Faza JUDEȚEANĂ a olimpiadei de INFORMATICĂ pentru GIMNAZIU

Vă prezentăm în continuare enunțurile celor opt probleme propuse în acest an la faza județeană a olimpiadei de informatică pentru clasele V-VIII. Subiectele au fost aceleași în toate județele țării.

Clasa a V-a

P030411: Cifre

Se consideră două numere naturale *a* și *b* care pot avea cel mult 9 cifre.

Să se determine cifrele distincte, comune numerelor *a* și *b* și numărul cel mai mare format din toate cifrele lui *a* și *b*.

Date de intrare

De la tastatură se citesc numerele *a* și *b*.

Date de ieșire

Pe ecran se vor afișa pe prima linie cifrele distincte comune ale lui *a* și *b* și pe cea de-a doua linie cel mai mare număr format din toate cifrele lui *a* și *b*.

Precizare

- datele de intrare sunt corecte (nu necesită validare).

Exemplu

Intrare

```
a=2115
b=29025
```

Ieșire

```
2 5
955222110
```

Timp de execuție: 1 secundă/test

P030412: Concurs

La un concurs de matematică participă elevi din mai multe școli din diferite orașe. Pentru a se putea deosebi între ele lucrările lor, fiecare lucrare este codificată printr-un număr natural cu 3 cifre, să zicem *abc*, unde *a* (cifra sutelor) este codul orașului, *b* (cifra zecilor) este codul școlii din orașul *a*, iar *c* (cifra unităților) este codul unui elev din școala *b* din orașul *a*.

De exemplu, lucrarea cu codul 328 este lucrarea elevului cu codul 8 de la școala cu codul 2 din orașul având codul 3.

Se cunosc: un cod (al lucrării unui elev *H*, prietenul nostru), numărul *n* de lucrări premiate și codurile fiecăreia dintre ele.

Se cere:

a) să se verifice dacă *H* este premiant sau nu.
b) să se determine numărul de premii obținute de elevii din orașul lui *H* (inclusiv *H*, dacă a fost premiat).
c) să se determine numărul de premii obținute de elevii din școala lui *H* (inclusiv *H*, dacă a fost premiat).

Date de intrare

De la tastatură se citesc: codul lui *H*, *n* și codurile celor *n* lucrări premiate.

Date de ieșire

Cele trei răspunsuri se vor afișa pe ecran, pe câte un rând.

Pentru prima cerință se va afișa un mesaj (`DA` sau `NU`), după cum *H* a obținut, sau nu un premiu. Corespunzător ultimelor două cerințe se va scrie câte un număr.

Fiecare răspuns va fi precedat de o literă (*a*, *b* sau *c*), pentru a se preciza la care dintre cerințe s-a răspuns.

Restricții și precizări

- codurile orașelor (cifra sutelor din fiecare cod) sunt cuprinse între 1 și 5;

- codurile școlilor din fiecare oraș (cifra zecilor) sunt cuprinse între 0 și 9;
- codurile elevilor din fiecare școală (cifra unităților) sunt cuprinse între 0 și 9;
- *n* este un număr natural nenul mai mic decât 20;
- datele de intrare sunt corecte (nu necesită validare).

Exemplu

Intrare

```
H=234
n=6
123 232 125 222 421 235
```

Ieșire

```
a) NU
b) 3
c) 2
```

Explicație

Codul lui *H* nu se regăsește printre cele citite (premiate), deci mesajul este NU.

Lucrările premiate din orașul lui *H* (cu cifra sutelor egală cu 2) sunt în număr de trei: 232, 222 și 225.

Lucrările premiate din școala lui *H* (cu cifra sutelor egală cu 2 și cu cifra zecilor egală cu 3) sunt în număr de două: 232 și 235.

Timp de execuție: 1 secundă/test

Clasa a VI-a

P030413: Număr de control

Gigel a primit spre păstrare un set de *n* cutii de greutăți nu neapărat distincte.

