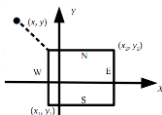


### 1. PLUTĂ

În mijlocul lacului plutește o plută, care are forma unui dreptunghi. Părțile laterale ale plutei sunt direcționate de-a lungul paralelelor și meridianelor. Introducem un sistem de coordonate în care axa OX este direcționată către est, iar axa OY spre nord. Fie colțul sud-vest al plutei are coordonatele  $(x1, y1)$ , colțul nord-est – coordonatele  $(x2, y2)$ . Înătorul se află în punctul cu coordonatele  $(x, y)$ .



**Sarcină.** Elaborați un program, care determinăm ce parte a plutei (nord, sud, vest sau est) sau la care colț al plutei (nord-vest, nord-vest, sud-vest, sud-est) înătorul trebuie să înoate pentru a ajunge cât mai curând posibil.

**Date de intrare.** Fișierul text PLUTA.IN conține trei linii în fiecare câte două numere întregi:  $x1, y1$  (coordonatele colțului sud-vest al plutei),  $x2, y2$  (coordonatele colțului nord-est al plutei),  $x, y$  (coordonatele înătorului).

**Date de ieșire.** Fișierul text PLUTA.OUT va conține pe prima linie partea a plutei sau colțul ei spre care trebuie să înoate înătorul: dacă înătorul ar trebui să înoate spre partea de nord a plutei, fișierul va conține simbolul "N", spre sud - simbolul "S", spre vest - simbolul "W" spre est - simbolul "E"; dacă înătorul ar trebui să înoate spre colțul plutei, în fișier se scrie unul dintre următoarele șiruri: "NW", "NE", "SW", "SE".

#### Restricții:

- $|x1| \leq 100, |x2| \leq 100, |y1| \leq 100, |y2| \leq 100, |x| \leq 100, |y| \leq 100, x1, x2, y1, y2, x, y \in \mathbb{Z}$ ;
  - $x1 < x2, y1 < y2, x \neq x1, x \neq x2, y \neq y1, y \neq y2$ , coordonatele înătorului sunt în afara plutei;
  - Timpul de execuție nu va depăși 1 sec;
  - Programul va folosi cel mult 32 MB de memorie.
- Fișierul sursă va avea denumirea xxx1.PAS, xxx1.C sau xxx1.CPP, unde xxx fiind codul de înregistrare a elevului format din litera N (Z) și 2 cifre. De exemplu, N011.PAS.

#### Exemplu.

PLUTA.IN	PLUTA.OUT
-1 -2	NW
5 3	
-4 6	

### 2. ORARUL CERCULUI

Vlad e pasionat de rezolvarea problemelor pentru olimpiadă la disciplina Informatica. Datorită acestui fapt el a decis să frecventeze cercul de informatică. El a aflat că orele se vor promova săptămânal în aceeași zi a săptămânii în perioada 1 septembrie – 31 martie ale unui an școlar (se consideră că vacanțele și sărbătorile lipsesc, iar lecția poate fi desfășurată în orice zi a săptămânii).

**Sarcină.** Elaborați un program, care alcătuiește orarul lecțiilor până la sfârșitul perioadei indicate, știind data primei ore frecventate de Vlad.

**Date de intrare.** Fișierul text CERC.IN conține trei numere întregi pozitive:  $an$  – numărul anului,  $l$  – numărul lunii,  $zi$  – numărul zilei din lună pentru prima lecție frecventată în cadrul cercului, fiecare din rând nou.

**Date de ieșire.** Fișierul text CERC.OUT va conține în fiecare linie câte trei numere întregi pozitive separate prin spațiu în următoarea ordine: numărul anului, numărul lunii și numărul zilei, pentru fiecare lecție a cercului până la expirarea perioadei de promovare în ordinea cronologică.

#### Restricții:

- $an \geq 2018, an \in \mathbb{N}$ ;
  - $l \in \{1, 2, 3, 9, 10, 11, 12\}, l \in \mathbb{N}$ ;
  - $zi \in 1..30$  pentru  $l \in \{9, 11\}$ ;  $zi \in 1..31$  pentru  $l \in \{10, 12, 1, 3\}$ ;  $zi = 28$  (nu este an bisect) sau  $zi = 29$  (an bisect) pentru  $l \in \{2\}$ ;
  - Timpul de execuție nu va depăși 2 sec.;
  - Programul va folosi cel mult 32 MB de memorie.
- Fișierul sursă va avea denumirea xxx2.PAS, xxx2.C sau xxx2.CPP, unde xxx fiind codul de înregistrare a elevului format din litera N (Z) și 2 cifre. De exemplu, N012.PAS.

