



# BURSELE AGORA 2004/ 2005. RUNDELE #01-#12

Vă prezentăm în continuare soluțiile problemelor propuse spre rezolvare la primele 12 runde a ediției a șasea a concursului de programare organizat de revista noastră .

## P050225: Alegeri

Elfii decid întotdeauna împreună, indiferent de motivul pentru care trebuie luată o decizie.

Chiar dacă trebuie aleasă santinela care va veghea *Poiana Vieții*, întregul neam contribuie la alegere.

Din fericire pentru ei, în momentul în care trebuie făcută o alegere, este suficient ca fiecare elf să se gândească la cea mai bună opțiune.

După ce toți și-au exprimat voința în acest fel, se știe imediat cine este elful desemnat.

Practic, de fiecare dată când trebuie ales un elf, fiecare se gândește la favoritul său.

Imediat se știe dacă există sau nu o persoană care întrunește opțiunile a mai mult de jumătate dintre elfi și decizia majorității este pusă imediat în practică.

### Date de intrare

Fișierul de intrare **INPUT.TXT** conține mai multe numere de ordine care identifică elfii, câte un număr pe o linie. Fiecare număr reprezintă alegerea unui anumit elf.

### Date de ieșire

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină o singură linie pe care se va afla numărul majoritar (cel care apare în fișierul de intrare pe mai mult de jumătate dintre linii).

Dacă în fișierul de intrare nu există un număr majoritar, atunci se va scrie valoarea -1.

### Restricții și precizări

- există cel mult 10000000 de numere în fișierul de intrare;
- numerele din fișierul de intrare sunt cuprinse între 1 și 10000000.

### Exemplu

INPUT.TXT	OUTPUT.TXT
7	7
20	
15	
7	
7	
20	
15	
7	
7	
15	
15	
7	
7	

**Timp de execuție/test:** 5 secunde

## P050226: Paradă

Imediat după ce au devenit războinici, mai mulți tineri elfi au fost nevoiți să participe la o paradă. Ei se află cu fața spre instructorul lor și, la un moment dat, acesta strigă "*La stânga!*".

Datorită emoțiilor, tinerii elfi se întorc fie spre stânga, fie spre dreapta. Cei care, după întoarcere, nu văd pe nimeni sau văd spatele unui alt elf, consideră că s-au întors corect. Cei care își dau seama că au ajuns față în față cu un elf, se sperie puțin și cred că nu s-au întors corect. Imediat se întorc în direcția cealaltă.

Pentru simplitate, vom considera că toți elfii care realizează o întoarcere se întorc simultan cu 180 de grade și au nevoie de o unitate de timp pentru efectuarea întoarcerii. Evident, după întoarceri, anumiți elfi pot ajunge din nou față în față cu alți elfi. Așadar, întoarcerile pot continua.

Pentru o configurație inițială dată, se dorește determinarea numărului unităților de timp care trec până când nu se mai întoarce nici un elf (indiferent dacă pozițiile lor finale sunt corecte sau nu).

### Date de intrare

Fișierul de intrare **INPUT.TXT** conține un șir de caractere reprezentând direcțiile în care se întorc elfii în momentul în care instructorul strigă "*La stânga!*".

Pentru un elf care se întoarce spre stânga, caracterul corespunzător va fi S, iar pentru un elf care se întoarce spre dreapta, caracterul corespunzător va fi D. Aceste caractere nu sunt separate prin spații.

**Date de ieșire**

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină două linii.

Pe prima linie se va afla configurația finală obținută în momentul în care nici un elf nu se mai întoarce.

Pe cea de-a doua linie se va afla numărul de unități de timp care trec până când nu se mai întoarce nici un elf.

**Restricție**

- în șir se află cel mult 1600 de elfi.

**Exemplu**

INPUT.TXT	OUTPUT.TXT
SDSDS	SSSDD
	2

**Timp de execuție/test:** 0.5 secunde

**P050227: Aliniere**

Un grup de  $n$  războinici elfi stau aliniați în linie dreaptă în fața conducătorului lor.

Acesta nu e mulțumit de ceea ce vede; soldații nu sunt așezați în ordinea înălțimii.

