Concursul INTERNAȚIONAL de programare ACM 2004/2005

Vă prezentăm în continuare enunțurile celor nouă probleme propuse spre rezolvare la etapa regională sud-est europeană a celui mai important concurs de programare destinat studenților.

P040701: Perioadă

Pentru fiecare prefix al unui șir de caractere *S* format din *N* caractere, dorim să determinăm dacă prefixul este sau nu un șir periodic.

Cu alte cuvinte, pentru fiecare valoare *i* cuprinsă între 2 și *N*, dorim să determinăm cel mai mare număr întreg *K* mai mare decât 1 (dacă există un astfel de număr) astfel încât prefixul de lungime *i* al lui *S* poate fi scris sub forma A^K, unde prin A^K se înțelege șirul obținut prin concatenarea a *K* șiruri *A*. De exemplu, avem $(ab)^3$ = ababab.

În plus, pentru fiecare astfel de prefix, dorim să determinăm lungimea perioadei (*K*).

Date de intrare

Denumirea fișierului de intrare este **A.IN**.

Fișierul de intrare conține mai multe seturi de date, fiecare set pe câte două linii.

Prima linie conține lungimea *N* a șirului *S*, iar cea de-a doua linie conține șirul *S*.

Sfârșitul fișierului este marcat printr-o linie care conține doar valoarea 0.

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare set de date, se va scrie pe o linie mesajul `Test case #`, urmat de numărul de ordine al setului de date (în ordinea din fișierul de intrare; numerotarea începe de la 1).

În continuare, pentru fiecare prefix al șirului *S* care conține o perioadă de lungime mai mare decât 1, se va scrie câte o linie pe care se vor afla câte două numere întregi separate printr-un spațiu.

Primul dintre aceste numere va reprezenta lungimea prefixului, iar al doilea lungimea perioadei.

Aceste linii trebuie ordonate crescător în funcție de lungimea prefixului.

După prezentarea rezultatelor pentru fiecare set de date, se va scrie o linie vidă.

Restricții

Lungimile șirurilor sunt cuprinse între 2 și 1.000.000.

Toate caracterele care formează șirurile au coduri *ASCII* cuprinse între 97 și 126, inclusiv.

Exemplu

A.IN

```
3
aaa
12
aabaabaabaab
0
```

Ieșirea standard

```
Test case #1
2 2
3 3

Test case #2
2 2
6 2
9 3
12 4
```

P040702: Corporație

O corporație foarte mare dorește să își construiască o rețea proprie. La început, fiecare dintre cele *N* companii care formează corporația își construiește propriul centru de calcul și telecomunicații.

La scurt timp, pentru îmbunătățirea serviciilor, corporația va aranja anumite companii în mai multe grupuri, fiecare grup fiind servit de un singur centru de calcul și telecomunicații.

Modul în care sunt realizate grupurile este prezentat în cele ce urmează.

Corporația va alege unul dintre centrele existente *I* (care deservește un grup *A*) și o anumită companie *J* care face parte dintr-un grup *B* (nu este obligatoriu ca *J* să fie centrul de calcul și comunicații al grupului *B*).

Apoi, companiile *I* și *J* sunt legate printr-o linie de comunicații. Lungimea unei astfel de linii este întotdeauna dată de restul împărțirii la 1000 a modulului diferenței valorilor *I* și *J*.

După stabilirea legăturii, cele două grupuri vor fi unite în unul singur, care va fi deservit de centrul de calcul și telecomunicații al fostului grup *B*.

Din nefericire, după stabilirea fiecărei legături, suma distanțelor liniilor care leagă o companie de centrul de calcul și telecomunicații care o deservește s-ar putea modifica, iar utilizatorii ar dori să știe imediat noua lungime.

Va trebui să scrieți un program care poate răspunde în fiecare moment la întrebările utilizatorilor referitoare la distanțele dintre companii și centrele care le deservesc.

Date de intrare

Denumirea fișierului de intrare este **B.IN**.

Fișierul de intrare conține mai multe seturi de date, iar pe prima linie a acestuia este specificat numărul *T* al seturilor.

Pe prima linie corespunzătoare unui set de date se află valoarea *N* care reprezintă numărul companiilor care formează corporația.

