

LUYỆN TẬP ICPC PTIT : ROUND 7

BÀI 7A - DÃY CON DÀI NHẤT

Cho dãy số gồm A và B gồm có N phần tử. Hãy tìm dãy con dài nhất của dãy A (các phần tử không nhất thiết phải liên tiếp nhau) sao cho tổng các phần tử của dãy con này lớn hơn hoặc bằng tổng các phần tử tương ứng trong dãy B.

Input

Dòng đầu tiên là số nguyên N ($2 \leq N \leq 100\,000$)

Dòng thứ hai gồm N số nguyên của dãy A[].

Dòng thứ hai gồm N số nguyên của dãy B[] ($0 \leq A[i], B[i] \leq 10^9$).

Output

In ra một số nguyên là độ dài của dãy con lớn nhất thỏa mãn yêu cầu của đề bài.

Test ví dụ:

Input:	Output
3 100 100 5 2 2 1000	2

Giải thích test:

Dãy con được lựa chọn có chỉ số lần lượt là 1, 2.

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

BÀI 7B - XẾP BÓNG

Có N quả bóng khác nhau và N chiếc hộp giống nhau (không phân biệt thứ tự các hộp). Một chiếc hộp có thể có nhiều quả bóng, nhưng cũng có thể không có quả bóng nào. Hãy đếm số cách xếp N quả bóng vào N chiếc hộp?

Ví dụ với $N = 3$ và 3 quả bóng a, b, c, ta có 5 cách xếp bóng là:

1. 3 quả cùng vào 1 hộp.
2. 3 quả mỗi quả 1 hộp.
3. Quả a vào một hộp, b và c cùng vào 1 hộp khác.
4. Quả b vào một hộp, c và a cùng vào 1 hộp khác.
5. Quả c vào một hộp, b và a cùng vào 1 hộp khác.

Input

Dòng đầu tiên là số lượng bộ test T ($T \leq 50$).

Mỗi test gồm một số nguyên N duy nhất ($1 \leq N \leq 50$).

Output

Với mỗi test, in ra đáp án tìm được trên một dòng.

Test ví dụ:

Input	Output
5	1
1	2
2	5
3	15
4	115975
10	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

BÀI 7C - KHÔNG CHIA HẾT

Cho dãy số $A[]$ có N phần tử. Hãy tìm số chỉ số i ($1 \leq i \leq N$) thỏa mãn tính chất:

Với mọi chỉ số j (j khác i) thì $A[i]$ không chia hết cho $A[j]$.

Input:

Dòng đầu tiên là số nguyên N ($1 \leq N \leq 200000$).

Dòng thứ hai gồm N số nguyên $A[i]$ ($1 \leq A[i] \leq 10^6$).

Output:

In ra số lượng chỉ số thỏa mãn tính chất của đề bài

Test ví dụ:

Input	Ouput
5 32 11 8 5 24	3
4 4 4 4 4	0
10 1 2 3 4 5 6 7 8 9 10	1

Giải thích test 1: Các chỉ số $i = 2, 3, 4$ thỏa mãn.

BÀI 7D - BẢN ĐỒ

Các nhà khảo cổ mới phát hiện ra một khu Hoàng Thành được xây dựng từ nhiều thế kỷ trước. Theo nhận định ban đầu, Hoàng Thành gồm nhiều bức tường song song hoặc vuông góc với nhau. Để tiến hành nghiên cứu, các nhà khảo cổ đã xây dựng bản đồ các bức tường của khu Hoàng Thành. Cụ thể, bản đồ được mô tả trên mặt phẳng tọa độ Đề các vuông góc Oxy trong đó các bức tường là các đoạn thẳng song song với một trong hai trục tọa độ. Theo các dữ liệu thu thập được, có N bức tường được đánh số từ 1 đến N . Bức tường thứ i là đoạn thẳng nối từ điểm có tọa độ nguyên $(x[i], y[i])$ đến điểm có tọa độ nguyên $(u[i], v[i])$, $i = 1, 2, \dots, N$. Hai đoạn thẳng mô tả hai bức tường khác nhau có không quá một điểm chung.

Để in bản đồ, các nhà khảo cổ dùng một máy vẽ chuyên dụng hoạt động theo nguyên tắc:

- Bước 1: Chọn một điểm có tọa độ nguyên, bút vẽ của máy vẽ được kích hoạt để di chuyển

và hạ đầu bút xuống điểm này;

- Bước 2: Bút vẽ di chuyển liên tục song song với một trong hai trục để vẽ các đoạn thẳng mô tả các bức tường trong bản đồ;

- Bước 3: Nếu đã vẽ xong toàn bộ các bức tường thì kết thúc công việc, trong trường hợp ngược lại, quay về Bước 1 để tiếp tục thực hiện việc vẽ.