El a cântărit cutiile și pentru fiecare greutate distinctă a notat pe o foaie, în ordine crescătoare a greutăților, numărul de cutii cu greutatea respectivă.

Deoarece fratele său mai mic avea prostul obicei să se joace cu numerele scrise de el pe foaie, *Gigel* s-a gândit să calculeze un "număr de control" după următorul algoritm: începând de la primul număr a grupat numerele de apariții ale greutăților câte trei (dacă îi rămân numere negrupate la sfârșit, le ignoră).

Dacă într-un grup sunt numai numere pare sau numai impare notează grupul cu cifra 1, altfel îl notează cu cifra 0. Din șirul astfel obținut, se formează un număr care are ca valoare cifra zecilor egală cu numărul de valori 1 și cifra unităților egală cu numărul de valori 0, obținându-se astfel "numărul de control".

Cunoscând greutățile cutiilor, să se determine "numărul de control" și să se verifice dacă este număr prim.

Date de intrare

De la tastatură se vor citi numărul *n* și greutatea fiecărei cutii.

Date de ieșire

Pe ecran se va scrie pe prima linie "numărul de control", iar pe următoarea linie se va scrie 1 dacă numărul este prim, respectiv 0 în caz contrar.

Restricție

- $1 \le n \le 100$;
- fiecare greutate este un număr natural, mai mic sau egal cu 200;
- datele de intrare sunt corecte (nu necesită validare);
- se vor afișa doar două valori, fără mesaje, pe două linii diferite.

Exemplu

Intrare

```
n=21
1 3 2 6 2 6 2 8 9 8 8 9 10 8 11 18 11 12 14 15 17
```

Ieșire

```
31
1
```

Explicație

După ordonare se obține șirul: 1 2 2 2 3 6 6 8 8 8 8 9 9 10 11 11 12 14 15 17 18.

Se obține apoi: 1 3 1 2 4 2 1 2 1 1 1 1 1. Deci avem grupurile: (1 3 1), (2 4 2), (1 2 1), (1 1 1) și obținem șirul de valori: 1 1 0 1. Din valorile 1 1 0 1 se obține numărul de control 31, care este număr prim.

Timp de execuție: 1 secundă/test

P030414: Vânătoare

Vânătorul șef al regelui *Arthur* a primit însărcinare să vâneze rațele care se întorc primele din țările calde. Regele *Arthur*, fiind un tip cu idei fixe, i-a cerut vânătorului să vâneze rațele albe cu săgeți albe, iar rațele negre cu săgeți negre.

Rațele vin în rânduri (stoluri) din ce în ce mai mari: mai întâi una, apoi două, trei, cinci, opt, treisprezece, ș.a.m.d.

Se observă că numărul de rațe dintr-un rând este egal cu numărul de rațe de pe cele două rânduri anterioare. Rațele fiind niște creaturi ordonate zboară în rânduri, în care nu vei putea găsi două rațe de aceeași culoare alăturate, fiecare rând începând cu o rață albă.

Vânătorul știe că dacă a doborât o rață, atunci trebuie să le doboare pe toate de pe rândul acesteia, deoarece supraviețuitoarele le vor alerta pe celelalte rațe și acestea nu se vor mai întoarce niciodată, iar vânătorul nostru își va pierde slujba.

Știind că vânătorul a primit *ka* săgeți albe și *kb* săgeți negre, trebuie să determinați câte rânduri de rațe a doborât și câte săgeți de fiecare tip i-au rămas, știind că el vrea să-și păstreze slujba.

Date de intrare
De la tastatură se vor citi numerele *ka* și *kb* în această ordine.

Date de ieșire
Pe ecran se va afișa: numărul de rânduri de rațe doborâte, numărul de săgeți albe rămase și numărul de săgeți negre rămase.

Restricții și precizări
- $1 \le ka, kb \le 2.000.000.000$;
- datele de intrare sunt corecte (nu necesită validare).