Sfârșitul fișierului este marcat printr-o linie care conține doar valoarea 0.

Pe fiecare dintre următoarele linii se va afla câte o comandă. Primul caracter al unei comenzi poate fi E, I sau O.

În cazul în care primul caracter este E, atunci comanda va indica faptul că are loc o cerere a unui utilizator pentru distanța dintre o companie și centrul care o deservește.

Linia va mai conține un număr care va reprezenta numărul de ordine al companiei pentru care este realizată cererea. Între acest număr și caracterul E se va afla un singur spațiu.

În cazul în care primul caracter este I, atunci comanda va indica faptul că se realizează o legătură. Linia va mai conține două numere întregi; primul va reprezenta numărul de ordine al unui centru *I*, iar al doilea va reprezenta numărul de ordine al unei companii *J*.

Efectul acestei comenzi va fi realizarea unei legături între companiile *I* și *J* și pierderea statutului de centru de calcul și telecomunicații pentru compania *I*.

Cele două numere vor fi separate între ele printr-un spațiu, iar între primul număr și caracterul I se va afla tot un spațiu.

În cazul în care primul caracter este O, atunci comanda va indica încheierea descrierii setului de date respectiv și linia nu va mai conține nici un alt caracter.

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare comandă din fișierul de intrare în care se cere determinarea unei lungimi, programul va trebui să scrie în fișierul de ieșire câte o linie care va conține un singur număr reprezentând lungimea cerută de către utilizator.

Restricții

Numărul mesajelor care trebuie trimise este cuprins între 5 și 20.000.

Vor exista cel mult 200.000 de comenzi pentru fiecare set de date.

Pentru fiecare set de date, numărul comenzilor de stabilire a unei noi legături va fi mai mic decât numărul companiilor care formează corporația.

Exemplu

B.IN

```
1
4
E 3
I 3 1
E 3
I 1 2
E 3
I 2 4
E 3
O
```

Ieșirea standard

```
0
2
3
5
```

P040703: Peșteră

Cu mult timp în urmă, un om susținea că a explotrat coridoarele unei peșteri. Aceasta înseamnă că, la anumite momente de timp, el s-a aflat în fiecare dintre coridoare.

De fapt, coridoarele sunt segmente verticale sau orizontale, iar un coridor este considerat a fi vizitat în cazul în care exploratorul s-a aflat cel puțin o dată într-un punct al acestuia.

De asemenea, se știe că oricare două coridoare verticale nu au nici un punct comun și oricare două coridoare orizontale nu au nici un punct comun.

Acum dorim să aflăm dacă exploratorul a spus adevărul. Avem la dispoziție harta peșterii și știm că exploratorul a folosit următorul algoritm:

În fiecare moment, exploratorul încearcă să se îndrepte spre stânga. Dacă nu poate, atunci el încearcă să meargă înainte. Dacă nu poate merge nici înainte, atunci încearcă să se îndrepte spre dreapta. În sfârșit, dacă nu se poate îndrepta nici

spre dreapta, atunci exploratorul se întoarce.

Explorarea se încheie în momentul în care exploratorul revine la punctul de plecare.

Va trebui să determinați numărul coridoarelor care nu au fost vizitate.

Date de intrare

Denumirea fișierului de intrare este **C.IN**.

Fișierul de intrare conține mai multe hărți ale unor peșteri, iar pe prima linie se află numărul *T* al acestora. În continuare sunt descrise cele *T* hărți.

Prima linie corespunzătoare unei hărți conține numărul *N* al coridoarelor din peștera prezentată de harta respectivă.

Fiecare dintre următoarele *N* linii conține câte un caracter, urmat de câte trei numere. Cele trei numere sunt separate între ele prin câte un spațiu, iar primul număr este separat de caracterul de la începutul liniei tot printr-un spațiu.

Caracterul indică orientarea coridorului respectiv și poate avea una dintre valorile `V` (pentru coridoarele verticale) sau `H` (pentru coridoarele orizontale).

În cazul unui coridor vertical, primul dintre cele trei numere reprezintă coordonata orizontală a coridorului, iar următoarele două numere reprezintă coordonatele verticale ale extremităților coridorului.