Căpitanul cere câtorva războinici să iasă din rând, astfel ca cei rămași, fără a-și schimba locurile, doar apropiindu-se unul de altul (pentru a nu rămâne spații mari între ei) să formeze un șir în care fiecare războinic vede privind de-a lungul șirului, cel puțin una din extremități (stânga sau dreapta).

Un războinic vede o extremitate dacă între el și capătul respectiv nu există un alt soldat cu înălțimea mai mare sau egală cu a lui.

Așadar, trebuie determinat numărul minim de războinici care trebuie să părăsească formația, astfel încât șirul rămas să respecte condiția impusă de conducător.

**Date de intrare**

Prima linie a fișierului de intrare

**INPUT.TXT** conține numărul  $n$  al războinicilor din șir.

Fiecare dintre următoarele  $n$  linii va conține înălțimea unui războinic din șir. Aceste linii vor respecta ordinea amplasării inițiale a războinicilor.

**Date de ieșire**

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină pe prima linie numărul  $k$  al războinicilor care trebuie să părăsească șirul.

Pe fiecare dintre următoarele  $k$  linii se va afla numărul de ordine al unui războinic care trebuie să părăsească șirul.

Aceste numere vor fi scrise în ordine crescătoare.

**Restricții și precizări**

Numărul războinicilor din șir este cuprins între 2 și 1000.

Înălțimile soldaților sunt numere reale cuprinse între 1 și 10.

Dacă există mai multe soluții va putea fi aleasă oricare dintre ele.

**Exemplu**

INPUT.TXT	OUTPUT.TXT
8	4
1.86	1
1.86	3
1.30621	7
2	8
1.4	
1	
1.97	
2.2	

**Timp de execuție:** 2 secunde/test

**P050228: Culori**

Doi elfi au pus pe o masă  $n$  pătrățele și  $m$  cerceuțe. Unul a ales pătrățelele și celălalt cerceuțele și au desenat pe ele mai multe benzi colorate.

Apoi au început să se joace cu pătrățelele și cerceuțele. Au decis că un cerceuț poate fi amplasat pe un pătrățel dacă există cel puțin o culoare care apare pe ambele.

Ei doresc să formeze perechi din care fac parte un cerceuț și un pătrățel astfel încât să se obțină cât mai multe perechi.

**Date de intrare**

Fișierul de intrare **INPUT.TXT** conține pe prima linie numărul  $n$  al pătrățelelor. Pe fiecare dintre următoarele  $n$  linii sunt descrise benzile corespunzătoare unui pătrățel. Primul număr de pe o astfel de linie este nu-

mărul  $b$  al benzilor, iar următoarele  $b$  numere reprezintă codurile celor  $b$  culori.

Următoarea linie conține numărul  $m$  al cerceuțelor. Pe fiecare dintre următoarele  $m$  linii sunt descrise benzile corespunzătoare unui cerceuț. Primul număr de pe o astfel de linie este numărul  $b$  al benzilor, iar următoarele  $b$  numere reprezintă codurile celor  $b$  culori.

Numerele de pe o linie vor fi separate prin câte un spațiu. Pătrățelele și cerceuțele vor fi descrise în ordinea dată de numărul lor de ordine.

**Date de ieșire**

Prima linie a fișierului de ieșire **OUTPUT.TXT** trebuie să conțină pe prima linie numărul  $k$  al perechilor formate.

Fiecare dintre următoarele  $k$  linii va conține câte două numere, separate printr-un spațiu, reprezentând numerele de ordine ale unui pătrățel, respectiv cerc, care formează o pereche.

**Restricții și precizări**

Numărul cerceuțelor este cuprins între 1 și 100. Numărul pătrățelelor este cuprins între 1 și 100.

Pătrățelele sunt identificate prin numere cuprinse între 1 și  $n$ . Cerceuțele sunt identificate prin numere cuprinse între 1 și  $m$ .

Numărul benzilor colorate de pe cerceuțe și pătrățele este cuprins între 1 și 10.

Un pătrățel sau un cerc poate face parte din cel mult o pereche.