Hoạt động của máy vẽ được tối ưu hóa sao cho không có đoạn nào được vẽ quá một lần và số lần kích hoạt bút vẽ là ít nhất.

Yêu cầu: Cho biết các đoạn thẳng mô tả các bức tường, hãy tính số lần kích hoạt bút vẽ ít nhất vẽ được bản đồ mô tả các bức tường của Hoàng Thành.

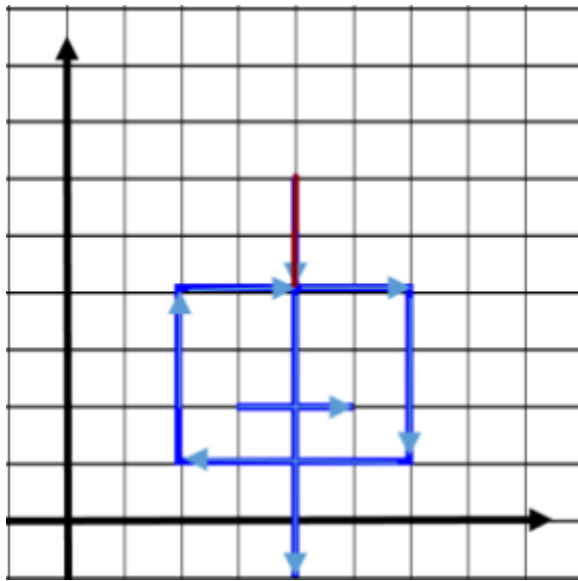
Input:

Dòng đầu là số nguyên N ($N \leq 1000$) là số lượng bức tường.

N dòng tiếp theo, mỗi dòng chứa 4 số nguyên $x[i], y[i], u[i], v[i]$ ($-1000 \leq x[i], y[i], u[i], v[i] \leq 1000$) là các tọa độ của hai đầu mút của đoạn thẳng mô tả bức tường thứ i ($i = 1, 2, \dots, N$).

Output:

In ra một số nguyên S là số lần kích hoạt bút vẽ.



Test ví dụ:

Input	Output
6	2
2 1 2 4	
2 4 6 4	
6 4 6 1	
6 1 2 1	
4 -1 4 6	
3 2 5 2	
10	4
0 -1 0 4	
0 -1 4 -1	
4 4 0 4	
4 4 4 -1	
-2 -1 0 -1	
1 0 1 2	
3 0 6 0	
-1 0 3 0	
-3 1 -3 4	
1 2 1 5	

BÀI 7E - PHẦN TỬ CHUNG

Cho K dãy số nguyên, các số trong dãy thuộc $[-10^9 \dots 10^9]$. Hãy viết chương trình tìm số xuất hiện trong cả K dãy.

Nếu không có số nào xuất hiện trong cả K dãy thì in ra “NO”, còn nếu có nhiều số cùng xuất hiện trong K dãy thì ghi số nhỏ nhất tìm được.

Input:

Dòng đầu tiên là số nguyên K ($1 \leq K \leq 100000$).

Dòng thứ hai gồm K số nguyên mô tả độ dài của từng dãy số. Tổng độ dài của tất cả dãy số không vượt quá 500000.

K dòng tiếp theo, mỗi dòng mô tả một dãy số.

Output:

Ghi ra phần tử chung nhỏ nhất tìm được, in ra “NO” trong trường hợp ngược lại.

Test ví dụ:

Input	Ouput
2 3 4 1 2 3 4 3 2 -1	2
3 5 6 7 2 1 3 4 3 4 5 -1 0 0 3 11 4 7 8 9 0 3	3

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

BÀI 7F - TỔNG CHẴN LẺ

Cho dãy số $A[]$ có N phần tử là nguyên dương không quá 100, N không quá 50.

Hãy đếm số cách chọn ra các phần tử trong dãy để được giá trị tổng là số chẵn (hoặc lẻ).

Input

Dòng đầu có số N và giá trị 0 nếu muốn tìm tổng chẵn, giá trị 1 nếu muốn tìm tổng lẻ.

Dòng tiếp theo ghi N số của dãy $A[]$

Output

Ghi ra số cách tìm được, nếu giá trị quá lớn thì hãy chia dư cho $10^9 + 7$

Ví dụ

Input	Output
2 0	2
1 3	

Giải thích ví dụ: có hai cách chọn:

- *Cách 1: không lấy phần tử nào*
- *Cách 2: lấy cả 2 phần tử*

BÀI 7G - DI CHUYỂN THEO HƯỚNG GIÓ

Bạn là thuyền trưởng của một con tàu, hiện tại bạn đang ở vị trí $[x1, y1]$.