Similar, în cazul unui coridor orizontal, primul dintre cele trei numere reprezintă coordonata verticală a coridorului, iar următoarele două numere reprezintă coordonatele orizontale ale extremităților coridorului.

Ultima linie corespunzătoare unei hărți conține două numere întregi, separate printr-un spațiu, urmate de un caracter (între al doilea număr și caracter se află tot un spațiu).

Numerele reprezintă coordonatele orizontală și verticală (în această ordine) ale poziției de start.

Caracterul indică direcția în care va începe explorarea și poate avea una dintre următoarele valori: `N` (în sus), `S` (în jos), `W` (spre stânga) sau `E` (spre dreapta).

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare hartă descrisă în fișierul de intrare va fi scrisă câte o linie care va conține un singur număr reprezentând numărul coridoarelor care nu au fost vizitate.

Restricții

Fișierul de intrare conține descrierea a cel mult 20 de hărți.

Fiecare hartă va conține cel mult 1000 de coridoare, dintre care cel mult 500 vor fi verticale și cel mult 500 vor fi orizontale.

Coordonatele tuturor coridoarelor sunt numere întregi a căror valoare absolută este cel mult egală cu 32767.

Poziția de start nu se va afla niciodată la intersecția a două coridoare, iar exploratorul se va putea deplasa întotdeauna în direcția precizată în fișierul de intrare.

Exemplu

C.IN

```
2
6
H 0 6 0
H 2 1 6
V 1 0 4
V 5 3 0
V 3 0 2
H 1 2 4
6 0 W
1
V 0 -5 5
0 0 S
```

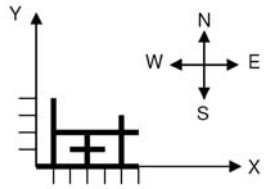
Ieșirea standard

```
1
0
```

Pentru acest exemplu, avem două hărți.

Cea de-a doua peșteră conține un singur coridor vertical cu intrarea în peșteră situată în mijlocul coridorului.

Prima peșteră este prezentată în figura următoare:



P040704: Clădire

Bob este specialist în programarea jocurilor strategice de construcție. În noul său joc avem un oraș care este format din mai multe zone în care se află străzi, copaci, fabrici și clădiri. Există și anumite porțiuni ale zonelor care nu sunt ocupate.

Partea strategică a jocului constă în a obține o chirie cât mai mare folosind spațiile libere. Pentru a câștiga bani, trebuie construite clădiri (a căror formă poate fi doar dreptunghiulară) cât mai lungi și cât mai late.

Bob încearcă să găsească o modalitate de a construi cea mai mare clădire posibilă în fiecare dintre zone.

El se confruntă cu următoarea problemă: nu pot fi distruse clădirile, copacii, străzile sau fabricile existente.

Pentru fiecare zonă se cunoaște lungimea și lățimea acesteia, iar zona este împărțită în unități de formă pătrată a căror latură este egală cu unitatea.

Chiria plătită pentru fiecare unitate pe care se construiește clădirea este de 3$.

Va trebui să scrieți un program care îl va ajuta pe *Bob* să rezolve problema.

Orașul este împărțit în *K* zone, fiecare zonă este dreptunghiulară și are o anumită lungime *M* și o anumită lățime *N*.

Unitățile ocupate sunt notate prin simbolul `R`, iar cele libere prin simbolul `F`.

Date de intrare

Denumirea fișierului de intrare este **D.IN**.

Fișierul de intrare conține mai multe descrieri ale unor zone ale orașului, iar pe prima linie se află numă-

rul *K* al acestora. În continuare sunt descrise cele *K* zone.

Prima linie corespunzătoare unei zone conține două numere întregi *M* și *N*, separate printr-un spațiu, reprezentând lungimea, respectiv lățimea zonei respective.

Fiecare dintre următoarele *M* linii va conține câte *N* simboluri, separate prin câte un spațiu. Simbolurile vor indica unitățile ocupate și unitățile libere.

După descrierea fiecărei zone urmează o linie vidă.