Dacă există mai multe soluții trebuie determinată una dintre acestea.

**Exemplu**

INPUT.TXT	OUTPUT.TXT
3	2
1 1	1 1
1 2	3 2
1 3	
4	
2 1 2	
1 3	
2 3 4	
1 4	

**Timp de execuție:** 0,5 secunde/test

**P050229: Copaci**

Într-o poiană elfii au plantat  $n$  noi copaci. Se cunosc coordonatele fiecărui copac, iar dimensiunile copacilor sunt neglijabile. Riscul ca un copac să nu se dezvolte normal este invers proporțional cu distanța față de cel mai apropiat copac.

Elfii doresc să identifice unul dintre copacii pentru care riscul de a nu se dezvolta corect este maxim.

Cu alte cuvinte, se dorește determinarea unui copac pentru care distanța față de cel mai apropiat copac este minimă.

**Date de intrare**

Fișierul de intrare **INPUT.TXT** conține pe prima linie numărul  $n$  al copacilor plantați. Fiecare dintre următoarele  $n$  linii va conține câte două numere întregi, separate printr-un spațiu, care reprezintă coordonatele unuia dintre copacii plantați.

**Date de ieșire**

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină o singură linie pe care se vor afla două numere, separate printr-un spațiu, care reprezintă coordonatele unuia dintre copacii pentru care riscul de a nu se dezvolta normal este maxim.

**Restricții și precizări**

Numărul copacilor plantați este cuprins între 2 și 5000.

Toate coordonatele copacilor sunt numere cuprinse între 0 și 1000.

Nu pot exista doi sau mai mulți copaci situați la aceleași coordonate.

Întotdeauna vor exista cel puțin doi copaci pentru care riscul este maxim; în fișierul de ieșire vor fi scrise doar coordonatele unuia dintre ei.

**Exemplu**

INPUT.TXT	OUTPUT.TXT
5	0 0
0 0	
0 2	
1 1	
2 0	
2 2	

**Timp de execuție:** 0,5 secunde/test

**P050230: Risc**

După ce au revenit de la culesul fructelor, elfii au format  $n$  grămezi de fructe, fiecare conținând câte  $m$  fructe. Fructele dintr-o grămadă sunt de același tip, dar fiecare grămadă conține fructe de tipuri diferite.

Un elf mai distrat începe să se amuze și să amestece fructele din grămezi. În speranța că jocul său nu va fi observat, el are grijă ca în fiecare dintre grămezi să rămână exact  $m$  fructe.

Totuși, un străjer a observat că el s-a jucat cu fructele și îi cere acestuia să aducă câte un fruct de fiecare tip (din fiecare grămadă).

Din nefericire pentru elf, străjerul a acoperit grămezile de fructe pentru ca elful să nu vadă ce fructe alege. Ca urmare, elful va trebui să aleagă, aproape la întâmplare, câte un fruct din fiecare grămadă.

Elful acordă fiecărei extrageri un grad de risc. Astfel, la alegerea unui fruct de tipul  $i$  din grămada care conține la început conținea fructe de tipul  $j$ , gradul de risc va fi valoarea absolută a diferenței dintre  $i$  și  $j$ .

Gradul de risc al tuturor alegerilor va fi dat de suma gradelor de risc ale alegerilor individuale.

Elful dorește să determine ce tip de fructe va trebui să aleagă din fiecare grămadă pentru ca gradul total de risc (pentru toate alegerile) să fie cât mai mic posibil.

**Date de intrare**

Prima linie a fișierului de intrare **INPUT.TXT** conține numărul  $n$  al grămezilor de fructe.

Cea de-a doua linie conține numărul  $m$  al fructelor din fiecare grămadă.

Fiecare dintre următoarele  $n$  linii va conține câte  $m$  numere, separate prin câte un spațiu. Cele  $m$  numere reprezintă tipurilor de fructe care se găsesc în fiecare grămadă, după ce elful s-a jucat cu fructele.

Pe prima linie se vor afla fructele din grămada cu numărul 1, pe a doua linie se vor afla fructele din grămada cu numărul 2 și așa mai departe până la grămada cu numărul  $n$ .

