



Concursul de Informatică

"ȘTEFAN ODOBLEJA"

În continuare vă vom prezenta enunțurile celor nouă probleme propuse spre rezolvare la Concursul de informatică "Ștefan Odoobleja" organizat în primăvara anului trecut la Craiova. Problemele prezentate mai jos sunt atât cele pentru proba individuală, cât și cele pentru proba pe echipe. Redacția Glnfo a modificat ușor enunțurile pentru a oferi cititorilor enunțuri mai concise.

Clasa a IX-a

P010407: Dilema furnicuțelor

În curtea din spatele casei unui gospodar trăiau nestingherite mai multe familii de furnicuțe, organizate în mai multe mușuroaie.

Una dintre furnicuțe îl aude într-o zi pe gospodar spunându-i soției sale că vrea să amplaseze un obiect chiar în zona în care se aflau mușuroaiele.

Furnicuța le anunță și pe celelalte, însă neștiind să aprecieze exact unde va fi amplasat acel obiect, furnicuțele au nevoie de ajutorul vostru. Dacă află care din mușuroaie se află în pericol, ele se pot adăposti fără grijă în cele care sunt în siguranță.

Cunoscându-se numărul de puncte (mușuroaie) M , coordonatele x și y ale celor M puncte prin numărul vârfurilor poligonului (obiectului) și coordonatele vârfurilor, determinați mușuroaiele care sunt în pericol să fie distruse de obiectul care va fi amplasat de gospodar.

Date de intrare

Fișierul de intrare `furnici.in` conține pe prima linie un număr întreg pozitiv M , care reprezintă numărul de mușuroaie.

Pe următoarele M linii se găsesc câte două numere întregi pozitive, separate printr-un singur spațiu, care reprezintă coordonatele punctelor (mușuroaielor).

Următoarea linie conține un singur număr întreg pozitiv N care reprezintă numărul de vârfuri ale poligonului.

Cele N linii care urmează conțin câte două numere întregi pozitive, separate prin câte un spațiu, care reprezintă coordonatele pentru vârfurile poligonului. Poligonul poate fi convex sau concav.

Date de ieșire

Fișierul de ieșire `furnici.out` va conține M linii. Pe fiecare dintre acestea se va scrie unul dintre mesajele "în siguranță!" sau "în pericol" dacă mușuroaiele de furnici din fișierul de intrare se găsesc în exteriorul, respectiv în interiorul poligonului.

Punctele corespunzătoare mușuroaielor vor fi tratate în ordinea în care apar în fișierul de intrare.

Restricții și precizări

- se consideră că un mușuroi este în pericol atunci când coordonatele sale sunt conținute în interiorul poligonului sau când mușuroiul se găsește pe una dintre laturile acestuia;
- $3 \leq N \leq 50$;
- $1 \leq M \leq 100$.

Exemplu

`furnici.in`

```
5
80 100
400 170
230 160
340 347
230 350
6
100 165
380 200
400 140
520 350
395 310
220 360
```



furnici.out
in siguranta!
in pericol!
in siguranta!
in siguranta!
in pericol

P010408: Verde la semafor

Lucrând în tura de noapte, *Ovidiu* pleacă din parcare la ora 02:00 AM fix, în fiecare dimineață.

Drumul său spre casă este drept și de-a lungul lui sunt plasate unul sau mai multe semafoare. *Ovidiu* s-a întrebat adesea dacă, dându-se locațiile exacte ale semafoarelor și perioadele de timp la care se schimbă culoarea acestora, există viteze cu care să meargă acasă fără să fie nevoit să accelereze sau să încetinească în funcție de culoarea roșie a semaforului.

Pentru a-și satisface curiozitatea, el vă roagă să faceți un program care să calculeze aceste viteze.

Programul ar trebui să găsească toate vitezele (numere întregi, pozitive), în kilometri pe oră, pe care să le urmeze *Ovidiu* în drumul său spre casă.

Fiecare viteză (în kilometri pe oră) trebuie menținută de el din momentul în care pleacă din parcare la ora 02:00 AM până ajunge acasă, astfel încât el să nu treacă niciodată pe culoarea roșie a semaforului.