Exemplu

Intrare	*Ieșire*
`ka=9`	`4`
`kb=10`	`2`
	`6`

Explicație
Pentru exemplul anterior avem rândurile de rațe:

```
A
A N
A N A
A N A N A
```

unde litera `A` reprezintă o rață albă, iar litera `N` reprezintă o rață neagră.

Timp de execuție: 1 secundă/test

Clasa a VII-a

P030415: Șiruri

Se consideră un vector unidimensional *x* cu *n* componente numere naturale distincte, cel mult egale cu 32 000.

Scrieți un program care să construiască vectorul *y* cu elemente din mulțimea $\{1, 2, \ldots, n\}$ astfel încât oricare ar fi numerele naturale *i*, *j* cu proprietatea că $1 \le i \le n$, $1 \le j \le n$ și $x[i] < x[j]$ să avem $y[i]<y[j]$.

Date de intrare
Fișierul de intrare **SIRURI.IN** va conține pe prima linie un singur număr natural care reprezintă valoarea lui *n*.

Cea de-a doua linie conține *n* numere naturale, separate între ele prin spațiu, care reprezintă elementele vectorului *x*.

Date de ieșire
Fișierul de ieșire **SIRURI.OUT** va conține o singură linie pe care se vor afla componentele vectorului *y* separate printr-un spațiu.

Restricții și precizări
- datele de intrare sunt corecte (nu necesită validare);
- numărul de componente ale vectorului *x* (*n*) este un întreg cuprins între 1 și 100;
- componentele vectorului *x* sunt numere naturale cel mult egale cu 32 000.

Exemplu

SIRURI.IN
```
6
12 3 7 16 10 1
```

SIRURI.OUT
```
5 2 3 6 4 1
```

Timp de execuție: 0,1 secunde/test

P030416: Nu te supăra, frate!

Jocul "Nu te supăra, frate!" se joacă de către doi jucători. Pe o pistă circulară cu *n* căsuțe numerotate de la 1 la *n*, în ordinea dată de sensul acelor de ceasornic, se află înscrise valorile 0, 1 și 10.

Cei doi jucători vor avea fiecare câte un pion și vor porni pe rând, din căsuța 1.

Jocul este început de jucătorul cu numărul 1.

Fiecare jucător va muta pionul său conform cu valoarea obținută prin aruncarea zarului, acumulând sau pierzând puncte în funcție de căsuța în care ajunge.

Când îi vine rândul, fiecare jucător citește valoarea zarului.

Jocul are următoarele reguli:

Câștigătorul poate fi jucătorul care ajunge primul din nou în căsuța cu numărul 1, indiferent de punctaj (exceptând cazul în care are punctaj 0);

În cazul în care șirul aruncărilor cu zarul se termină, înseamnă că jucătorii s-au plictisit și câștigă cel care a acumulat mai multe puncte, iar dacă au punctaje egale câștigă cel care se află în căsuța care are numărul de ordine mai mare.

După aruncarea zarului, jucătorul mută pionul, avansând cu atâtea căsuțe cu câte indică valoarea zarului, în ordinea acelor de ceasornic, începând numărătoarea cu căsuța următoare poziției pe care se află. Prima căsuță nu conține valoarea 0 (zero).

După mutare pot apărea următoarele situații:
- ajunge într-o căsuță cu valoarea 0 (zero) - jucătorul este penalizat, pierde toate punctele acumulate și reia jocul din poziția 1;
- ajunge într-o căsuță cu valoarea 10 - primește un bonus de 10 puncte;
- ajunge într-o căsuță cu valoarea 1 - primește 1 punct;
- ajunge într-o căsuță în care se află celălalt pion (cu excepția căsuței 1, când câștigă) - jucătorul care ajunge ultimul este penalizat, pierde toate punctele și reia jocul de la căsuța 1.

Sarcina voastră este să determinați jucătorul câștigător, pozițiile fiecărui jucător pe cerc și punctajul fiecărui jucător.

