

## PROBLEMA 1

100 puncte

## COD

Intrarea în biblioteca universității este permisă pe baza unui cod de acces. Fiecare cititor dispune de un cod pe baza căruia poate consulta doar anumite categorii de date din resursele bibliotecii. Codul său este format printr-o combinație de cifre și codul de acces la o anumită categorie este format din cifrele comune celor două coduri: codul categoriei și codul cititorului, luate în ordine crescătoare, o singură dată.

## Cerință

Cunoscând numărul  $N$  de persoane care doresc să acceseze resursele bibliotecii într-o zi, codul categoriei și codurile utilizatorilor scrieți numărul utilizatorilor neautorizați (situațiile în care codul de acces nu se poate forma) și codul de acces al ultimului utilizator autorizat din acea zi.

## Date de intrare

Fișierul cod.in conține pe prima linie numărul  $N$  de persoane, pe a doua linie codul categoriei, pe următoarele  $N$  linii codurile utilizatorilor din acea zi.

## Date de ieșire

Fișierul cod.out are următoarea structură :

- pe prima linie o valoare naturală nenulă care reprezintă numărul utilizatorilor neautorizați sau mesajul “nici un utilizator neautorizat”(fără ghilimele), în cazul în care au fost autorizați toți utilizatorii în ziua respectivă.
- pe a doua linie valoarea codului de acces al ultimului utilizator autorizat din acea zi sau mesajul “nici un utilizator autorizat” (fără ghilimele), dacă au fost detectați doar utilizatori neautorizați în acea zi.

## Restricții și precizări

- $1 \leq N \leq 1000000$
- Codul categoriei și codurile utilizatorilor sunt scrise după următoarea regulă: primele 2 caractere din șir reprezintă numărul de caractere ale codului respectiv (dacă numărul este format dintr-o singură cifră aceasta va fi precedată de cifra 0), următoarele caractere formează codul propriu-zis. Codurile sunt combinații de maxim 16 cifre.

## Exemple

cod.in	cod.out	Explicatii:
6 049977 03076 018 1486357049194490 0821682491 132608702956340 15083887847628791	1 79	Este un utilizator neautorizat, acesta nu are în codul lui (018 are codul format dintr-o singură cifră, cifra 8) cifre comune cu cifrele codului categoriei (049977 – codul categoriei este format din 4 cifre - 9977) Codul de acces al ultimului utilizator autorizat din acea zi este format din cifrele comune celor două coduri luate în ordine crescătoare o singură dată - 79

cod.in	cod.out	Explicatii:
1 049977 0578491	nici un utilizator neautorizat 79	Nu există utilizatori neautorizați, Codul de acces al ultimului utilizator autorizat din acea zi este format din cifrele comune celor două coduri luate în ordine crescătoare o singură dată - 79

**Timp maxim de execuție:** 0,3 secunde/test.

**Memorie totală disponibilă** 4 MB, din care 2 MB pentru stivă

**Dimensiunea maximă a sursei:** 5 KB

## PROBLEMA 2

100 puncte

### JOC

Ionel are la dispoziție două pungi cu piese de joc. Într-o pungă sunt piese roșii pe care se află înscrisă valoarea 0. În cealaltă pungă sunt piese albastre pe care sunt notate diverse numere naturale nenule. Ionel își propune mai întâi să construiască un șir de  $n$  piese albastre. Deoarece nu îi plac numerele prime, el se gândește să înlocuiască piesele ce conțin aceste numere cu piese roșii. Apoi îi vine ideea să găsească niște piese câștigătoare și repetă următorii doi pași:

1. înlocuiește piesele cu numerele rămase cu piese ce au ca valoare suma cifrelor numărului plus unu.
2. observă că au apărut alte numere prime și le înlocuiește și pe acestea cu piese roșii.

Ionel continuă jocul până când înlocuiește toate piesele albastre cu piese roșii. Ultimele piese înlocuite le declară câștigătoare.

### Cerință

Determinați numerele câștigătoare. În cazul în care toate numerele extrase inițial sunt numere prime, Ionel consideră că nu există piese câștigătoare și se afișează mesajul **Nu exista**.

### Date de intrare

Fișierul **joc.in** în care pe prima linie se află numărul  $N$ , iar pe linia a doua se află numerele pe care le extrage Ionel din pungă cu piese albastre pentru a construi șirul inițial.

### Date de ieșire

Fișierul **joc.out** conține numerele câștigătoare.

### Restricții și precizări:

- $N < 100000$
- numerele înscrise pe piesele albastre  $\leq 33000$
- Piesele sunt suficiente și conțin toate numerele necesare desfășurării jocului.

## Exemplu

joc.in	joc.out	Explicație
5 7 20 30 44 13	44	Se înlocuiesc numerele prime: 0 20 30 44 0 Se calculează suma cifrelor + 1: 0 3 4 9 0 Se înlocuiesc numerele prime: 0 0 4 9 0 Se calculează suma cifrelor + 1: 0 0 5 10 0 Se înlocuiesc numerele prime: 0 0 0 10 0 Se calculează suma cifrelor + 1: 0 0 0 2 0 Se înlocuiesc numerele prime: 0 0 0 0 0 Numărul câștigător este: 44

**Timp maxim de execuție:** 0,2 secunde/test.

**Memorie totală disponibilă:** 4 MB, din care 2 MB pentru stivă

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

## PROBLEMA 3

100 puncte

## DECATHLON

La concursul național de decathlon pentru elevi "Cupa Primăverii", participă  $N$  concurenți, numerotați de la 1 la  $N$ . Fiecare concurent trebuie să participe obligatoriu la cele 10 probe ale concursului, în ordinea stabilită de organizatori prin tragere la sorți. La finalul concursului, fiecare concurent primește punctajele pe probe, în ordinea evoluției în concurs, sub forma unor perechi  $(x,y)$ , unde  $x$  este proba, iar  $y$  punctajul obținut.

