

PROBLEMA 1

100 puncte

BARA

Pe o bară rectilinie se află piese și găuri. Piesele se pot deplasa pe bară și elimina prin găurile existente. Bara este ca o riglă gradată în unități, iar printr-o deplasare a unei piese se înțelege mutarea cu o unitate spre stânga sau dreapta, dacă evident poziția respectivă nu este ocupată de altă piesă.

Cerință

Să se elimine piesele de pe bară efectuând un număr minim de deplasări.

Date de intrare

Fișierul **bara.in** ce conține;

- pe prima linie numerele n și m ce reprezintă numărul de piese respectiv numărul de găuri;
- pe linia a doua abscisele x_i , $1 \leq i \leq n$, ale pieselor;
- pe linia a treia abscisele y_i , $1 \leq i \leq m$, ale găurilor.

Date de ieșire

Fișierul **bara.out** ce conține pe primele n linii perechi de forma $x_i y_j$ cu semnificația că piesa din poziția x_i se elimină prin gaura din poziția y_j perechile sunt afișate în ordinea eliminării. Pe ultima linie din fișier se afișează numărul minim de deplasări.

Restricții și precizări

$$1 \leq n, m, x_i, y_i \leq 30000$$

Exemplu

bara.in	bara.out	Explicație
5 3	2 1	Piesa 2 se elimină prin gaura 1, 5 prin 4, 7 prin 6, 8 prin 6 și 10 prin 6. Numărul minim de mutări este 9.
2 5 7 8 10	5 4	
1 4 6	7 6	
	8 6	
	10 6	
	9	

Timp maxim de execuție: 0.5 secunde/test.

Memorie totală disponibilă 3 MB, din care 2 MB pentru stivă

Dimensiunea maximă a sursei: 10 KB.

PROBLEMA 2

100 puncte

CLĂTITE

Gigel adoră două lucruri: matematica și clătitele bunicii sale. Într-o zi, aceasta s-a apucat să prepare clătite. Gigel mănâncă toate clătitele începând de la a N-a clătită preparată, până la a M-a clătită preparată (inclusiv N și M). Pentru că el vrea să mănânce clătite cu diferite umpluturi și-a făcut următoarea regulă:



“Dacă numărul de ordine al clătitei este prim atunci aceasta va fi cu ciocolată.

Dacă numărul de ordine este pătrat perfect sau cub perfect aceasta va fi cu gem. Dacă suma divizorilor numărului este egală cu însuși numărul de ordine atunci aceasta va fi cu înghețată. (se iau în considerare toți divizorii în afară de numărul în sine, inclusiv 1).

În cazul în care o clătită îndeplinește simultan mai multe condiții, se respectă prioritatea sortimentelor: ciocolată > gem > înghețată.

Dacă niciuna dintre condițiile de mai sus nu este îndeplinită, pentru cele cu numărul de ordine par, clătita va fi cu zahar, iar pentru numărul de ordine impar, clătita va fi simplă.”

Cerințe

Cunoscându-se N și M, numere naturale, să se determine câte clătite a mâncat Gigel în total și numărul de clătite din fiecare tip.

Date de intrare

Fișierul **clatite.in** conține pe prima linie numerele N și M separate printr-un spațiu.

Date de ieșire

Fișierul **clatite.out** conține pe prima linie numărul total de clătite mâncate de Gigel iar pe a doua linie 5 numere naturale separate printr-un spațiu reprezentând numărul de clătite mâncate, din fiecare tip (ordinea: ciocolata, gem, înghețată, zahăr, simplă).

Restricții și precizări

- $1 \leq N \leq M \leq 300.000$

Exemplu

clatite.in	clatite.out	Explicație
3 11	9 4 3 1 1 0	Gigel a mâncat 9 clătite dintre care 4 cu ciocolată, 3 cu gem, una cu înghețată și una cu zahăr

Timp maxim de execuție: 0.2 secunde/test.

Memorie totală disponibilă 1 MB, din care 0,5 MB pentru stivă

Dimensiunea maximă a sursei: 4 KB.

