereche este 15480, deci posesorul este un băiat , îl cheamă Valentin (scris cu 'v' mic în fișier).
1 9 1345 2649012 2480 2345 2345 1345 1649012 2345 1345	valentina	$p=1$, se rezolvă NUMAI cerința 1 Biletul fără pereche este 2480, deci posesorul este fată , o cheamă Valentina (toate literele fiind mici). Observăm că în acest caz biletele 1345 și 2345 au fost falsificate chiar de două ori.
2 7 2947899999 10000345 2947899999 20000345 20000345 10000345 1947899999	2947899999	$p=2$, se rezolvă NUMAI cerința 2 Biletul lui Vali are numărul 2947899999 putem observa că numărul acestui bilet coincide cu un alt bilet falsificat, mai sunt și alte bilete falsificate.
2 5 10000345 20000345 2947 10000345 1947	10000345	$p=2$, se rezolvă NUMAI cerința 2 Biletul lui Vali are numărul 10000345 observăm că numărul acestui bilet coincide cu un alt bilet falsificat, alte bilete falsificate nu există.

Timp maxim de execuție/test: 0,8 secunde.

Memorie totala: 32 MB din care 32 MB pentru stivă.

Dimensiunea maximă a sursei: 10 KB