



TEMPLU

Denumirea subdirectorului:

Denumirea fișierului de intrare:

Denumirea fișierului de ieșire:

TEMPLE

TEMPLE.IN






TEMPLE.OUT

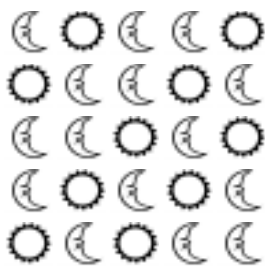
Descrierea problemei

Arheologii au descoperit ruinele unui vechi templu. Pe un perete se află o inscripție foarte interesantă, formată din simboluri ale lunii și ale soarelui. Inscripția apare sub forma unei matrice pătratice cu N linii și N coloane.

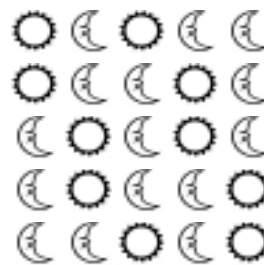
Din nefericire, trecerea timpului a dus la deteriorarea peretelui și acum numai ultima coloană mai este vizibilă. Până la urmă, arheologii au reușit să descopere o regulă interesantă chiar dacă simbolurile nu mai sunt integral vizibile.

Se consideră simbolurile lunii și ale soarelui de pe o linie care sunt dispuse într-o ordine aparent aleatoare. Totuși, în continuare se observă o regulă foarte clară. O altă linie a matricei se obține din prima, efectuând o permutare cu o poziție, adică primul element ajunge pe ultima coloană, toate celelalte sunt mutate la stânga cu o poziție. O altă linie se obține permutând din nou cu o poziție și așa mai departe, până la obținerea tuturor rândurilor.

De exemplu, în situația ipotetică în care un rând ar fi      atunci cele cinci rânduri ale inscripției sunt:



Este evident că toate aceste rânduri apar pe inscripție, dar ele nu se află în această ordine. S-a observat că există o regulă de ordonare lexicografică a acestora. Singura regulă care stă la baza acestei ordonări este aceea că soarele se află, din punct de vedere lexicografic, înaintea lunii. Așadar, fiecare rând poate fi privit ca un cuvânt și aceste cuvinte sunt ordonate folosind această regulă. În acest caz, pentru linia considerată, configurația inscripției ar fi:



Folosind regulile prezentate, va trebui să determinați prima linie a inscripției (celelalte sunt ușor de aflat dacă este cunoscută aceasta), având în vedere faptul că aveți la dispoziție doar simbolurile de pe ultima coloană.

Date de intrare

Prima linie a fișierului de intrare va conține un număr întreg N , care va indica dimensiunea inscripției. Următoarea linie va conține N numere întregi, separate prin câte un spațiu, care vor indica simbolurile de pe ultima coloană a inscripției. Pentru a simboliza luna se va folosi valoarea 1, iar pentru a simboliza soarele se va folosi valoarea 0.

Date de ieșire

Fișierul de ieșire va conține o singură linie pe care se vor afla N numere întregi, separate prin câte un spațiu, care vor reprezenta simbolurile de pe prima linie a inscripției. Se va folosi valoarea 0 pentru soare și valoarea 1 pentru lună.

Restricții

- $1 \leq N \leq 50000$;
- Datele de test sunt astfel alese încât să permită determinarea soluției.

Exemplu

TEMPLE.IN
5
1 1 1 0 0

TEMPLE.OUT
0 1 0 1 1

Timp de execuție/test: 1 secundă



SCÂNDURI

Denumirea subdirectorului:

Denumirea fișierului de intrare:

Denumirea fișierului de ieșire:

PLANKS

PLANKS.IN

PLANKS.OUT

Descrierea problemei

Unul dintre participanții la concursul *Bursele Agora* avea, la un moment dat, N scânduri de lungimi a_1, a_2, \dots, a_N . Mănat de porniri distructive, datorită faptului că nu a reușit să rezolve toate problemele de la primele două runde, el a luat un topor și a tăiat fiecare scândură în două bucăți. Astfel a obținut $2 \cdot N$ bucăți de lungimi b_1, b_2, \dots, b_{2N} .

Câteva zile mai târziu, participantului i-a părut rău că a stricat scândurile și a dorit să reconstituie scândurile inițiale. Din nefericire, bucățile s-au amestecat, deci el nu a mai știut cum să le lipească.

Sarcina voastră este de a scrie un program care determină perechile de bucăți care trebuie lipite pentru a reconstitui scândurile inițiale.

Date de intrare

Prima linie a fișierului de intrare conține numărul N care reprezintă numărul scândurilor inițiale. Pe următoarele N linii se află N numere întregi, reprezentând lungimile celor N scânduri inițiale. Pe următoarele $2 \cdot N$ linii se află $2 \cdot N$ numere întregi care reprezintă cele $2 \cdot N$ lungimi ale bucăților obținute după tăiere.

Date de ieșire

Pe prima linie a fișierului de ieșire se află numărul K al scândurilor reconstituite. Pe următoarele K linii se află câte trei numere întregi S, X și Y ($S = X + Y$) cu semnificația: o scândură de lungime S este reconstituită din două bucăți de lungime X și Y .