Công việc của bạn là tìm cách đưa con tàu đến vị trí $[x2, y2]$.

Trên biển gió có thể thổi theo 4 chiều: trên, dưới, trái, phải - tương ứng với U, D, L và R.

Giả sử con tàu đứng yên:

- Nếu gió thổi theo hướng U, sau 1 giờ nó sẽ di chuyển từ tọa độ $[x, y]$ đến $[x, y + 1]$.
- Nếu gió thổi theo hướng D, sau 1 giờ nó sẽ di chuyển từ tọa độ $[x, y]$ đến $[x, y - 1]$.
- Nếu gió thổi theo hướng L, sau 1 giờ nó sẽ di chuyển từ tọa độ $[x, y]$ đến $[x - 1, y]$.
- Nếu gió thổi theo hướng R, sau 1 giờ nó sẽ di chuyển từ tọa độ $[x, y]$ đến $[x + 1, y]$.

Con tàu có tốc độ di chuyển 1 (đơn vị độ dài / giờ) và chỉ được phép đi song song với 2 trục tọa độ.

Giả sử con tàu di chuyển theo hướng L và gió thổi theo hướng U, nó sẽ di chuyển từ vị trí $[x, y]$ tới $[x - 1, y + 1]$, còn nếu hướng di chuyển con tàu và hướng gió đều là U, sau 1 tiếng nó sẽ di chuyển từ $[x, y]$ tới $[x, y + 2]$.

Cho $[x1, y1]$, $[x2, y2]$ và xâu ký tự S cho biết hướng gió TẠI THỜI ĐIỂM MỖI GIỜ, hãy tìm thời gian nhỏ nhất để con tàu tới được đích, biết hướng gió biến đổi tuần hoàn.

Input

- Dòng đầu tiên ghi 2 số $x1, y1$ ($0 \leq x1, y1 \leq 10^9$).
- Dòng thứ hai ghi 2 số $x2, y2$ ($0 \leq x2, y2 \leq 10^9$).
- Dòng thứ ba ghi số thứ tự của xâu S.
- Dòng thứ tư ghi xâu S - hướng gió tại thời điểm mỗi giờ (bắt đầu từ thời điểm 0).

Output

Một số nguyên duy nhất ghi thời gian nhỏ nhất để tới đích

Nếu không thể tới đích, in ra -1.

Ví dụ:

Input	Output
0 0 4 6 3 UUU	5

BÀI 7H - SẮP XẾP DÃY CON

Cho dãy số a_1, a_2, \dots, a_n và dãy số b_1, b_2, \dots, b_n .

Trong mỗi thao tác, bạn có thể thực hiện sắp xếp một dãy con $a[l \dots r]$ của dãy a theo thứ tự không giảm.

Ví dụ, nếu dãy a là $a[4, 2, 2, 1, 3, 1]$ và bạn chọn dãy con $a[2 \dots 5]$ thì dãy a sẽ biến đổi thành $[4, 1, 2, 2, 3, 1]$.

Hãy xác định xem bạn có thể tạo được dãy b bằng cách áp dụng (không giới hạn số lượng) các thao tác trên với dãy a hay không?

Input

Dòng đầu ghi số $t (1 \leq t \leq 3 \cdot 10^5)$ - là số bộ dữ liệu.

Dòng đầu của mỗi bộ dữ liệu ghi số $n (1 \leq n \leq 3 \cdot 10^5)$.

Dòng tiếp theo ghi n số nguyên dương $a_1, a_2, \dots, a_n (1 \leq a_i \leq n)$

Dòng tiếp theo ghi n số nguyên dương $b_1, b_2, \dots, b_n (1 \leq b_i \leq n)$.

Dữ liệu vào đảm bảo $\sum n \leq 3 \cdot 10^5$ với tất cả các bộ dữ liệu.

Output

Với mỗi bộ dữ liệu, ghi ra YES hoặc NO theo kết quả kiểm tra.

Ví dụ

Input	Output
4	YES
7	YES
1 7 1 4 4 5 6	NO
1 1 4 4 5 7 6	NO

5	
1 1 3 3 5	
1 1 3 3 5	
2	
1 1	
1 2	
3	
1 2 3	
3 2 1	

Giải thích test ví dụ đầu tiên:

Ta có thể sắp xếp a_1, a_2, \dots, a_5 sau đó a trở thành $[1, 1, 4, 4, 7, 5, 6]$; sau đó tiếp tục sắp xếp
dãy con $a_5 \dots a_6$

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb

BÀI 71 - DIỆN TÍCH TOÀN PHẦN

Trên mỗi ô của vùng diện tích có kích thước $M \times N$, các khối vuông đơn vị (kích thước $1 \times 1 \times 1$) được xếp chồng lên nhau để tạo thành một khối lớn có chiều cao bằng $H[i, j]$. Các khối lớn khi được đặt cạnh nhau sẽ che phủ một phần mặt bên của chúng.