Ovidiu poate trece de un semafor chiar în momentul în care acesta își schimbă culoarea din galben în roșu, sau chiar în momentul în care roșu se transformă în verde.

Datorită faptului că *Ovidiu* este un cetățean care nu vrea să încalce prea mult legea, trebuie considerate numai vitezele mai mici sau egale cu 60 km/h. El nu vrea însă nici să meargă foarte încet, de aceea nu trebuie considerate nici vitezele mai mici de 30 km/h.

Ovidiu a aflat dintr-o sursă sigură faptul că toate semafoarele se schimbă în verde la ora 02:00 AM.

Date de intrare

Fișierul de intrare **semafoare.in** conține una sau mai multe mulțimi de date care descriu semnale de trafic. Lista de mulțimi este urmată de valoarea -1 pe o singură linie.

O mulțime este dată pe mai multe linii. Astfel, prima linie a unei mulțimi conține un singur număr natural N care reprezintă numărul de semafoare de pe traseu.

Pe următoarele N linii corespunzătoare unei mulțimi se află patru numere reale pozitive L , V , G și R separate între ele prin spații. L reprezintă distanța la care este plasat un semafor, în kilometri, față de parcare. V , G și R sunt perioadele de timp exprimate în secunde până la schimbarea culorilor *verde*, *galben*, respectiv *roșu* ale semafoarelor.

Date de ieșire

Fișierul de ieșire **semafoare.out** constă din tratarea fiecărei mulțimi în parte, urmată de o listă a tuturor vitezelor care sunt potrivite pentru ca *Ovidiu* să evite culoarea roșie.

Vitezele, care sunt consecutive, ar trebui să fie specificate ca intervale de forma $m-M$, unde m și M sunt vitezele cele mai mici, respectiv cele mai mari din interval. Intervalele de forma $M-M$ (adică de lungime 1) trebuie scrise doar ca M . Intervalele trebuie să fie separate prin virgule.

Dacă nu există viteze care să îndeplinească cerințele, în fișierul de ieșire se va scrie "Nu exista nici o viteză".

Pentru fiecare mulțime din fișierul de intrare, în fișierul de ieșire se va scrie o singură linie. La începutul unei linii se va scrie mesajul "Cazul X:", unde X este numărul de ordine al mulțimii din fișierul de intrare. Acest mesaj este urmat de un spațiu și fie de mesajul "Nu exista nici o viteză", fie de o listă de viteze separate prin caracterul ", " așa cum indică exemplul prezentat în continuare.

Restricții și precizări

- numărul N de semafoare are cel mult valoarea 6;
- distanța unui semafor față de fabrică este un număr real și nu va depăși valoarea 32000;
- intervalul de timp care se scurge de la aprinderea unei culori a unui semafor și până la schimbarea acesteia este dat de un număr întreg a cărui valoare este cuprinsă între 1 și 256;
- numărul de mulțimi din fișierul de intrare nu va depăși valoarea 100.

Exemplu

semafoare.in

```
1
5.5 40 8 25
3
10.7 10 2 75
12.5 12 5 57
17.9315 4 67
-1
```

semafoare.out

Cazul 1: 30, 32-33, 36-38, 41-45, 48-54, 59-60
Cazul 2: Nu exista nici o viteză.

Clasa a X-a

P010409: Traectoria bilei

Se consideră o masă dreptunghiulară de dimensiune $M \times N$ (asemănătoare cu cea de biliard), cu o gaură în interior.

Pe masă se află o bilă (se va considera de dimensiune neglijabilă), care se deplasează în interiorul acestei mese și care se poate lovi de marginile acesteia sau poate cădea în gaura care se află pe masă după cum se poate observa în figura 1.

Ciocnirile cu manta (marginile mesei) se fac cu un unghi de incidență egal cu unghiul de reflexie.

Traectoria pornește dintr-un punct inițial dat (x, y) . Se precizează și direcția în care aceasta pornește. Traectoria se calculează până când s-a efectuat un număr de ciocniri dat sau până când bila cade în gaură.