PROBLEMA 3**100 puncte****INTERVALE**

Se consideră un șir de numere naturale cu elemente distincte.

Cerință:

Să se exprime mulțimea formată din aceste elemente printr-o reuniune de intervale disjuncte. Intervalele se vor lista în ordinea crescătoare a capetelor lor.

Astfel, pentru vectorul : $x=(10, 16, 3, 5, 4, 8, 11)$ se va afișa: $[3..5]+[8]+[10..11]+[16]+$.

Date de intrare:

Fișierul **intervale.in** conține;

- pe prima linie numărul N ce reprezintă numărul de elemente ale șirului;
- pe linia a doua elementele șirului separate prin spațiu

Date de ieșire:

Fișierul **intervale.out** conține lista intervalelor în formatul precizat anterior.

Restricții și precizări:

$N \leq 1000$.

Elementele șirului au valori mai mici decât 32000.

Exemple:

intervale.in	intervale.out	Explicație
8 2 10 3 11 4 8 7 6	$[2..4]+[6..8]+[10..11]+$	Se formează 3 intervale disjuncte, pentru fiecare interval precizându-se capetele acestuia
6 1 11 3 8 2 12	$[1..3]+[8]+[11..12]+$	Se formează 3 intervale disjuncte.
5 1 9 5 7 3	$[1]+[3]+[5]+[7]+[9]+$	Se formează 5 intervale disjuncte

Timp maxim de execuție: 1 secundă/test.

Memorie totală disponibilă 5 MB, din care 2 MB pentru stivă

Dimensiunea maximă a sursei: 10 KB.

PROBLEMA 4

100 puncte

HODOR

Deși pare că are o minte simplă, nefiind capabil să spună nimic altceva în afară de propriul său nume, Hodor din Game of Codes este o fire destul de analitică. În călătoria sa către Nord, Hodor a remarcat N castele. El a notat în carnetul său atât pozițiile castelurilor cât și numărul de apărători de care dispune fiecare. Pe drum realizează că singurul mod pentru a supraviețui este să meargă într-unul din castele care îi asigură protecție maximă. Și fiindcă se consideră centrul Universului și atribuie poziției sale actuale coordonatele $(0,0)$ și un sistem de axe astfel încât toate castelele să fie deasupra axei OX , după care calculează coordonatele castelurilor în acest sistem. Știm că niciun castel nu are nicio coordonată egală cu zero și că nu există două castele coliniare cu originea sistemului. El dorește să ajungă într-un castel care îi asigură o cât mai bună protecție și pentru asta el definește pericolul unui castel ca fiind modulul diferenței dintre numărul tuturor apărătorilor ce se află de o parte a dreptei ce unește punctul $(0,0)$ cu castelul și numărul tuturor apărătorilor ce se află de cealaltă parte.

Cerință

Scrieți un program care, cunoscând numărul castelurilor, locația acestora în planul xOy și numărul de apărători aflați în fiecare castel, determină care este minimul pericolului pe care Hodor îl poate obține, mergând într-un castel.

Date de intrare

Fișierul **hodor.in** va conține, pe prima linie numărul de castele n , iar pe următoarele n linii câte 3 valori ce reprezintă coordonata x , coordonata y și numărul de apărători ai fiecăruia.

Date de ieșire

Fișierul **hodor.out** va conține pe prima linie pericolul minim ce se poate obține mergând într-unul dintre castele.

Restricții

- Numărul de castele n este cel mult 100 000 și fiecare castel are cel mult 20 de apărători.
- Coordonatele sunt numere întregi, mai mici sau egale cu 1 000 000.
- Pentru 50% din teste n este cel mult 1 000, iar coordonatele sunt maxim 1 000.

Exemplu

Intrare	Ieșire	Explicație
6 -2 3 1 3 5 3 2 6 1 -5 4 2 1 1 2 4 2 1	1	Mergând în castelul cu coordonatele $(3,5)$, pe o parte Hodor are $1+2=3$ apărători (în coordonatele $(1,1)$ și $(4,2)$), iar pe cealaltă are $1+2+1=4$ apărători (în coordonatele $(-2,3)$, $(2,6)$ și $(-5,4)$).