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare zonă descrisă în fișierul de intrare va fi scrisă câte o linie care va conține un singur număr reprezentând profitul maxim care poate fi obținut prin construirea unei clădiri în zona respectivă.

Restricție

Dimensiunile zonelor sunt numere întregi cel mult egale cu 1000.

Exemplu

D.IN

```
2
5 6
R F F F F F
F F F F F F
R R R F F F
F F F F F F
F F F F F F

5 5
R R R R R
R R R R R
R R R R R
R R R R R
R R R R R
```

Ieșirea standard

```
45
0
```

P040705: Rădăcină cubică

Sarcina voastră este de a determina rădăcina cubică a unui număr întreg pozitiv.

Nu se mai știe exact care este motivul pentru care este nevoie de un program care să determine un astfel de număr, dar cu siguranță are legătură cu o prințesă, un tânăr țăran, un sărut și o jumătate de împărăție (una uriașă, vă asigurăm).

Așadar, va trebui să scrieți un program care va rezolva această problemă crucială.

Date de intrare

Denumirea fișierului de intrare este **E.IN**.

Fișierul va conține mai multe numere întregi, fiecare pe câte o linie. Pot exista mai multe linii vide.

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare număr din fișierul de intrare va fi scrisă câte o linie care va conține două numere, separate printr-un spațiu.

Cea de-a doua valoare reprezintă rădăcina cubică a numărului, trunchiată (nu rotunjită!) la zece zecimale exacte.

Prima valoare reprezintă o sumă de control a cifrelor scrise ale rădăcinii cubice.

Suma de control este dată de restul împărțirii la 10 a numărului obținut prin însumarea cifrelor scrise ale rădăcinii cubice.

Restricții

Numerele din fișierul de intrare conțin cel mult 150 de cifre.

Lungimea nici unei linii a fișierului de intrare nu depășește 255 de caractere.

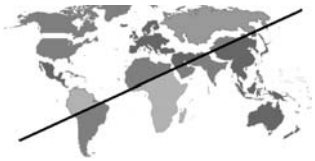
Exemplu

E.IN

```
1
                    8
1000
        2
33076161
```

Ieșirea standard

```
1 1.0000000000
2 2.0000000000
1 10.0000000000
0 1.2599210498
6 321.0000000000
```



P040706: Cablu TV

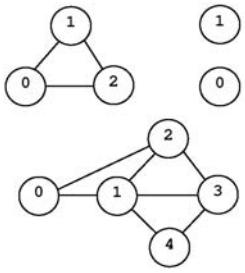
Într-o rețea de cablu TV legăturile între relee sunt bidirecționale. Rețeaua este considerat a fi *completă* dacă între oricare două relee ale rețelei există posibilitatea realizării unei legături directe sau indirecte.

O rețea vidă sau o rețea formată dintr-un singur releu este considerată a fi completă.

Factorul de siguranță al unei rețele cu *n* relee este considerat a fi *n* dacă rețeaua rămâne completă indiferent câte relee sunt eliminate.

Dacă există posibilitatea ca rețeaua să nu rămână completă, atunci factorul de siguranță este dat de numărul minim de relee care trebuie eliminate pentru ca rețeaua să nu mai fie completă.

Să considerăm rețelele din imaginea de mai jos unde cercurile reprezintă relee, iar liniile reprezintă legăturile dintre acestea.



Prima rețea rămâne completă indiferent câte relee eliminăm. Așadar,

potrivit primei reguli, factorul său de siguranță este 3.

Cea de-a doua rețea nu este completă chiar dacă nu eliminăm nici un releu. Așadar, potrivit celei de-a doua reguli, factorul său de siguranță este 0.

Pentru ca cea de-a treia rețea să nu mai fie completă putem elimina fie releele 1 și 2, fie releele 1 și 3. Așadar, factorul de siguranță al acestei rețele este 2.

Date de intrare

Denumirea fișierului de intrare este **F.IN**.

Fișierul de intrare conține mai multe seturi de date, fiecare set pe câte o linie.

Primele două numere de pe o linie reprezintă numărul *n* al releelor unei rețele și numărul total *m* al legăturilor dintre acestea.

