

Poligon

100 puncte

Fișier sursă: poligon.pas, poligon.c sau poligon.cpp

Geo a învățat o metodă de a fixa n puncte pe un cerc de rază r , astfel încât să împartă cercul în n coarde egale ca lungime. Apoi și-a ales un număr k și a început să unească punctele succesiv, din k în k , păstrând același sens, până ce a ajuns în punctul din care a pornit. Astfel, dacă a fixat $n=10$ puncte pe cerc pe care le-a numerotat $1, 2, \dots, 10$ (vezi figura) și și-a ales $k=6$, atunci el unește punctul 1 cu 7 , apoi pe 7 cu 3 , apoi 3 cu 9 , apoi 9 cu 5 , și în sfârșit 5 cu 1 .

Apoi a colorat poligonul format în interior, pornind din centrul cercului și fără a depăși vreuna dintre liniile desenate. El se întreabă în final câte laturi are poligonul colorat și care este aria acestuia.

Cerință

Pentru n , k și r numere naturale date, se cere numărul de laturi L ale poligonului colorat și aria S a acestuia (cu 2 zecimale exacte).

Date de intrare

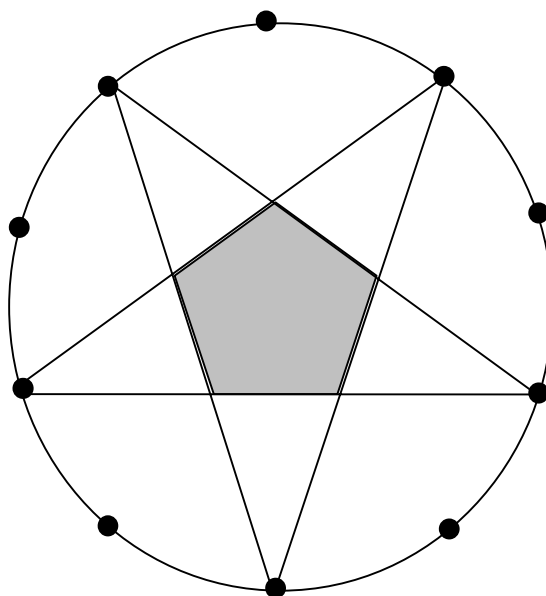
Din fișierul **poligon.in** se citesc trei numere naturale n , k și r despărțite prin câte un spațiu.

Date de ieșire

În fișierul **poligon.out** se scriu, pe linii diferite două valori: pe prima linie numărul L de laturi ale poligonului colorat, iar pe linia a doua numărul real reprezentând aria acestuia.

Restricții

- $3 < n < 10001$ număr natural
- $0 < k < n$ număr natural
- pentru n par, $2 * k \neq n$
- $10 < r < 501$
- Pentru fiecare test, dacă numărul de laturi determinat este corect, primiți 20% din punctajul maxim de pe testul respectiv. În plus, dacă și aria determinată este corectă, primiți punctajul maxim.
- Pentru 70% din testele folosite la evaluare, $n < 501$



Exemple

poligon.in	poligon.out	poligon.in	poligon.out
10 6 100	5 3468.92	30 13 200	30 5452.04

Timp maxim de execuție/test: 0.1 secunde pentru Linux și 0.1 secunde pentru Windows.

Poliție

100 puncte

Fișier sursă: `politie.pas`, `politie.c` sau `politie.cpp`

Gigel și Costel sunt doi polițiști cu experiență. Ei lucrează împreună de mulți ani și de multe ori au fost nominalizați pentru premiul „Polițiștii anului”. Anul acesta sunt hotărâți să-l câștige și pentru aceasta trebuie să încaseze cât mai mulți bani din amenzi.

În fiecare zi, Gigel și Costel pot aplica trei tipuri de amenzi pentru următoarele evenimente:

Tip	Semnificație	Suma încasată	Durata de aplicare
1	Pentru pietoni care traversează neregulamentar	S1	T1
2	Pentru șoferi de autovehicule care încalcă regulile de circulație	S2	T2
3	Pentru șoferi de mașini grele, cu transport ilegal de mărfuri	S3	T3

Amenzile de tipul 1 și 2 pot fi aplicate de un singur polițist (Gigel sau Costel). Pentru o amendă de tipul 3, Gigel și Costel trebuie să lucreze împreună (unul verifică actele de transport, iar celălalt verifică marfa).

Durata de aplicare a unei amenzi reprezintă timpul necesar polițiștilor pentru a verifica acte, a scrie proces verbal, etc. Dacă un polițist aplică o amendă la momentul x , iar durata aplicării amenzii este y , polițistul care aplică amenda va deveni disponibil abia la momentul $x+y$. Polițiștii nu fac minute suplimentare, ei sunt în activitate din minutul 1 până în minutul T exclusiv, deci trebuie să fie liberi să plece acasă în minutul T . Pentru că în noul cod rutier nu le mai este permis polițiștilor să tragă șoferii pe dreapta și să-i lase să aștepte, pentru a aplica o amendă de tipul 1 sau 2 trebuie să fie liber măcar un polițist, iar pentru a aplica o amendă de tipul 3 ambii polițiști trebuie să fie liberi la momentul în care survine evenimentul.

Cei doi polițiști stau la pândă și observă evenimentele din trafic. Dacă nu pot aplica amenzi pentru toate evenimentele care intervin în trafic, ei sunt nevoiți să le aleagă pe acelea care, în total, le aduc mai mulți bani.

Cerință

Scrieți un program care să determine suma maximă pe care o pot încasa din amenzi Gigel și Costel într-o tură.

