

## PROBLEMA 1

100 puncte

## CONCURS

La un concurs de programare s-au înscris  $n$  elevi. Concursul se desfășoară pe două secțiuni, secțiunea 1 pentru începători și secțiunea 2 pentru avansați.

Proba de concurs se desfășoară pe parcursul a 3 ore și elevii au de rezolvat 2 probleme. Fiecare problemă poate avea punctajul minim de 0 puncte și punctajul maxim de 100 de puncte. Punctajul final al concurentului este format din suma punctajelor celor două probleme.

## Cerințe

Să se scrie un program care citește numărul de elevi înscriși și apoi date despre fiecare elev înscris (secțiunea la care s-a înscris, punctajul obținut pentru prima problemă și punctajul obținut pentru a doua problemă) și rezolvă următoarele cerințe:

1. Mesajul „DA” dacă toți cei  $N$  elevi înscriși au reușit să obțină un punctaj nenul la ambele probleme propuse, respectiv „NU” în caz contrar.
2. Afișați pentru fiecare secțiune numărul de elevi înscriși. Afișarea se va face în ordinea crescătoare a numărului secțiunii, prin perechi de numere de forma „nr\_secțiune nr\_elevi” .
3. Afișați pentru fiecare secțiune punctajul maxim obținut și numărul de elevi care au obținut acest punctaj. Afișarea se va face în ordinea crescătoare a numărului secțiunii, prin triplete de numere de forma „nr\_secțiune punctaj\_maxim nr\_elevi”. Știind ca premiile se acordă doar celor care au luat punctaj maxim, afișați și numărul de premii care vor fi acordate.

## Date de intrare

Fișierul de intrare **concurs.in** conține pe prima linie un număr natural  $C$ . Pentru toate testele de intrare, numărul  $C$  are una din valorile 1,2 sau 3. Pe linia a doua a fișierului se găsește numărul natural  $N$  - numărul de elevi înscriși iar pe următoarele  $N$  linii, separate prin spațiu, trei numere  $S$   $P1$   $P2$  cu semnificația  $S$  secțiunea la care participă elevul,  $P1$  punctajul obținut la prima problemă și  $P2$  punctajul obținut la a doua problemă.

## Date de ieșire

Dacă  $C=1$ , se va rezolva numai punctul 1). În acest caz, în fișierul de ieșire **concurs.out** se va scrie un singur mesaj „DA” sau „NU”.

Dacă  $C=2$ , se va rezolva numai punctul 2). În acest caz, în fișierul de ieșire **concurs.out** se vor scrie 2 linii, pe fiecare linie o pereche de numere, separate prin spațiu reprezentând valorile cerinței 2

Dacă  $C=3$ , se va rezolva numai punctul 3). În acest caz, în fișierul de ieșire **concurs.out** se vor scrie 3 linii, pe primele două linii câte un triplet de numere separate prin spațiu reprezentând valorile cerinței 3 iar pe ultima linie numărul de premii.

## Restricții și precizări

- $1 \leq C \leq 3$ ,  $1 \leq S \leq 2$ ,  $s$  reprezintă secțiunea
- $0 < n \leq 300$
- $0 \leq P1, P2 \leq 100$

## Exemple

concurs.in	concurs.out	Explicații
1 7 1 100 100 2 100 100 1 50 0 1 100 100 2 0 40 1 100 100 2 30 70	NU	- Există elevi care au punctajul 0 la probleme
2 7 1 100 100 2 100 100 1 50 0 1 100 100 2 0 40 1 100 100 2 30 70	1 4 2 3	- La secțiunea 1 sunt 4 inscriși, - La secțiunea 2 sunt 3 inscriși
3 7 1 100 100 2 100 100 1 50 0 1 100 100 2 0 40 1 100 100 2 30 70	1 200 3 2 200 1 4	- La secțiunea 1 punctajul maxim (200) a fost obținut de 3 elevi, - La secțiunea 2 punctajul maxim 200 a fost obținut de 1 elev - Număr de premii 4

**Timp 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 2

100 puncte

## BALAURI

În livada Împăratului Verde, vin la furat de mere aurii, balauri. Făt Frumos s-a hotărât să-l scape pe împărat de aceștia și îi poate alunga dacă taie capetele celor mai puternici doi balauri.