Urmează *m* perechi de forma `(u,v)`, unde *u* și *v* sunt numere întregi cuprinse între 0 și *n* - 1 și identifică două relee între care există o legătură. Vom avea întotdeauna $u < v$, dar legăturile pot fi descrise în orice ordine.

Cu excepția descrierilor legăturilor care nu conțin spații, spațiile pot apărea oriunde în fișierul de intrare și au rol de separatori.

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare rețea descrisă în fișierul de intrare va fi scrisă câte o linie care va conține un singur număr reprezentând factorul de siguranță al rețelei respective.

Restricție

Fiecare rețea conține cel mult 50 de relee.

Exemplu

F.IN

```
1 0
3 3 (0,1) (0,2) (1,2)
2 0
5 7 (0,1) (0,2) (1,3) (1,2)
            (1,4) (2,3) (3,4)
```

Ieșirea standard

```
0
1
3
0
2
```

Fișierul de intrare conține cinci linii care descriu cinci rețele (ultima dintre ele este prezentată în exemplul anterior pe două linii din motive de spațiu).

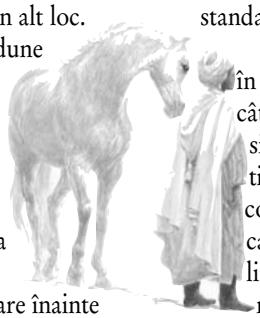
Prima rețea este vidă, cea de-a doua este formată dintr-un singur releu, iar următoarele trei sunt cele prezentate în imagine.

P040707: Ali Baba

Ali Baba, faimosul personaj al poveștilor din copilărie ar vrea să devină nemuritor pentru a putea continua să îi facă fericiți pe copii.

Pentru a-și mări renumele el trebuie să demonstreze că este în continuare capabil să facă diferite lucruri neobișnuite.

De-a lungul unei cărări se află *n* comori, fiecare într-un alt loc. *Ali Baba* trebuie să adune toate cele *n* comori și să realizeze acest lucru foarte repede.



Așadar, el trebuie să își dea seama care este ordinea în care va aduna comorile astfel încât să ajungă la fiecare înainte de termenul limită pentru fiecare dintre ele.

De asemenea, el va trebui să își aleagă cea mai favorabilă poziție de start cu putință.

Ali Baba are o listă cu locațiile tuturor comorilor, împreună cu termenele limită pentru fiecare dintre acestea.

Poziția fiecărei comori este dată de distanța față de extremitatea stângă a cărării.

Timpul necesar ridicării unei comori (după ce a ajuns la ea) este neglijabil.

Ali Baba trebuie să determine timpul minim în care poate să ia toate comorile respectând termenele limită pentru fiecare dintre acestea.

Date de intrare

Denumirea fișierului de intrare este **G.IN**.

Fișierul de intrare conține mai multe seturi de date, fiecare descriind o cărare pe care se află comori.

Prima linie corespunzătoare unui set de date conține numărul *c* al comorilor.

Fiecare dintre următoarele *c* linii conține două numere întregi. Primul dintre ele reprezintă locația unei comori, iar al doilea reprezintă termenul ei limită.

Comorile sunt descrise în ordinea distanțelor față de extremitatea din stânga a cărării.

În fișierul de intrare pot apărea oricâte spații; acestea au rolul de a separa numerele.

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare cărare descrisă în fișierul de intrare va fi scrisă câte o linie care va conține un singur număr reprezentând timpul minim necesar strângerii comorilor. Totuși, în cazul în care comorile nu pot fi strânse, linia corespunzătoare va conține numai mesajul `No solution`.

Restricție

Pe fiecare cărare se află cel mult 10000 de comori.

Exemplu

G.IN

```
15
1 3
3 1
5 8
8 19
10 15
5
1 5
2 1
3 4
4 2
5 3
```

Ieșirea standard
```
11
No solution
```

P040708: Broșuri

Bob are o sarcină dificilă; el trebuie să distribuie în diferite școli broșuri cu reclame pentru anumite activități extrașcolare.

Fiecare broșură conține un anumit număr de pagini, dar *Bob* are o listă pe care se află, pentru fiecare broșură, numărul paginilor broșurii respective. De asemenea, el cunoaște numărul școlilor care trebuie vizitate.

