

Descrierea generală a problemelor

Nr. crt.	Denumirea problemei	Restricția referitoare la volumul utilizat de memorie	Restricția referitoare la timpul de execuție, secunde	Punctajul alocat problemei
1.	MoldSEF	≤ 32 Mb	$\leq 1,0$	100
2.	Aria	≤ 256 Mb	$\leq 1,0$	100
3.	Or	≤ 256 Mb	$\leq 1,0$	100

Notă. În caz de egalitate de punctaj, prioritate se va da concurentului care a obținut punctajul respectiv primul.

MoldSEF

Se știe că în fiecare an Agenția Națională pentru Curriculum și Evaluare organizează concursul MoldSEF. Concursul este organizat pe trei categorii tematice: Științe, Științe aplicate și Tehnologii informaționale. Spre marea surprindere a organizatorilor anul acesta s-au înscris foarte mulți participanți, astfel încât Agenția a fost nevoită să organizeze o preselecție a participanților. Pentru simplitate elevii sunt grupați în echipe de doi. Evident, că sunt elevi care vor să participe la mai multe categorii. Pentru a selecta cele mai bune echipe, Agenția a organizat preselecția în felul următor:

1. Fiecare elev susține un test care conține întrebări pe fiecare din cele 3 categorii.
2. Juriul evaluează fiecare elev și oferă puncte (numere întregi pozitive) pentru fiecare categorie. Elevul i va avea punctajele A_i, B_i, C_i
3. Se consideră că doi elevi i și j sunt compatibili și pot forma echipă pe o categorie dacă și numai dacă au coincidență la evaluare pe o singură categorie, adică în relațiile $A_i = A_j, B_i = B_j, C_i = C_j, i < j$, doar una este adevărată.
4. La concursul final sunt admiși doar echipele formate din elevi compatibili. Un elev poate fi compatibil cu mai mulți elevi.

Sarcină. Elaborați un program care din N elevi calculează numărul de potențiale echipe compatibile (perechi de elevi compatibili) în baza rezultatelor obținute la evaluare.

Date de intrare. Prima linie a intrării standard conține un număr întreg N – numărul de elevi. Fiecare dintre următoarele N linii conține trei numere întregi separate prin spațiu. Linia i conține valorile A_i, B_i, C_i - evaluarea elevului i pe cele trei categorii.

Date de ieșire. Prima și singura linie a ieșirii standard va conține numărul de potențiale echipe compatibile.

Restricții. $1 \leq N \leq 100\,000$, $1 \leq A_i, B_i, C_i \leq 100$, $1 \leq i \leq N$. Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în descrierea generală a problemelor propuse pentru rezolvare. Fișierul sursă va avea denumirea `moldsef.pas`, `moldsef.c` sau `moldsef.cpp`.

Exemplu 1.*Intrare*

```
3
1 2 3
1 4 5
1 2 4
```

Ieșire

```
2
```

Explicație: Cele 2 potențiale echipe pot fi formate din primul elev și al doilea elev (coincid evaluările pe prima categorie) sau al doilea și al treilea elev (la fel au coincidență pe evaluarea la prima categorie).

Exemplu 2.*Intrare*

```
4
100 100 100
100 100 100
100 99 99
99 99 100
```

Ieșire

```
5
```

Explicație: Cele 5 potențiale echipe pot fi formate din:

Echipa 1 – elevii 1, 3 (coincidență pe categoria 1)

Echipa 2 – elevii 1, 4 (coincidență pe categoria 3)

Echipa 3 – elevii 2, 3 (coincidență pe categoria 1)

Echipa 4 – elevii 2, 4 (coincidență pe categoria 3)

Echipa 5 – elevii 3, 4 (coincidență pe categoria 2)

Punctarea.

1. Pentru 45% din teste $1 \leq N \leq 100$, $1 \leq A_i, B_i, C_i \leq 50$
2. Restul 55% fără restricții adiționale. ($1 \leq N \leq 100\,000$, $1 \leq A_i, B_i, C_i \leq 100$)

Aria

Victor s-a angajat ca animator la o tabără de vară pentru copii. Într-o zi el are sarcina să organizeze o activitate de colorare. Pentru această activitate Victor are la dispoziție o coală mare de hârtie în pătrățele și un set mare de carioci de diferite culori. Pentru a implica toți cei N copii în activitate Victor trasează o axă de coordonate centrată pe mijlocul colii și distribuie fiecărui copil câte o cariocă și dimensiunile unui dreptunghi indicând lățimea și înălțimea acestora. Sarcina fiecărui copil este să deseneze și coloreze în culori diferite dreptunghiul de dimensiunile indicate centrat în centrul de coordonate cu lățimea paralelă cu axa x și înălțimea paralelă cu axa y .

După finalizarea activității Victor se întrebă care ar fi aria părții colorate a colii de hârtie, adică care este numărul de pătrățele unitare colorate, care aparțin cel puțin unui dreptunghi.

Sarcină. Elaborați un program care determină aria acoperită de cele N dreptunghiuri.

Date de intrare. Prima linie a intrării standard conține un număr întreg N – numărul de dreptunghiuri. Fiecare dintre următoarele N linii conține două numere întregi pare X și Y separate prin spațiu, dimensiunile (lățimea și respectiv înălțimea) fiecărui dreptunghi.

Date de ieșire. Prima și singura linie a ieșirii standard va conține aria calculată.

Restricții. $1 \leq N \leq 1\,000\,000$, $2 \leq X, Y \leq 10^7$. Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în descrierea generală a problemelor propuse pentru rezolvare. Fișierul sursă va avea denumirea `aria.pas`, `aria.c` sau `aria.cpp`.