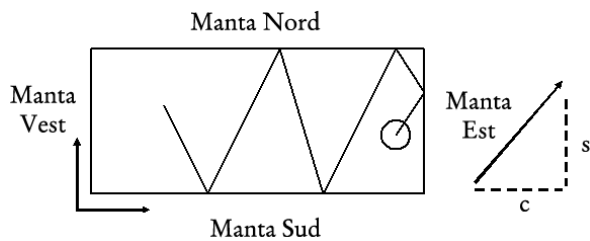


Figura 1

Să se calculeze traiectoria bilei care se deplasează în interiorul acestei mese.

Date de intrare

Fișierul de intrare **BILA.IN** va conține pe prima linie numărul de ciocniri pentru care se calculează traiectoria.

Pe cea de-a doua linie se află două numere reale M și N , separate printr-un singur spațiu, care reprezintă dimensiunile mesei.

Cea de-a treia linie conține trei numere separate între ele prin spații care reprezintă coordonatele centrului găurii circulare și raza acesteia.

Cea de-a patra linie conține patru numere separate între ele printr-un singur spațiu, care indică poziția și direcția inițiale ale bilei.

Direcția inițială a bilei este dată de două numere (c , s), care sunt lungimile proiecțiilor unei săgeți de lungime 1 (de-a lungul direcției de mișcare a bilei) pe axele Ox și Oy . Cu alte cuvinte, c și s sunt *cosinusul* și *sinusul* unghiului măsurat în sens invers acelor de ceasornic de pe direcția $+Ox$. Dacă aceste proiecții sunt în direcțiile $-Ox$ sau $-Oy$, numerele respective sunt negative.

Trebuie tratat și cazul în care bila ajunge într-un colț, situație în care se întoarce pe aceeași linie pe care a venit.

Date de ieșire

În fișierul de ieșire **BILA.OUT** se va scrie câte o linie pentru fiecare ciocnire de pe traiectorie, identificând numărul ciocnirii, manta (*Nord*, *Sud*, *Est*, *Vest*) cu care s-a ciocnit bila și coordonatele de-a lungul mantei ale punctului de ciocnire așa cum apare în exemplul prezentat în continuare.

Dacă ciocnirea are loc într-un colț, se va afișa pe o linie, despre care colț este vorba (*SudEst*, *SudVest*, *NordEst*, *NordVest*). Spre exemplu, dacă la a treia ciocnire cu bila a ajuns în colțul *NordEst* se va afișa: "ciocnirea 3 în colțul NordEst".

Liniile traiectoriei se vor genera până când se atinge numărul maxim de ciocniri sau până în momentul în care bila cade în gaură.

Bila cade în gaură dacă traiectoria calculată intersectează cercul care reprezintă suprafața găurii. În acest caz în fișierul de ieșire se va scrie pe o singură linie mesajul "Bila a cazut in gaura!".

Restricții și precizări

- numărul de ciocniri cu manta este un număr întreg pozitiv cu valoarea cuprinsă între 1 și 100;

- raza găurii este un număr întreg pozitiv a cărui valoare este cuprinsă între 1 și 100;
- $4 \leq N, M \leq 1.000$.

Exemplu

BILA.IN

```
12
100 60
70 40 10
14 35 0.6 0.8
```

BILA.OUT

```
ciocnirea 1 la zidul Nord x = 32.7500
ciocnirea 2 la zidul Sud x = 77.7500
ciocnirea 3 la zidul Est y = 29.6667
ciocnirea 4 la zidul Nord x = 77.2500
Bila a cazut in gaura!
```

Timp de execuție: 1 secundă/test

P010410: Comoara

Un grup de arheologi format din K persoane au găsit o comoară compusă din M lingouri de aur pe care doresc să le împartă fără să le sectioneze.

Există mai multe posibilități de împărțire a acestora.

Există posibilitatea ca la împărțire unele persoane să nu primească nici un lingou.

La rugămintea arheologilor, sarcina voastră este să găsiți numărul total de modalități de a împărți comoara, astfel încât, la fiecare împărțire, fiecare persoană să primească un număr diferit de lingouri.

Date de intrare

Fișierul de intrare **comoara.in** conține pe o singură linie două numere întregi M și K , separate printr-un singur spațiu, reprezentând numărul de lingouri, respectiv numărul de persoane.