El trebuie să distribuie broșurile astfel încât fiecare școală să primească un număr de broșuri egal cu partea întreagă inferioară (*inf*) a numărului obținut prin împărțirea numărului total de broșuri la numărul de școli sau egal cu partea întreagă superioară (*sup*) a numărului obținut prin împărțirea numărului total de broșuri la numărul de școli.

Din nefericire, *Bob* trebuie să respecte și alte reguli. De exemplu, el trebuie să distribuie mai întâi broșuri la toate școlile care primesc *sup* broșuri și apoi la școlile care primesc *inf* broșuri.

Mai mult, orice broșură *A* distribuită la o școală *i* trebuie să aibă mai puține sau cel mult la fel de multe pagini ca o broșură *i* distribuită la o școală *j*, în cazul în care școala *i* a primit broșuri înaintea școlii *j*.

În plus, când *Bob* distribuie broșurile, el trebuie să respecte ordinea lor în lista pe care o are și, în sfârșit, el trebuie să le distribuie cât mai repede posibil.

Șeful său îl va verifica întrebându-l care a fost numărul de pagini al primei broșuri distribuite la o anumită școală. Șeful va specifica numărul de ordine al școlii (în ordinea în care *Bob* a vizitat școlile).

Prima școală vizitată de către *Bob* are numărul de ordine 0.

Date de intrare

Denumirea fișierului de intrare este **H.IN**.

Fișierul de intrare conține mai multe seturi de date, fiecare descriind un set de broșuri.

Prima linie corespunzătoare unui astfel de set conține numărul școlilor care trebuie vizitate.

Cea de-a doua linie conține numărul de ordine al școlii alese de șeful lui *Bob*.

Cea de-a treia linie conține numărul *b* al broșurilor.

În continuare în fișier se află *b* numere întregi reprezentând numărul paginilor din fiecare broșură.

Seturile de date sunt separate printr-o linie vidă.

În fișierul de intrare pot apărea oricâte spații; acestea au rolul de a separa numerele.

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare set de date din fișierul de intrare va fi scrisă câte o linie care va conține un singur număr reprezentând răspunsul la întrebarea șefului.

Restricții

Bob va trebui să distribuie cel mult 3000 de broșuri.

Numărul paginilor fiecărei broșuri este un număr întreg pentru care sunt folosiți doi octeți pentru reprezentare.

Exemplu

H.IN
```
3
2
7
3 5 9 1 11 14 2
```

Ieșirea standard
```
11
```

P040709: Cantor

Una dintre cele mai frumoase demonstrații ale matematicii moderne este cea realizată de *George Cantor*, potrivit căreia mulțimea numerelor raționale este numărabilă. Demonstrația constă în enumerarea explicită a numerelor raționale, așa cum se arată în continuare:

```
1/1 1/2 1/3 1/4 1/5 ...
2/1 2/2 2/3 2/4
3/1 3/2 3/3
4/1 4/2
5/1
```

Primul termen este `1/1`, al doilea este `1/2`, al treilea este `2/1`, al patrulea `3/1`, al cincilea `2/2` și așa mai departe.

Pentru o valoare dată *N*, va trebui să determinați cel de-al *N*-lea termen.

Date de intrare

Denumirea fișierului de intrare este **I.DAT**.

Fișierul de intrare conține mai multe numere, câte unul pe o linie.

Date de ieșire

Datele de ieșire vor fi scrise la ieșirea standard.

Pentru fiecare număr din fișierul de intrare, în fișierul de ieșire va fi scrisă o linie de forma `TERM` **X** `IS` **Y**/**Z** unde **X** este numărul din fișierul de intrare, **Y** este numărătorul celui de-al **X**-lea termen al șirului prezentat de *Cantor*, iar **Z** este numitorul celui de-al **X**-lea termen al șirului.

Restricție

În fișierul de intrare se află doar numere întregi cuprinse între 1 și 10.000.000.

Exemplu

I.DAT
```
3
14
7
```

Ieșirea standard
```
TERM 3 IS 2/1
TERM 14 IS 2/4
TERM 7 IS 1/4
```