Timp maxim de execuție: 1 secunda/test.

Memorie totală disponibilă 16 MB, din care 5 MB pentru stivă

Dimensiunea maximă a sursei: 10 KB.

Ploiești, 6 iunie 2015

PROBLEMA 5

100 puncte

CIOCOLATA

După un rezultat slăbuț la un concurs de informatică, Cristina s-a cam supărat. Dan vrea să-i ridice moralul și știe că cel mai bun mod în care poate face asta este ciocolata. Totuși, Dan nu este dispus să-i ofere Cristinei toată ciocolata pe care o are (și el a avut un rezultat slab la concurs, deci.. și el trebuie să-și ridice moralul).

Astfel, îi propune Cristinei următoarea ofertă: ”Desenează pe o hârtie un caroiaj format din N linii și M coloane pe care îl umple cu valori întregi. Cristina va primi un număr de pătrățele de ciocolată egal cu suma valorilor dintr-un dreptunghi ales de ea.”

Deoarece Cristina este prea bosumflată ca să rezolve această ”provocare” și prea obosită ca să-l convingă pe Dan să-i dea ciocolata pur și simplu, vă roagă pe voi să o ajutați. (Poate primiți și voi niște ciocolată dacă rezolvați problema. Poate...)

**Cerința**

Cunoscându-se configurația caroiajului, determinați numărul maxim de pătrățele de ciocolată pe care Cristina îl poate obține alegând un dreptunghi din matrice, precum și coordonatele celor patru colțuri ale acestuia.

Date de intrare

Fișierul **ciocolata.in** conține pe prima linie două numere naturale N și M reprezentând numărul de linii și numărul de coloane ale caroiajului. Pe următoarele N linii se găsesc câte M valori întregi reprezentând valorile din caroiaj.

Date de ieșire

Fișierul **ciocolata.out** va conține pe prima linie numărul maxim de pătrățele de ciocolată care se poate obține. Pe a doua linie se vor afla patru numere naturale reprezentând coordonatele colțurilor stânga-sus și dreapta-sus (în această ordine) a dreptunghiului ales. Pe cea de-a treia linie se vor afla tot patru numere naturale reprezentând coordonatele colțurilor stânga-jos și dreapta-jos (în această ordine) a dreptunghiului ales.

Restricții și precizări

- $1 \leq N, M \leq 500$
- Valorile din caroiaj aparțin intervalului $[-2.000.000.000 ; 2.000.000.000]$
- În cazul în care există mai multe dreptunghiuri din care se obține aceeași valoare maximă, se va alege cel cu indicele de linie al colțului stânga-sus minim. În cazul în care există mai multe dreptunghiuri cu această proprietate, se va alege cel ce are și indicele de coloană al colțului stânga-sus minim. Dacă există mai multe soluții cu proprietatea că au colțul stânga-sus cu indicii de linie și de coloană minimi, se va alege cel cu indicele de linie al colțului dreapta-jos minim. Dacă mai rămân soluții multiple, se va alege cel care are și indicele de coloană al colțului dreapta-jos minim.
- (schematic: valoare maximă \rightarrow indice linie stânga-sus minim \rightarrow indice coloană stânga-sus)

minim → indice linie dreapta-jos minim → indice coloană dreapta-jos minim)

- Se garantează că nici Cristinei și nici lui Dan nu li se va face rău de la ciocolată.

Exemplu

ciocolata.in	ciocolata.out	Explicație
5 9 3 4 -12 4 6 7 -9 5 12 0 4 5 7 9 -5 1 1 5 0 98 34 0 1 7 -7 1 1 6 7 8 -9 0 -2 3 5 22 47 62 -31 55 0 -83 23 77 -10	362 1 1 1 9 5 1 5 9	Dreptunghiul determinat de punctele (1, 1), (1, 9), (5, 1), (5, 9)

Timp maxim de execuție: 0.3 secunde/test.

Memorie totală disponibilă 5 MB, din care 1 MB pentru stivă

Dimensiunea maximă a sursei: 10 KB.