Organizatorii doresc să premieze concurenții astfel:

- Câte un premiu cu medalie pentru fiecare câștigător al unei probe. Dacă există mai mulți concurenți cu același punctaj la o probă, se premiază doar primul care a obținut acest punctaj la proba respectivă, iar dacă și acestea sunt egale, se premiază cel cu numărul de ordine mai mic.
- Un premiu cu medalie pentru câștigătorul concursului (punctajul maxim obținut prin însumarea punctajelor obținute de concurent la toate probele). Dacă există mai mulți concurenți cu același punctaj, se vor premia toți.
- Un premiu onorific pentru liga națională corespunzătoare probei cu punctaj maxim din concurs. Dacă există mai multe, se vor afișa toate
- Deasemeni, se dorește identificarea concurenților cu cel puțin  $k$  medalii

**Datele de intrare** se citesc din fișierul *decathlon.in* și sunt de forma:

Pe prima linie numărul  $N$ , reprezentând numărul de concurenți din concurs și un număr  $k$ , reprezentând numărul minim de medalii

Pe următoarele  $N$  linii câte 10 perechi de forma  $x y$  cu semnificația din enunț

**Datele de ieșire** vor fi afișate în fișierul *decathlon.out* sub forma:

Pe primele 10 linii, 10 perechi de numere, reprezentând numerele de ordine ale concurenților care au câștigat fiecare dintre cele 10 probe și punctajele pe care le-au obținut la aceste probe

- Pe următoarea linie punctajul total și numărul de ordine al concurentului / concurenților câștigători

- Pe următoarea linie punctajul total maxim dintre toate probele și numărul/ numerele de ordine asociate acestor probe cu punctaj total maxim

- Pe ultima linie numerele de ordine ale concurenților cu cel puțin  $k$  medalii, în ordinea numerelor de concurs sau 0, dacă nu există nici un concurent care să fi obținut  $k$  medalii.

**Restricții și precizări**

- $1 \leq N \leq 100\,000$ ,  $0 \leq k \leq 10$
- $0 \leq$  punctajele obținute de concurenți la fiecare probă  $\leq 100$

**Exemplu**

decathlon.in	decathlon.out	Explicații
3 3	3 6	La proba 1 s-au obținut punctajul maxim 6, de către concurentul 3
1 5 2 3 5 6 4 10 7 1 6 4 8 5 10 10 9 8	3 9	
3 5	2 8	La proba 2, s-a obținut punctajul maxim 9 tot de către concurentul 3
1 4 2 3 3 8 4 10 5 20 6 20 7 4 8 6 9	1 10	
10 10 15	2 20	La proba 4 s-a obținut punctajul maxim 10, de către concurenții 1 și 2, care au evoluat la același moment de timp, și a câștigat 1, cu numărul de ordine mai mic) etc.
10 4 9 8 8 10 7 15 6 8 5 5 4 10 3 5 2	2 20	
9 1 6	3 15	
	3 10	
	2 10	La proba 10 s-a obținut punctajul maxim 15, de către concurentul 2.
	2 15	
	100 2	Concursul a fost câștigat de către concurentul 2 cu 100 puncte
	32 6	
	2 3	La proba 6 a fost obținut un punctaj total de 32 puncte, care este maximul dintre punctajele de la toate probele
		Concurenții care au mai mult de 3 medalii sunt concurenții 2 (cu 6 medalii, 5 la probe și una la total, respectiv concurentul 3, cu 4 medalii)

**Timpe maxim de execuție:** 0,5 secunde/test.

**Memorie totală disponibilă** 2 MB, din care 2 MB pentru stivă

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

**PROBLEMA 4****100 puncte****REPARTIȚIE**

Viitorii elevi ai clasei pregătitoare au fost evaluați psihologic și somatic primind un punctaj (număr natural  $\leq 100000$ ). Urmează a fi repartizați în clase astfel încât numărul claselor să fie cât mai mic posibil. Un elev este repartizat de îndată ce se prezintă la înscriere singurul criteriu fiind acela că în clasa sa nu există până la atunci nici un punctaj mai mare decât al său.

**Cerință**

Determinați punctajul maxim și numărul  $c$  al claselor ce se vor forma.

**Date de intrare**

Fișierul de intrare **repartitie.in** conține punctajele elevilor în ordinea înscrierii.

**Date de ieșire**

Fișierul de ieșire **repartitie.out** va conține pe prima linie cele două valori cerute separate prin spațiu.

**Restricții și precizări**

- $c \leq 100000$
- punctajul unui elev este un număr natural  $\leq 100000$

**Exemplu**

<b>repartitie.in</b>	<b>repartitie.out</b>	<b>Explicații</b>
4 2 4 2 7 10 9 11 8	11 3	Se formează clasele: 4 4 7 10 11 2 2 9 8

**Timp maxim de execuție:** 0,5 secunde/test

**Memorie totală disponibilă:** 20 MB, din care 18 MB pentru stivă

**Dimensiune maximă a sursei:** 5 KB