#### Exemplu.

CERC.IN	CERC.OUT
2018	2018 2 20
2	2018 2 27
20	2018 3 6
	2018 3 13
	2018 3 20
	2018 3 27

### 3. PÂRJOALE DE LA BUCĂTARUL ȘEF

Bucătarul taberei „Andrieș” a instituit „sărbătoarea Stomacului”. Pentru a marca sărbătoarea el a pregătit  $K$  pârjoale foarte delicioase și a stabilit (în taină) că primul, care va veni la masă va primi cele mai multe pârjoale, iar fiecare din următorii veniți un număr strict mai mic decât precedentul.

În dependență de modul de împărțire, pârjoale vor ajunge unui număr diferit de persoane. De exemplu 6 pârjoale pot fi distribuite după 4 scheme diferite:  $3+2+1, 4+2, 5+1$  și 6.

**Sarcină.** Elaborați un program, care determinăm câte moduri pot fi distribuite cele  $K$  pârjoale, respectând criteriul bucătarului.

**Date de intrare.** Fișierul text BUCAT.IN va conține un singur număr întreg pozitiv  $K$  – numărul de pârjoale.

**Date de ieșire.** Fișierul text BUCAT.OUT va conține un număr întreg pozitiv – cel al modurilor în care pot fi distribuite pârjoalele.

#### Restricții:

- $(0 \leq K \leq 225), K \in \mathbb{N}$ ;
- Timpul de execuție nu va depăși 3 sec.;
- Programul va folosi cel mult 32 MB de memorie.

Fișierul sursă va avea denumirea xxx3.PAS, xxx3.C sau xxx3.CPP, unde xxx fiind codul de înregistrare a elevului format din litera N (Z) și 2 cifre. De exemplu, N013.PAS.

#### Exemplu.

BUCAT.IN	BUCAT.OUT
6	4

### 4. IMPOZIT TURISTIC

Pentru ajustarea bugetului în țara Avalon, cunoscută prin rutele sale turistice montane, a fost introdus un nou impozit pentru turiști. Mărimea impozitului este direct proporțională cu distanța rutei, dar, deoarece ruta trece prin munți și distanța parcursă, care depinde de urcări și coborări e dificil de calculat, impozitul se calculează fără a lua în considerație înălțimea, adică valoarea impozitului este proporțională cu parcurgerea orizontală pe care o realizează turiștii. Conform unei vechi tradiții grupurile turistice trebuie să se deplaseze prin munții Avalon strict de la vest spre est.

Agenția turistică dorește să economisească pe impozit, de aceea dorește să elaboreze ruta turistică cu impozit minim. Ținând cont că rutele trec prin munți, acestea trebuie să conțină urcare în vârf și coborâre din acesta, adică pe rută trebuie să fie un punct care se află strict mai sus decât începutul și sfârșitul traseului. Agenția turistică a alcătuit harta munților Avalon, care conține informația despre înălțimea munților pe traseul parcurs de la vest spre est. Înălțimile munților sunt măsurate la puncte echidistante.

**Sarcină.** Elaborați un program, care determină după harta munților Avalon ruta turistică cu traseul de distanță minimă, respectând condiția prezenței urcărilor și coborârilor.

**Date de intrare.** Fișierul text RUTA.IN conține pe prima linie un număr întreg pozitiv  $N$  – numărul de puncte pe harta munților Avalon. Următoarele  $N$  rânduri conțin câte un număr întreg pozitiv  $H_i$ , ce reprezintă informația despre înălțimea munților în fiecare din aceste puncte la deplasarea în direcția de la vest spre est.

**Date de ieșire.** În cazul existenței traseului ce satisface condițiile, fișierul text RUTA.OUT va conține două numere întregi pozitive: numărul de ordine al punctului ce reprezintă începutul și numărul de ordine al punctului ce reprezintă sfârșitul traseului cu impozit minim, fiecare din rând nou. Punctele se numerotează de la 1 la  $N$ . În cazul existenței câtorva trasee ce satisfac condiția se vor fixa datele la primul din ele. Dacă nu există traseu care respectă condițiile, în fișier se va înscrie doar o singură valoare 0.

#### Restricții:

- $N \leq 10^5, N \in \mathbb{N}$ ;
- $i \in 1 \dots N, i \in \mathbb{N}$
- $H_i \leq 10^5, H_i \in \mathbb{N}$ ;
- Timpul de execuție nu va depăși 5 sec.;
- Programul va folosi cel mult 32 MB de memorie.

Fișierul sursă va avea denumirea xxx4.PAS, xxx4.C sau xxx4.CPP, unde xxx fiind codul de înregistrare a elevului format din litera N (Z) și 2 cifre. De exemplu, N014.PAS.

#### Exemplu.

RUTA.IN	RUTA.OUT	Remarcă
7	3	Sunt date 7 puncte cu înălțimea 18, 10, 15, 20, 20, 10, 3. Cel mai scurt traseu, care conține urcare și coborâre este: 15, 20, 20, 10. El are început în punctul cu numărul de ordine 3 și sfârșit în punctul cu numărul de ordine 6.
18	6	
10		
15		
20		
20		
10		
10		
3		
3	0	
9		
8		
5		