**Date de ieșire**

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină pe prima linie un număr care reprezintă cel mai mic grad total de risc posibil. Fiecare dintre următoarele  $n$  linii va conține câte un număr întreg. Acest număr va reprezenta numărul de ordine al unei grămezi din care se va alege un anumit tip de fructe. Ordinea liniilor va fi dată de numerele de ordine ale tipurilor de fructe.

**Restricții și precizări**

Numărul de grămezi este cuprins între 1 și 100.

Numărul de fructe din fiecare grămadă este cuprins între 1 și 1000.

Sortimentele și grămezile sunt identificate prin numere cuprinse între 1 și  $n$ .

În cazul în care există mai multe posibilități de alegere cu risc total minim, poate determina doar una dintre ele.

**Exemplu**

INPUT.TXT	OUTPUT.TXT
7	10
3	1
1 2 7	5
6 6 4	3
4 7 3	4
4 2 3	7
2 1 3	2
7 5 1	6
5 5 6	

**Timp de execuție:** 5 secunde/test

**P050231: Divizori**

Elfii au început din nou să se joace cu numerele. De data aceasta ei vor ca, pentru un număr natural  $n$  dat, să determine cel mai mic număr natural care are exact  $n$  divizori.

**Date de intrare**

Fișierul de intrare **INPUT.TXT** conține pe prima linie numărul  $n$  al divizorilor pe care trebuie să îi aibă numărul căutat.

**Date de ieșire**

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină o singură linie pe care



se va afla cel mai mic număr natural care are exact  $n$  divizori.

### Restricții și precizări

Numărul divizorilor pe care trebuie să îi aibă numărul căutat este cuprins între 2 și 2457.

### Exemplu

INPUT . TXT	OUTPUT . TXT
6	12

**Timp de execuție:** 0,3 secunde/test

### P050232: Spioni

Pentru a elimina amenințarea oricilor, elfii și-au dat seama că trebuie să găsească o soluție eficientă pentru a rezista.

A devenit clar că, până la urmă, dacă se continuă lupta în același mod, orcii îi vor nimici pe elfi.

Ca urmare, ei au pus la punct o rețea formată din  $n$  spioni care să desopere punctele vulnerabile ale orcilor.

Din motive de securitate, nu oricare doi spioni vor putea comunica direct între ei. Totuși, există posibilitatea ca oricare doi spioni să comunice unul cu altul, eventual cu ajutorul unor spioni intermediari.

Rețeaua de spioni este considerată a fi sigură dacă eliminarea oricărui dintre spioni nu va afecta comunicarea dintre ceilalți spioni.

În primul rând se dorește să se știe dacă rețeaua de spioni este sigură. În cazul în care rețeaua nu este sigură, trebuie identificați spionii cheie, legăturile cheie, precum și toate grupurile de agenți care formează subrețele sigure.

O subrețea sigură trebuie să fie maximală, în sensul că nu poate face parte dintr-o altă subrețea sigură mai mare.

Un spion cheie este un spion a cărui eliminare duce la imposibilitatea comunicării între cel puțin doi dintre spionii rămași.

O legătură cheie este dată de doi spioni care pot comunica în mod direct, dar întreruperea legăturii directe dintre cei doi duce la imposibilitatea comunicării între ei.

### Date de intrare

Prima linie a fișierului de intrare **INPUT . TXT** conține numărul  $n$  al spionilor.

Cea de-a doua linie a fișierului de intrare conține numărul total  $m$  al perechilor de agenți care pot comunica direct.

Fiecare dintre următoarele  $m$  linii ale fișierului conține câte două numere întregi, separate printr-un spațiu, care identifică doi agenți care pot comunica direct.

### Date de ieșire

Fișierul de ieșire **OUTPUT . TXT** trebuie să conțină pe prima linie mesajul DA în cazul în care rețeaua este sigură și mesajul NU în caz contrar.

Cea de-a doua linie trebuie să conțină numărul  $k$  al spionilor cheie. Fiecare dintre următoarele  $k$  linii va conține un număr care va reprezenta numărul de identificate ale unui spion cheie.