Date de ieșire

Fișierul text **comoara.out** conține pe o singură linie numărul de modalități de a împărți comoara astfel încât fiecare persoană să primească un număr diferit de lingouri la fiecare împărțire.

Restricții

- $1 \leq M \leq 1.000$;
- $1 \leq K \leq 50$.

Exemplu

comoara.in

```
20 4
```

comoara.out

```
11
```

Timp de execuție: 1 secundă/test





Clasele a XI-a și a XII-a

P010411: Locuințe

În ținutul *Cucuberland* populația este în continuă creștere. Suprafața ținutului este destul de mică, foarte curând putându-se manifesta o criză a locuințelor. Pentru aceasta, primăria a luat decizia să construiască locuințe într-o regiune mlăștinoasă, întinsă ca suprafață, dar inutilizabilă din cauza prezenței apei. Mlaștina are forma unui pătrat cu latura de lungime N metri.

Deoarece costul asanării mlaștinii era prea mare, arhitecții au propus amplasarea unor piloni, pe aceștia urmând a se construi locuințe, fără a mai fi necesară asanarea mlaștinii. Pilonii vor fi amplasați într-o rețea rectangulară la distanța de 1 m pe verticală sau orizontală. În figura următoare este prezentată o rețea de piloni pentru o mlaștină cu latura de 2 metri.

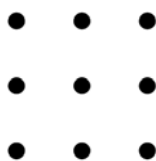


Figura 2

Se știe că o locuință are, de asemenea o formă pătratică, iar pentru a putea fi amplasată, toate cele patru colțuri ale sale trebuie să se sprijine pe câte un pilon.

Pentru a putea aproba planul de urbanism, primarul trebuie să cunoască numărul total de modalități în care poate fi amplasată o locuință pătrată în zona mlăștinoasă respectivă.

Date de intrare

Pe prima linie a fișierului de intrare `locuinte.in` se află un singur număr natural N reprezentând latura suprafeței mlăștinoase.

Date de ieșire

Fișierul de ieșire `locuinte.out` va conține un număr natural corespunzător numărului de modalități în care poate fi amplasată o locuință de formă pătrată.

Restricții și precizări

- $1 \leq N \leq 1.000.000.000$;
- lungimea laturii locuinței este un număr întreg cu valoarea cuprinsă între 1 și N .

Exemplu

`locuinte.in`
2

`locuinte.out`
6

Timp de execuție: 1 secundă/test

P010412: Vama

Se știe că vama *Nădlac* este unul dintre punctele de trecere a celei mai aglomerate frontiere din țară, traficul de marfă și călători crescând de la an la an.

Pentru ca un vehicul care vine din *România* să poată pătrunde pe teritoriul *Ungariei* el trebuie să fie vămuit mai întâi de către vameșii români și apoi de către vameșii maghiari, în această ordine. În vama română, într-un schimb se află M lucrători, iar în vama ungară P lucrători.

Un vehicul, odată ajuns în vamă, așteaptă să fie preluat de către un vameș român. După ce trece de vama română, acesta va trebui să parcurgă aceleași operații în vama ungară.

Un vameș controlează la un moment dat cel mult un vehicul. Nu toți vameșii lucrează în același ritm.

Un vameș controlează o mașină indiferent de tipul sau încărcătura acesteia în aceeași perioadă de timp cunoscută.

La miezul nopții în vama română se află un număr N de vehicule care așteaptă să ajungă pe teritoriul *Ungariei*.

Pentru fluidizarea traficului în vamă calculați timpul minim necesar pentru ca toate cele N vehicule să fie vămuite la punctul de trecere al frontierei *Nădlac*.

Date de intrare

Pe prima linie a fișierului de intrare `vama.in` se află, pe o singură linie, trei numere întregi N , M și P , unde N reprezintă numărul de vehicule care se află în așteptare la miezul nopții în vamă, M reprezintă numărul de lucrători din vama română, iar P reprezintă numărul de vameși din vama ungară. Linia a doua conține M numere întregi, separate prin câte un spațiu, care reprezintă timpii de verificare al unui vehicul corespunzător fiecărui vameș român în parte. Pe cea de-a treia linie a fișierului de ieșire sunt scrise P numere întregi, separate prin câte un spațiu, care reprezintă timpii de verificare al unui vehicul corespunzător fiecărui vameș ungar în parte.