Numerele S reprezintă lungimi ale scândurilor inițiale. Dacă o valoare S apare la începutul unor linii din fișierul de ieșire de x ori, atunci trebuie să existe cel puțin x scânduri de lungime S . Numerele X și Y reprezintă lungimi ale bucăților obținute după tăiere.

Dacă o valoare X apare ca al doilea sau al treilea număr pe o linie a fișierului de ieșire de y ori, atunci trebuie să existe cel puțin y bucăți obținute după tăiere care au lungimea X .

Restricții și precizări

- $1 \leq N, b_i \leq 100$;
- nu există mai mult de cinci scânduri care să aibă aceeași lungime;
- nu există mai mult de cinci bucăți obținute după tăiere care să aibă aceeași lungime;
- datele de intrare sunt alese în așa fel încât să se poată reconstitui toate cele N scânduri ($K = N$);
- în cazul în care sunt reconstituite toate cele N scânduri, veți primi întregul punctaj alocat unui anumit test.
- în cazul în care nu sunt reconstituite toate cele N scânduri, dar sunt reconstituite cel puțin $\lceil 3 \cdot N / 4 \rceil$ dintre acestea, veți primi $\lceil P / 2 \rceil$ din cele P puncte alocate unui anumit test.

Exemplu

PLANKS . IN	PLANKS . OUT
6	6
10	15 10 5
15	20 10 10
20	25 10 15
25	30 15 15
30	35 20 15
35	10 5 5
5	
5	
5	
10	
10	
10	
10	
15	
15	
15	
15	
20	

Timp de execuție/test: 1 secundă



CUB

Denumirea subdirectorului:

CUBE

Denumirea fișierului de intrare:

CUBE.IN

Denumirea fișierului de ieșire:

CUBE.OUT

Descrierea problemei

Se consideră un cub de dimensiune N . Acesta poate fi împărțit în N^3 cubulețe cu latura egală cu unitatea. În fiecare dintre aceste cubulețe se află câte o literă mică a alfabetului latin.

Sarcina voastră este de a determina numărul de apariții ale unui cuvânt S de lungime K , format din litere mici distincte. Cuvântul apare în cub dacă prima sa literă apare într-un anumit cubuleț, cea de-a doua literă apare în unul dintre cele cel mult 26 de cubulețe învecinate, a treia literă apare în unul dintre cele cel mult 26 de cubulețe vecine cu cubulețul în care se află a doua literă și așa mai departe.

Pentru un cubuleț de coordonate (x, y, z) , cele 26 de cubulețe învecinate au coordonatele $(x + d_x, y + d_y, z + d_z)$, unde d_x, d_y și d_z fac parte din mulțimea $\{-1, 0, 1\}$ și $|d_x| + |d_y| + |d_z| > 0$.

Aparițiile cuvântului se pot suprapune parțial, în sensul că un anumit cubuleț poate apărea în două sau mai multe apariții ale cuvântului dat.

Date de intrare

Pe prima linie a fișierului de intrare se află cuvântul S . Pe a doua linie se află un număr întreg N care reprezintă dimensiunea cubului.

Pe următoarele N^2 linii se află câte N litere ale alfabetului latin neseperate prin spații.

Considerăm că cele N^3 cubulețe sunt identificate prin coordonatele (x, y, z) , unde x, y și z variază între 1 și N . Pe primele N linii (dintre cele N^2) se vor afla literele corespunzătoare cubulețelor pentru care $z = 1$. Pe prima dintre acestea se vor afla literele corespunzătoare cubulețelor pentru care $y = 1$. Prima dintre aceste litere va corespunde cubulețului pentru care $x = 1$, a doua va corespunde cubulețului pentru care $x = 2$ și așa mai departe. Pe a doua dintre cele N linii se vor afla literele corespunzătoare cubulețelor pentru care $y = 2$ și așa mai departe. Pe următoarele N linii se vor afla literele corespunzătoare cubulețelor pentru care $z = 2$ și așa mai departe.

Date de ieșire

Fișierul de ieșire va conține o singură linie pe care se va afla un singur număr, reprezentând numărul de apariții ale cuvântului S în cub.

Restricții

- $1 \leq N < 40$;
- $1 \leq K \leq 26$;
- numărul soluțiilor nu depășește 2000000000.

Exemplu

CUBE.IN

ab

2

aa

aa

bb

bc

CUBE.OUT

12

Explicație

Cele 12 apariții ale cuvântului sunt:

- $(1, 1, 1) \rightarrow (1, 1, 2)$;
- $(1, 1, 1) \rightarrow (1, 2, 2)$;
- $(1, 1, 1) \rightarrow (2, 1, 2)$;
- $(1, 2, 1) \rightarrow (1, 1, 2)$;
- $(1, 2, 1) \rightarrow (1, 2, 2)$;
- $(1, 2, 1) \rightarrow (2, 1, 2)$;
- $(2, 1, 1) \rightarrow (1, 1, 2)$;
- $(2, 1, 1) \rightarrow (1, 2, 2)$;
- $(2, 1, 1) \rightarrow (2, 1, 2)$;
- $(2, 2, 1) \rightarrow (1, 1, 2)$;
- $(2, 2, 1) \rightarrow (1, 2, 2)$;
- $(2, 2, 1) \rightarrow (2, 1, 2)$.

Timp de execuție/test: 3 secunde