Următoarea linie a fișierului va conține numărul  $l$  al legăturilor cheie dintre agenți. Fiecare dintre următoarele  $l$  linii va conține câte două numere întregi, separate printr-un spațiu, care identifică doi agenți între care există o legătură cheie.

Următoarea linie trebuie să conțină numărul  $p$  al subrețelelor sigure. Fiecare dintre următoarele  $p$  linii trebuie să descrie o subrețea. Primul număr de pe o astfel de linie va conține numărul  $t$  al spionilor care fac parte din subrețeaua corespunzătoare liniei. Linia va mai conține  $t$  numere care identifică spionii care formează subrețeaua.

### Restricții și precizări

Numărul spionilor este cuprins între 3 și 100, iar cel al legăturilor directe dintre ei este cel mult egal cu 1000.

Spionii vor fi identificați prin numere cuprinse între 1 și  $n$ .

În cazul în care rețeaua de spioni este sigură, nu vom avea nici un spion cheie, nici o legătură cheie și va exista o singură subrețea sigură (formată din toți spionii).

Fiecare agent cheie va apărea o singură dată în fișierul de ieșire.

Fiecare legătură cheie va fi descrisă o singură dată. Subrețelele sigure pot fi descrise în fișierul de ieșire în orice ordine, iar numerele de ordine ale spionilor care fac parte dintr-o subrețea sigură pot fi scrise în orice ordine.

### Exemplu

INPUT . TXT	OUTPUT . TXT
10 13	NU
1 2	4
1 3	3
2 3	4
2 4	5
3 4	8
3 5	2
4 8	3 5
5 6	4 8
5 7	5
6 7	4 1 2 3 4
8 9	2 3 5
8 10	2 4 8
9 10	3 5 6 7
	3 8 9 10

**Timp de execuție:** 2 secunde/test

### P050233: Criză

Datorită faptului că atacurile oricilor continuă, elfii trebuie să ia măsuri drastice.

Ei vor să grupeze cele  $m$  cărări dintre cele  $n$  sate ale elfilor în grupuri astfel încât dacă se pierde controlul asupra uneia dintre satele care se află la extremitatea unei cărări dintr-un grup, atunci va exista în continuare posibilitatea de a călători între toate celelalte sate care sunt extremități ale cărărilor din grup.

Practic, se vor forma grupuri de sate (un sat poate face parte din mai multe grupuri), astfel încât, dacă se pierde controlul asupra unui sat dintr-un grup, va exista în continuare posibilitatea de a călători între oricare alte două sate ale grupului.

Un grup de sate nu poate fi subgrup al unui alt grup. Cu alte cuvinte, aceste grupuri trebuie să fie maximale.

### Date de intrare

Prima linie a fișierului de intrare **INPUT . TXT** conține numărul  $n$  al satelor elfilor. Cea de-a doua linie a



fișierului de intrare conține numărul total  $m$  al cărărilor dintre satele elfilor. Fiecare dintre următoarele  $m$  linii ale fișierului conține câte două numere întregi, separate printr-un spațiu, care identifică două sate care sunt legate printr-o cărare.

### Date de ieșire

Prima linie a fișierului de ieșire

**OUTPUT.TXT** trebuie să conțină numărul  $k$  al grupurilor formate.

Fiecare dintre următoarele  $k$  linii trebuie să descrie un grup. Primul număr de pe o astfel de linie va conține numărul  $t$  al satelor care fac parte din grupul corespunzător liniei. Linia va mai conține  $t$  numere care identifică satele care formează grupul.

### Restricții și precizări

Numărul satelor elfilor este cuprins între 3 și 100, iar cel al cărărilor dintre sate este cel mult egal cu 1000.

Satele vor fi identificate prin numere cuprinse între 1 și  $n$ .

Grupurile pot fi descrise în fișierul de ieșire în orice ordine, iar satele care fac parte dintr-un grup pot fi enumerate în orice ordine. Dacă există două sate care sunt unite printr-o cărare nu pot exista două grupuri diferite care să conțină ambele cele două sate.

Există cel mult o cărare între orice două sate. Pe o cărare se poate circula în ambele sensuri. Un grup va fi format din cel puțin două sate.