Date de ieșire

Fișierul de ieșire `vama.out` va conține un număr întreg corespunzător timpului minim de vămuire a tuturor celor N vehicule.

Restricții și precizări

- $1 \leq N \leq 1.000$;
- $1 \leq M \leq 100$;
- $1 \leq P \leq 100$.

Exemplu

`vama.in`
6 3 2
1 1 10
2 3

`vama.out`
9

Timp de execuție: 1 secundă/test

P010413: Mobuto

O bună parte din suprafața țării *Mobuto* este acoperită de junglă. Pentru a se mișca eficient prin junglă dintr-o localitate în alta, băștinașii se deplasează în "șir indian".

Din localitatea *Mbengdu* pleacă spre localitatea *Mulato* o delegație de N persoane. În dimineața celei de-a noua zile, șeful delegației observă că membri acesteia suferă de "stresul vecinului din față": fiecare s-a săturat să tot vadă aceeași ceafă roșie.

Pentru a preîntâmpina o revoltă, acesta hotărăște să schimbe locurile persoanelor din delegație astfel încât nici una să nu-l mai vadă pe vecinul de până acum.

Scrieți un program care determină numărul de posibilități pe care le are la dispoziție șeful delegației.

Date de intrare

Fișierul de intrare **mobuto.in** conține, pe o singură linie, un numărul natural N care reprezintă numărul persoanelor din delegație.

Date de ieșire

Fișierul de ieșire **mobuto.out** va conține, pe o singură linie, un singur număr natural, corespunzător numărului de modalități în care se poate schimba configurația șirului indian conform cerinței.

Restricție

- $1 < N < 100$.

Exemplu

mobuto.in

3

mobuto.out

3

Timp de execuție: 1 secundă/test

P010414: Spreadsheet

Un *spreadsheet* este un tablou dreptunghiular de celule după cum se poate observa în figura 3. O celulă conține o valoare sau o formulă care este o expresie.

Un *spreadsheet* "simplu" este unul în care valorile din celule sunt numere întregi și expresiile pot fi sume, diferențe, produse, cături de întregi și/sau referințe la alte celule.

Dacă o expresie referă un singur întreg, atunci ea este înlocuieetă de valoarea acesteia.

Să se realizeze un program care evaluează valorile fiecărei celule.

Date de intrare

Fișierul de intrarea **calcule.in** conține un astfel de *spreadsheet*. Un *spreadsheet* poate conține cel mult 256 linii și 256

	A	B	C	...	Z	AA	AB	...	AZ	BA	...	IV
1												
2												
3												
...												
256												

Figura 3

coloane după cum indică figura 3. Coloanele sunt etichetate cu litere mari $A, B, \dots, Z, AA, AB, \dots, IV$. Liniile sunt etichetate cu numere de la 1 la 256. O celulă este identificată prin specificarea coloanei urmată de numărul liniei, aceasta numindu-se referință.

Astfel, prima celulă din colțul stânga-sus este $A1$, iar ultima celulă din colțul dreapta-jos este $IV256$. Dacă o celulă conține o constantă, atunci aceasta se specifică prin precizarea valorii acesteia, iar dacă o celulă conține o formulă, atunci aceasta se precizează prin semnul "=" urmat de expresia formulei.

Fișierul de intrare va conține linii de forma:

- referință_celulă valoare
- sau
- referință_celulă =expresie

Celulele a căror valoare nu se specifică se consideră egale cu 0.

O formulă poate conține întregi, referințe la alte celule și operatorii $+, -, *, /$ reprezentând adunarea, diferența, înmulțirea și împărțirea, precum și paranteze $()$.

Se consideră că toate constantele și formulele sunt corecte din punct de vedere sintactic.