Date de intrare

Fișierul de intrare `politie.in` conține:

- pe prima linie numărul natural T , reprezentând minutul la care cei doi polițiști sunt liberi să plece acasă;
- pe linia a doua, două numere naturale $S1$ $T1$ (suma încasată și durată aplicării unei amenzi de tipul 1);
- pe linia a treia, două numere naturale $S2$ $T2$ (suma încasată și durată aplicării unei amenzi de tipul 2);
- pe linia a patra, două numere naturale $S3$ $T3$ (suma încasată și durată aplicării unei amenzi de tipul 3);
- pe linia a cincea, un număr natural N (numărul de evenimente survenite în trafic);
- pe fiecare dintre următoarele N linii se află două numere naturale `tip timp` (`tip` poate fi 1, 2 sau 3 și reprezintă tipul amenzii ce poate fi aplicată; `timp` reprezintă timpul la care a survenit evenimentul, exprimat în număr de minute față de începutul turei).

Numerele scrise pe aceeași linie sunt separate prin câte un spațiu. Evenimentele sunt în ordine cronologică.

Date de ieșire

Fișierul de ieșire `politie.out` conține o singură linie pe care este scrisă suma maximă ce poate fi încasată.

Restricții și precizări

$0 < T, T1, T2, T3 < 201$

$0 < S1, S2, S3 < 51$

$0 < N < 501$

Evident, există evenimente care intervin simultan în trafic.

Exemple

```
politie.in
300
10 20
30 30
50 25
8
1 5
3 10
3 20
1 130
2 142
2 160
3 180
2 280
```

```
politie.out
140
```

Explicație

Aplică amândoi amenda de tipul 3 din minutul 10, apoi Costel aplică singur amenda de tipul 1 din minutul 130, apoi Gigel aplică singur amenda de tipul 2 din minutul 142, apoi amândoi aplică amenda de tipul 3 din minutul 180.

Timp maxim de execuție/test: 0.2 secunde pentru Linux și 1.2 secunde pentru Windows.

Sea

100 puncte

Fișier sursă: sea.pas, sea.c sau sea.cpp

Pe mare se află N vapoare. Malul este în mod curios perfect drept și este reprezentat prin axa Ox a sistemului de coordonate. Cele N vapoare sunt reprezentate prin perechi de coordonate (Vx_i, Vy_i) , unde Vy_i este strict pozitiv (marea este deasupra axei Ox). Pe mal se află M faruri, date prin coordonatele lor Fx_i (fiind exact la limita dintre mare și uscat, y -ul lor este întotdeauna 0). Cele M faruri sunt ciudate pentru că ele nu pot lumina decât în stânga. Astfel aria luminată de fiecare far i este delimitată de un sfert de cerc cu o rază Fr_i . Mai exact, un vapor este luminat de un anumit far dacă se află în stânga farului (are x -ul mai mic) și distanța de la far la vapor este **mai mică sau egală** cu valoarea Fr_i asociată farului respectiv.

Pentru fiecare far se mai dă și un număr natural strict pozitiv Fni . Din motive greu de înțeles, șeful portului dorește ca fiecare far i să lumineze cel puțin Fni vapoare (un vapor poate fi luminat de mai multe faruri). El dorește consum minim de energie și vrea să afle pentru fiecare far raza minimă necesară pentru a lumina numărul cerut de vapoare.

Cerință

Determinați pentru fiecare far valoarea Fr_i care reprezintă raza minimă necesară pentru ca farul să lumineze cel puțin Fni vapoare.

Date de intrare

Prima linie a fișierului `sea.in` conține două numere întregi N și M separate printr-un spațiu, reprezentând numărul de vapoare, respectiv numărul de faruri. Fiecare dintre următoarele N linii conține câte o pereche de numere reale separate printr-un spațiu Vx_i și Vy_i (coordonatele vapoarelor). Fiecare dintre următoarele M linii conține câte o pereche de numere separate printr-un spațiu, unul real Fx_i și unul întreg Fni (coordonatele orizontale și numerele asociate farurilor).

Date de ieșire

Fișierul `sea.out` va conține M linii, fiecare linie conținând un număr real, dat cu 4 zecimale: pe linia i se află raza minimă necesară pentru ca farul i să lumineze Fni vapoare.

Restricții

- $1 \leq N \leq 400$, $1 \leq M \leq 100\ 000$
- $0 < y, r < 100\ 000$, $-100\ 000 < x < 100\ 000$, $1 \leq Fni \leq N$
- În fișierul de intrare farurile sunt **sortate crescător** după coordonatele x .
- Nu vor exista două vapoare, sau un far și un vapor cu același x . În schimb pot exista două sau mai multe faruri cu același x , caz în care ele vor fi unul lângă altul în fișierul de intrare (evident din moment ce sunt sortate după x). Ordinea în care apar în fișierul de intrare farurile cu același x nu este definită. Pot exista chiar două faruri identice.
- Numerele reale din fișierul de intrare vor avea maxim 4 zecimale
- Rezultatul va fi verificat cu o precizie de 0.001 (rezultatul va fi considerat corect dacă modulul diferenței dintre rezultatul corect și cel furnizat de concurent nu depășește 0.001)
- Există întotdeauna soluție (pentru fiecare far i vor exista întotdeauna cel puțin Fni vapoare în stânga lui).

Exemplu

sea.in	sea.out
3 5	5.0990
-0.5 0.5	0.7071
-2 5	5.3852
3 4	0.7071
-1 1	4.4721
0 1	
0 2	

Olimpiada Națională de Informatică
Baraj 2



0 1	
5 1	

Timp maxim de execuție/test: 0.8 secunde pentru Linux și 1.6 secundă pentru Windows.