Date de intrare

Fișierul de intrare **JOC.IN** conține pe prima linie numărul n al căsuțelor din cerc.

Cea de-a doua linie conține o succesiune de n valori (0, 1 sau 10), separate printr-un spațiu, reprezentând valoarea fiecărei căsuțe.

Cea de-a treia linie conține numărul de aruncări cu zarul.

Cea de-a patra linie conține o succesiune de valori întregi cuprinse între 1 și 6, separate printr-un spațiu, reprezentând aruncarea cu zarul.

Date de ieșire

Fișierul de ieșire **JOC.OUT** va trebui să conțină pe prima linie numărul jucătorului câștigător.

Cea de-a doua linie trebuie să conțină poziția și punctajul jucătorului cu numărul 1.

Cea de-a treia linie trebuie să conțină poziția și punctajul jucătorului cu numărul 2.

Restricții și precizări

- $7 \le n \le 100$;
- datele de intrare sunt corecte (nu necesită validare).

Exemple

JOC.IN
```
10
1 1 1 1 1 10 0 1 1 0
8
3 6 2 4 1 3 5 3
```

JOC.OUT
```
2
6 10
1 3
```

JOC.IN
```
7
1 1 0 10 1 10 1
16
6 4 2 4 6 2 1 3 5 6 3 3 2 1 4 5
```

JOC.OUT
```
1
1 3
1 0
```

JOC.IN
```
12
1 10 1 1 0 10 1 1 1 10 0 1
4
5 6 3 3
```

JOC.OUT
```
2
9 11
10 11
```

Timp de execuție: 0,1 secunde/test

Clasa a VIII-a

P030417: Spre culmi

Dornic de o condiție fizică perfectă, un viitor olimpic național la informatică își propune să escaladeze cea mai înaltă culme a unui un masiv muntos. Se echipează corespunzător, își cumpără un termos, îl umple cu apă, culege informațiile despre traseele existente și completează astfel fișierul de intrare **POPAS.IN**. Pe parcursul fiecărui traseu există mai multe izvoare de la care drumețul își poate umple termosul.

Știind că pe munte este bine să mergi cu pas constant și fără ruperi de ritm, își propune să atingă culmea facând cât mai puține popasuri (pentru umplerea termosului).

Dintre toate traseele existente către culme determinați-l pe cel pentru care numărul total de popasuri este minim. Dacă sunt mai multe astfel de trasee, se va alege cel care este scris ultimul în fișierul de intrare.

Date de intrare

Fișierul de intrare **POPAS.IN** conține pe prima linie un singur număr natural k, reprezentând numărul total de trasee către culme.

Fiecare dintre următoarele k linii conține descrierea câte unui traseu care constă din mai multe numere, separate prin câte un spațiu. Primul număr din descrierea unui traseu reprezintă numărul asociat traseului, fiecare traseu fiind identificat în mod unic printr-un număr natural cuprins între 1 și k. Cel de-al doilea număr reprezintă numărul r al izvoarelor cu apă rece de pe traseu, iar în continuare se află r numere, d_1, d_2, ..., d_r, care reprezintă distanța de la punctul de plecare până la fiecare izvor.

Penultima linie din fișier conține un singur număr natural t, care reprezintă distanța pentru care drumețului îi este suficientă apa din termos, iar ultima linie din fișier reprezintă distanța pe care drumețul o poate străbate fără apă.

Date de ieșire

Fișierul de ieșire **POPAS.OUT** va conține pe aceeași linie, despărțite prin spațiu, două numere care reprezintă numărul minim de popasuri necesare deplasarii, respectiv numărul traseului ales.

Dacă problema nu are soluție fișierul de ieșire va conține cifra 0.

Restricții și precizări

- $0 < k \le 100$;

- $0 < r \le 20$;
- $0 < d_i \le 360$, $\forall 1 \le i \le r$;
- $1 \le t \le 10$;
- $1 \le u \le 5$;
- datele de intrare sunt corecte (nu necesită validare).