O expresie are următoarea sintaxă:

- expresie:= referință_celulă
- expresie:= constantă
- expresie:= (expresie)
- expresie:= expresie + expresie
- expresie:= expresie - expresie
- expresie:= expresie * expresie
- expresie:= expresie / expresie
- expresie:= -expresie
- constantă:= număr_întreg_cu_semn

Date de ieșire

Fișierul de ieșire **calcule.out** va conține pe câte o linie valorile pentru celulele date în fișierul de intrare în aceeași ordine sub următoarea formă:

referință_celulă valoare.

Dacă valoarea unei celule nu se poate evalua, în locul acesteia se va afișa ###.

Operațiile din cadrul unei expresii se evaluează în ordine de la stânga la dreapta ținându-se cont de prioritate, care este cea obișnuită.

Restricții și precizări

- constantele sunt numere întregi cuprinse între -32768 și 32767;





- nici o expresie nu poate conține mai mult de 75 de caractere;
- nici o expresie nu conține spații în interior.

Exemplu

calcule.in

```
A2 =B1
B1 3
C8 =C1+B2
C1 =C8
AA25 =-B1+(A2/2)
```

calcule.out

```
A2 3
B1 3
C8 ###
C1 ###
AA25 -2
```

Timp de execuție: 1 secundă/test

P010415: Gardian

O galerie de artă are forma unui poligon. Datorită faptului că bugetul alocat întreținerii galeriei de artă este redus, directorul acesteia dorește să angajeze un singur gardian care să asigure paza imobilului.

Supraveghetorul trebuie să patruleze într-un perimetru, astfel încât în orice moment să poată observa fiecare latură a încăperii.

În figura 4, se află un exemplu de poligon, iar zona hășurată reprezintă suprafața în care supraveghetorul poate patrula astfel încât să observe fiecare latură a încăperii.

Se cere să se realizeze un program care să determine această suprafață.

Date de intrare

Pe prima linie a fișierului de intrare **gardian.in** se află numărul N de colțuri ale încăperii.

Următoarele N linii conțin două numere x și y , separate printr-un singur spațiu, reprezentând coordonatele unui colț al încăperii.

Coordonatele unui colț k sunt date de două numere întregi x_k, y_k , separate printr-un spațiu.

Colțurile sunt date în ordine trigonometrică.

Date de ieșire

Fișierul de ieșire **gardian.out** va conține pe prima linie numărul M de colțuri ale suprafeței cerute.

Pe următoarele M linii se vor scrie coordonatele suprafeței sub forma x_k, y_k , unde x_k și y_k sunt două numere reale afișate cu două zecimale exacte și vor fi separate printr-un spațiu.

Colțurile vor fi scrise în ordinea crescătoare a coordonatei x_k , iar pentru valori egale ale acesteia, în ordinea crescătoare a coordonatei y_k .

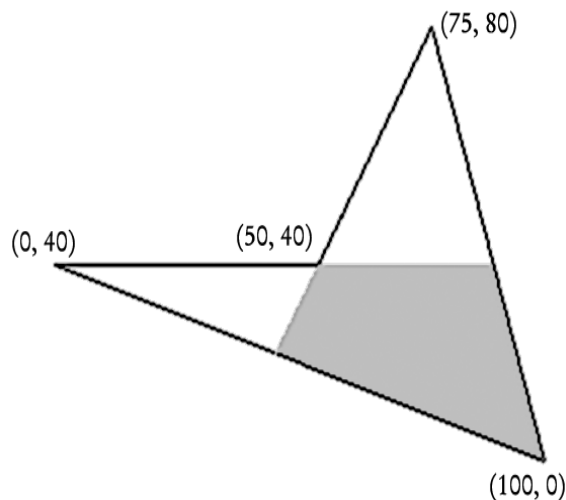


Figura 4

Dacă zona de supraveghere este vidă, atunci fișierul de ieșire va conține doar numărul 0.

Restricții

- $3 \leq N < 101$;
- $-32000 < x_k, y_k < 32000, 1 \leq k \leq N$.

Exemplu

gardian.in

```
4
100 0
75 80
50 40
0 40
```

gardian.out

```
4
40.00 24.00
50.00 40.00
87.50 40.00
```

Timp de execuție: 1 secundă/test