Nhiệm vụ của bạn là hãy tính diện tích toàn phần của khối thể tích thu được, bao gồm cả phần bề mặt trên, mặt đáy và bốn mặt xung quanh.

Input:

- Dòng đầu tiên là số lượng bộ test T ($T \leq 10$).
- Mỗi test bắt đầu bởi hai số nguyên M và N . ($0 \leq M, N \leq 1000$).
- M dòng tiếp theo, mỗi dòng gồm N số nguyên mô tả chiều cao của ô (i, j) . Chiều cao của mỗi ô không vượt quá 1000.

Output:

- Với mỗi test, in ra diện tích toàn phần của khối thể tích được tạo thành.

Ví dụ:

Input	Output
5	
1 1	
1	
1 2	
1 2	6
1 1	14
2	10
3 3	30
1 1 1	38
1 1 1	
1 1 1	
3 3	
1 1 1	
1 2 0	
1 0 2	

BÀI 7J - TRUY VẤN TRÊN CÂY

Cho cây có N node với gốc tại đỉnh 1. Ban đầu, mỗi node được gán sẵn một giá trị.

Bạn được phép thực hiện 2 thao tác sau:

- add u v t ($-10000 \leq t \leq 10000$)

Với mỗi node nằm trên đường đi từ u đến v, tăng giá trị của các node đó lên t

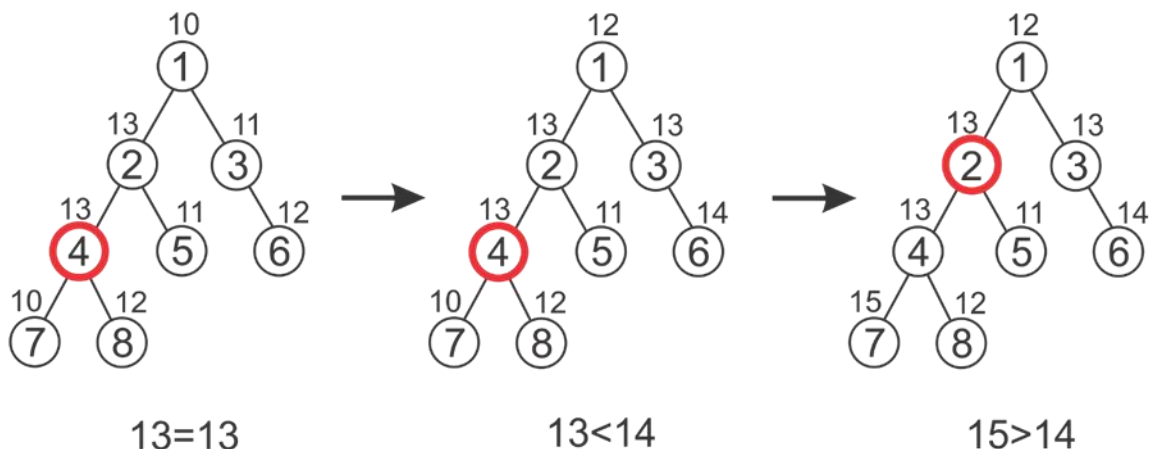
- com u ($u \neq 1$)

Xét 2 tập hợp: cây con của node u (phần A) và phần còn lại của cây (phần B).

Nếu phần tử lớn nhất của phần A nhỏ hơn phần tử lớn nhất của phần B thì in “<”;

ngược lại nếu bằng nhau thì in “=”

nếu không thì in “>”.



Input:

Dòng đầu tiên ghi số N ($2 \leq N \leq 10^5$) là kích thước của cây.

Dòng thứ 2 ghi N số nguyên x_i ($-10000 \leq x_i \leq 10000$) - giá trị ban đầu của mỗi node.

N-1 dòng tiếp theo ghi 2 số u và v ($1 \leq u, v \leq N$), biểu diễn cạnh từ u đến v.

Dòng tiếp theo ghi số M ($1 \leq M \leq 10^5$), là số truy vấn.

M dòng tiếp theo, mỗi dòng chứa một thao tác như mô tả trên.

Output:

Với mỗi thao tác loại 2, viết ra kết quả trên một dòng.

Ví dụ:

Input:	Output:
8	=
10 13 11 13 11 12 10 12	<
1 2	>
1 3	
2 4	
2 5	
3 6	
4 7	
4 8	
5	
com 4	
add 1 6 2	
com 4	
add 7 7 5	
com 2	

Giới hạn thời gian: 2s

Giới hạn bộ nhớ: 524288 Kb