Un balaur poate avea unul sau cel mult 9 capete, de dimensiuni diferite. Dimensiunea unui cap de balaur este între 1 și 9 m. Făt Frumos și-a notat dimensiunile capetelor fiecărui balaur și a obținut un șir de numere (exemplu: **257** înseamnă că balaurul are un cap de dimensiune **2** m, apoi un cap de dimensiune **5** m și un cap de dimensiune **7** m).

Cei mai puternici doi balauri sunt considerați:

- **Balaurul echilibrat**, adică un număr palindrom care nu are aspect de munte;

Ploiești, 28 mai 2016

- se spune că un număr este de tip munte dacă are cifrele din stânga cifrei maxime în ordine strict crescătoare, iar cele din dreapta cifrei maxime în ordine strict descrescătoare. Cifra maximă apare o singură dată în numărul munte și nu este nici prima, nici ultima cifră a sa (exemplu: **1623261** este un balaur echilibrat, deoarece este număr palindrom fără aspect de munte, **12421** nu este balaur echilibrat);
- dacă există mai mulți, cel mai puternic este acela care formează cel mai mare număr palindrom;
- **Balaurul ne-echilibrat**, adică un număr cu aspect de munte care nu este palindrom (exemplu: **12463** este un balaur ne-echilibrat; **14263** nu este balaur ne-echilibrat).
  - dacă există mai mulți, cel mai puternic este acela care are suma capetelor maximă (exemplu: balaurul **12463** este mai puternic decât balaurul **12561**);

**Cerință**

Fiind dat numărul **n** de balauri și apoi cele **n** numere codificate după dimensiunea capetelor, să se determine cei mai puternici doi balauri, **balaurul echilibrat** și **balaurul ne-echilibrat**.

**Date de intrare**

Din fișierul **balauri.in** se citesc:

- de pe prima linie, numărul natural **n**;
- de pe a doua linie, cele **n** numere naturale, fiecare având maxim nouă cifre, separate prin câte un spațiu.

**Date de ieșire**

Fișierul **balauri.out** va conține:

- pe prima linie, un număr natural corespunzător **balaurului echilibrat**;
- pe a doua linie, un număr natural corespunzător **balaurului ne-echilibrat**.

**Restricții și precizări:**

- $2 \leq n \leq 1000$
- toate cele **n** numere naturale de pe a doua linie au maxim nouă cifre diferite de cifra **0**.
- există cel puțin un balaur echilibrat și un balaur ne-echilibrat.

**Exemplu**

balauri.in	balauri.out	Explicații
5 1231 12321 313 231 525	525 231	525 este cel mai număr palindrom care nu are aspect de munte; 231 este număr cu aspect de munte care nu este palindrom și are suma capetelor maximă (și 1231 are aspect de munte și nu este palindrom dar suma capetelor este 2);

**Timp 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 3

100 puncte

## UNGHI

La geometrie, domnul profesor de matematică le-a vorbit elevilor săi despre unghiuri. Pentru a fi sigur că aceștia au înțeles noțiunile predate, le-a dat o listă cu  $n$  probleme.

**Cerință:** Fiind date numărul de ore în variabila  $h$  și numărul de minute în variabila  $m$ , să se determine câte grade avea unghiul dintre orarul și minutarul unui ceas clasic, la ora  $h$  și  $m$  minute?

## Date de intrare

Fișierul unghi.in va conține pe prima linie un număr natural  $n$ , reprezentând numărul de probleme date ca temă, iar pe a doua linie, două numere  $h$  și  $m$ , separate printr-un spațiu, cu semnificațiile din cerință.

## Date de ieșire

Fișierul unghi.out va conține pe câte o linie numărul de grade corespunzător unghiului format de orar și minutar, pentru fiecare dintre cele  $n$  probleme din temă.

## Restricții și precizări


$n$  este număr natural,  $1 \leq n \leq 100$

$h$  este număr natural,  $0 \leq h < 24$

$m$  este număr natural,  $0 \leq m \leq 60$

Dimensiunea unui unghi se măsoară în grade. Un cerc are 360 de grade.

## Exemplu:

unghi.in	unghi.out	Explicații
3 8 20 1 10 5 15	130 25 67.5	 <p>Așa cum se vede și în figura alăturată unghiul dintre orar și minutar este de 130 grade pentru prima pereche de valori din fișierul de intrare. Asemănător se obțin și celelalte valori corespunzătoare următoarelor perechi.</p>

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

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

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