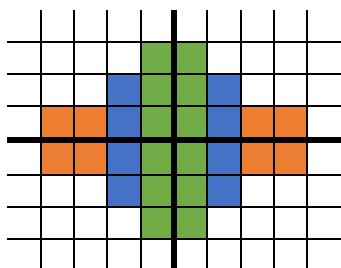
Exemplu 1.

Intrare

```
3
8 2
4 4
2 6
```

Ieșire

```
28
```



Explicație: Cele 3 dreptunghiuri colorate în culori diferite acoperă 28 de pătrățele unitare, adică aria este 28.

Exemplu 2.*Intrare*

```
5
2 10
4 4
2 2
8 8
6 6
```

Ieșire

```
68
```

Punctarea. Testele vor fi grupate în următoarele categorii, după cum urmează:

- A. 10% din punctaj: *toate dreptunghiurile sunt una în alta*
- B. 35% din punctaj: $X, Y < 3333$.
- C. 45% din punctaj: *niciun dreptunghi nu va fi amplasat strict în interiorul altui dreptunghi.*
- D. 10% din punctaj: *fără restricțiile din categoriile precedente.*

Or

Victor construiește un circuit logic simplu în atelierul său. Circuitul constă din n fire de pornire, notate cu x_1, x_2, \dots, x_n și m elemente logice *OR*, notate cu c_1, c_2, \dots, c_m . Fiecare element are exact două intrări și o ieșire. Fiecare dintre intrări este conectată fie la o fire de pornire x_j , fie la ieșirea altui element c_j . Desigur, într-un circuit logic nu există cicluri și, mai mult, se respectă faptul că intrarea lui c_j poate fi conectată la ieșirea lui c_i doar atunci când $i < j$.

Fiecare fir de pornire din circuit poate fi setat la valoarea 0 sau 1, iar valoarea ieșirii fiecărui element este operația logică *OR* a intrărilor sale - valoarea este 0 dacă valorile ambelor intrări sunt 0, în caz contrar este 1.

Victor nu știe valorile inițiale ale firelor de pornire, dar cu măsurători atente, a determinat valorile ieșirii unor elemente.

Sarcină. Elaborați un program care identifică valorile rămase ale ieșirilor care pot fi determinate fără ambiguitate pe baza măsurătorilor.

Date de intrare. Prima linie a intrării standard conține numerele întregi pozitive n și m separate prin spațiu – numărul de fire de pornire și numărul de elemente din circuit. Următoarea linie conține un șir de m caractere care descrie valoarea măsurată a ieșirii elementului c_j , sau caracterul '?' dacă Victor nu a efectuat această măsurare. Următoarele m linii conțin etichetel intrărilor. Linia j , $j \leq m$ conține etichetele celor două intrări ale circuitului c_j , fiecare etichetă este fie a firului de pornire în forma " x_i " unde se respectă $1 \leq i \leq n$, fie a elementului " c_i " unde se respectă $1 \leq i < j$. Cele două intrări ale elementului c_j pot fi la fel. Puteți presupune că valorile măsurate nu sunt contradictorii.

Date de ieșire. Prima și singura linie a ieșirii standard va conține un șir de m caractere - caracterul j din șir trebuie să corespundă valorii ieșirii c_j sau să fie '?' dacă acea valoare nu poate fi determinată fără ambiguitate.

Restricții. $2 \leq n \leq 10\,000$; $1 \leq m \leq 10\,000$. Restricțiile referitoare la timpul de execuție și volumul utilizat de memorie sunt date în descrierea generală a problemelor propuse pentru rezolvare. Fișierul sursă va avea denumirea `or.pas`, `or.c` sau `or.cpp`.

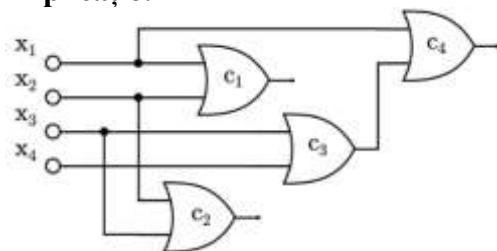
Exemplu 1.*Intrare*

```
4 4
10??
x1 x2
x2 x3
x3 x4
x1 c3
```

Ieșire

```
10?1
```

Explicație: Datele de intrare formează următorul circuit:



Observăm că $c_1 = 1$, $c_2 = 0$, iar valorile lui c_3 și c_4 nu se cunosc.

Din $c_1 = 1$ putem deduce că ori x_1 ori x_2 ori ambii au valoarea 1. Totodată din $c_2 = 0$, putem deduce că x_2 și x_3 au valoarea 0. Rezultă că x_1 are valoarea 1. Valoarea lui c_3 nu poate fi determinată deoarece valoarea lui x_3 este 0, respectiv c_3 poate fi 0 sau 1, deoarece x_4 poate fi 0 sau 1. Deoarece valoarea lui x_1 este 1, deducem că c_4 este 1. Obținem răspunsul 10?1

Exemplu 2.*Intrare*

```
4 5
11??
x1 x2
x3 x4
x1 x3
x2 x4
c3 c4
```

Ieșire

```
11??1
```

Punctarea. Testele vor fi grupate în următoarele categorii, după cum urmează:

- A. Pentru 7% din teste, $n \leq 15$, $m \leq 20$.
- B. Pentru alte 42% din teste, $n \leq 500$, $m \leq 500$
- C. Pentru restul 51% testele sunt *fără restricții adiționale*.