### Exemplu

INPUT.TXT	OUTPUT.TXT
10 13	5
1 2	4 1 2 3 4
1 3	2 3 5
2 3	2 4 8
2 4	3 5 6 7
3 4	3 8 9 10
3 5	
4 8	
5 6	
5 7	
6 7	
8 9	
8 10	
9 10	

**Timp de execuție:** 2 secunde/test

### P050234: Plantare

Elfii au stabilit dimensiunile parcelelor și cele  $k$  tipuri de copaci care se vor planta într-o anumită zonă.

După plantare au dorit să verifice respectarea dimensiunilor parcelelor prin compararea planului inițial cu modul în care a fost împărțită zona.

Au verificat dacă a fost plantat fiecare tip de copac și dacă dimensiunea fiecărei parcele existente corespunde cu dimensiunea din planul inițial.

Inițial nu s-a stabilit locația exactă a parcelelor, deci nu este importantă poziția unei parcele.

Tipurile de copaci sunt identificate prin numere cuprinse între 1 și  $k$ .

### Date de intrare

Fișierul de intrare **INPUT.TXT** conține pe prima linie dimensiunile zonei în care au fost plantați copacii. Cele două numere ( $m$  și  $n$ ) sunt separate printr-un spațiu.

Pe fiecare dintre următoarele  $m$  linii se vor afla câte  $n$  numere. Aceste numere reprezintă tipurile copacilor plantați în zona respectivă. Numerele de pe o linie sunt separate printr-un singur spațiu.

Pe următoarea linie se află un număr  $l$  care indică numărul parcelelor din planul inițial de plantare a copacilor.

Fiecare dintre următoarele  $l$  linii conține câte trei numere separate prin câte un spațiu.

Primele două numere reprezintă dimensiunile parcelei corespunzătoare, iar al treilea număr reprezintă tipul de copac care trebui plantat pe parcela respectivă.

### Date de ieșire

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină  $l$  linii.

Fiecare dintre aceste rânduri va conține un mesaj corespunzător uneia dintre parcelele din planul inițial (liniile vor respecta ordinea în care au fost descrise parcelele în fișierul de intrare).

În cazul în care în zonă există o parcelă de dimensiunea dată care

conține tipul corect, atunci mesajul va fi Corect!

Dacă nu există nici o parcelă conținând tipul corect, atunci mesajul va fi Parcela inexistentă!

Dacă există o parcelă de tipul specificat, dar dimensiunea sa nu este corectă, atunci mesajul va fi Dimensiuni eronate!

### Restricții și precizări

Dimensiunile zonei sunt cuprinse între 1 și 20.

Dacă o anumită parcelă are dimensiuni mai mari decât cele specificate în plan, ea este totuși considerată corectă deoarece există și o "sub-parcelă" de dimensiunea specificată.

### Exemplu

**INPUT.TXT**

```

5 10
1 1 1 2 2 2 2 3 3 4
1 1 1 2 2 2 2 3 3 4
5 5 5 2 2 2 2 7 7 4
5 5 5 6 6 6 6 7 7 4
5 5 5 6 6 6 6 7 7 4
8
2 3 1
3 4 2
2 2 3
5 1 4
3 4 5
2 4 6
3 2 7
4 4 8

```

### OUTPUT.TXT

```

Corect!
Corect!
Corect!
Corect!
Dimensiuni eronate!
Corect!
Corect!
Parcela inexistentă!

```

**Timp de execuție:** 2 secunde/test

### P050235: Alți copaci

Elfii doresc să planteze mai mulți copaci în *Pădurea Aurie*. Ei au decis ca noii copaci să fie plantați în linie.

Cu mult entuziasm, un tânăr elf, din păcate fără prea multă experiență, a decis să dea și el o mână de ajutor



și a marcat pozițiile în care el credea că ar fi cel mai bine să fie plantați copacii.

Din păcate el nu a luat în considerare faptul că între oricare doi copaci trebuie să existe o distanță de cel puțin  $x$  pași.