Exemple

POPAS.IN

```
3
2 3 12 5 9
1 4 2 9 7 11
3 5 2 16 7 9 8
6
2
```

POPAS.OUT

```
1 1
```

POPAS.IN

```
2
1 3 12 5 9
2 3 2 7 11
1
2
```

POPAS.OUT

```
0
```

Timp de execuție: 1 secundă/test

P030418: Ronul fermecat

Lidorienii și *senopictii* sunt în conflict pentru *ronul fermecat*, fiind arbitrați de orintieni, aleși de părțile beligerante drept judecători. *Orintia* a propus ca:

ronul fermecat să fie ascuns printre alți *k roni* cu același aspect, dar toți realizați dintr-un material mai greu decât originalul, având o masă standard, diferită de cea a ronului femecat.

Pentru a-l descoperi, vă gândiți că aveți la dispoziție o balanță și toți cei $k + 1$ *roni*. *Lidorienii*, apoi *senopictii* vor spune un singur număr, reprezentând numărul maxim de cântăriri admis pentru descoperirea ronului fermecat.

Dacă nici una dintre părți nu descoperă numărul corect, atunci *ronul fermecat* va rămâne în *Orintia*. Dacă ambele părți descoperă numărul corect, ronul va rămâne tot la *orintieni*.

Sarcina voastră este să indicați țara care câștigă *ronul fermecat*: *Lidoria*, *Senopictia* sau *Orintia*, identificate prin literele: *L*, *S*, respectiv *O*.

Date de intrare

Fișierul de intrare **RON.IN** conține pe prima linie numărul *k*, iar cea de-a doua linie conține numerele *RL* și *RS*, separate printr-un singur spațiu, care reprezintă răspunsul *lidorienilor*, respectiv răspunsul *senopictilor*.

Date de ieșire

Fișierul de ieșire **RON.OUT** trebuie să conțină una dintre literele *L*, *S*, și *O*.

Litera *L* indică faptul că *Lidoria* a câștigat *ronul fermecat*. Litera *S* indică faptul că *Senopictia* a câștigat *ronul fermecat*, iar litera *O* indică faptul că *Orintia* a câștigat *ronul fermecat*.

Restricții și precizări

- $1 < K < 10000$;
- $1 \le RL, RS \le K$;
- *ron fermecat* reprezintă un cuboid gravat cu semnele fixe ale puterii;
- numărul maxim de cântăriri admis nu se obține cântărind un *ron* de mai multe ori și nici cântărind de cât mai multe ori *ronii*; cântărirea presupune să existe, pe fiecare braț al balanței, un număr egal de *roni* (1-1, 2-2, etc.)

Exemplu

RON.IN

```
7
1 3
```

RON.OUT

```
O
```

RON.IN

```
4
2 2
```

RON.OUT

```
O
```

Explicație

În cadrul primului exemplu, numărul maxim de cântăriri admis este 2, deci *ronul fermecat* rămâne în *Orintia*.

În cadrul celui de-al doilea exemplu, numărul maxim de cântăriri admis este 2, dar fiind egalitate, *ronul fermecat* rămâne în *Orintia*.

Timp de execuție: 1 secundă/test

Despre OJI...

La ediția din aceast an a *Olimpiadei Județene de Informatică* au participat elevi din majoritatea școlilor generale și liceelor cu profil informatică.

În urma relatărilor participanților la acest concurs redacția *GInfo* a constatat că foarte mulți concurenți au fost *distrați* și nu au înțeles bine enunțurile unor probleme și astfel au considerat că ele sunt *complicate*.

O altă categorie de participanți au repetat aceleași greșeli ca în alți ani și au fost dezamăgiți când au observat punctajul pe care l-au obținut deși, în urma modificării unui singur caracter din program ar fi obținut punctajul maxim.

În general problemele au fost accesibile concurenților, obținându-se și punctaje maxime la toate.