S-a dovedit că, deși din punct de vedere estetic poiana ar fi arătat minunat, nu toți copacii ar fi putut supraviețui.

Pentru ca el să nu se supere prea tare, elfii au decis să planteze cât mai mulți copaci, numai în câteva dintre punctele respective, dar între oricare doi copaci să fie o distanță de cel puțin  $x$  pași (aceasta este unitatea de măsură pentru lungime a elfilor).

### Date de intrare

Prima linie a fișierului de intrare **INPUT.TXT** conține numărul  $n$  al pozițiilor care au fost marcate.

Cea de-a doua linie conține un număr care reprezintă distanța minimă (exprimată în pași) care trebuie să se afle între doi copaci care vor fi plantați.

Fiecare dintre următoarele  $n - 1$  linii conține câte un singur număr care reprezintă distanța (exprimată în pași) de la o anumită poziție față de poziția anterioară.

### Date de ieșire

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină pe prima linie un număr  $k$  care reprezintă numărul copacilor care vor fi plantați.

Fiecare dintre următoarele  $k$  linii trebuie să conțină numărul de ordine al unei poziții în care va fi plantat un copac.

### Restricții și precizări

Numărul pozițiilor marcate este cuprins între 1 și 1000.

Distanța minimă care trebuie să existe între doi copaci este un număr cuprins între 1 și 2000.

Distanțele dintre două poziții marcate sunt numere cuprinse între 1 și 5000.

Dacă există mai multe posibilități de plantare a copacilor, se poate alege oricare dintre ele.

### Exemplu

#### INPUT.TXT

```
10
60
100
50
25
25
50
10
10
80
20
```

#### OUTPUT.TXT

```
5
1
2
5
8
9
```

**TimP de execuție:** 0,3 secunde/test

### P050236: Rețea magică

În mijlocul fiecărui sat al elfilor se află câte un arbore strămoșesc. Acești arbori pot comunica unii cu alții prin intermediul unor mesaje magice.

Arborii sunt identificați prin numere cuprinse între 1 și  $n$ , unde  $n$  este numărul total al arborilor.

Din nefericire nu este posibilă comunicarea dintre oricare doi arbori strămoșești.

Mai mult, dacă un arbore poate transmite un mesaj unui alt arbore, nu este obligatoriu ca al doilea arbore să poată transmite mesaje primului arbore.

Se dorește identificarea grupurilor de arbori strămoșești în cadrul cărora este posibilă comunicarea directă sau indirectă între oricare doi arbori.

Mai mult, numărul total al grupurilor trebuie să fie minim.

### Date de intrare

Prima linie a fișierului de intrare

**INPUT.TXT** conține numărul  $n$  al arborilor strămoșești.

Cea de-a doua linie a fișierului de intrare conține numărul total  $m$  al legăturilor directe dintre arborii strămoșești.

Fiecare dintre următoarele  $m$  linii conține câte două numere întregi, separate printr-un spațiu, care indică doi arbori strămoșești. Arborele identificat prin primul număr va putea transmite întotdeauna mesaje către arborele identificat prin al doilea număr.

### Date de ieșire

Fișierul de ieșire **OUTPUT.TXT** trebuie să conțină pe prima linie numărul  $k$  al grupurilor de arbori care sunt formate.

Fiecare dintre următoarele  $k$  linii va conține descrierea unui grup de arbori.

O astfel de linie va conține numerele de ordine ale arborilor care formează grupul corespunzător liniei. Aceste numere vor fi separate prin câte un spațiu.

### Restricții și precizări

Numărul arborilor strămoșești este cuprins între 1 și 500.

Numărul total al legăturilor directe dintre arborii strămoșești este cel mult egal cu 5000.

Există posibilitatea ca unul sau mai multe grupuri să fie formate dintr-un singur arbore.

Un arbore poate face parte dintr-un singur grup.

Numerele de identificare ale arborilor care fac parte dintr-un grup pot fi scrise în orice ordine.

Grupurile pot fi descrise în orice ordine.

### Exemplu

#### INPUT.TXT

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

#### OUTPUT.TXT

```
2
1 2 3
4
```

**TimP de execuție:** 0,5 secunde/test