

Đề Tin_11
(Đề thi gồm 4 trang)

KỶ THI CHỌN HỌC SINH GIỎI
LẦN THỨ XVII NĂM 2026
ĐỀ THI MÔN TIN HỌC LỚP 11
Thời gian làm bài 180 phút
(Không kể thời gian giao đề)

STT	Bài	Tên tệp chương trình	Dữ liệu	Kết quả	Điểm
1	Bít tết	beefs.*	Bàn phím	Màn hình	6
2	Ếch du hành	frog.*	Bàn phím	Màn hình	7
3	Gốc khác biệt	diffroot.*	Bàn phím	Màn hình	7

* là kí hiệu phần mở rộng của chương trình viết bằng ngôn ngữ lập trình C++ (cpp) hoặc Python (py)

Bài 1. Bít tết

Đầu bếp tại nhà hàng nổi tiếng nhất Ba Lan đang chuẩn bị các miếng bít tết tomahawk cho một nhóm các lập trình viên trẻ tuổi, vui vẻ (và rất đói)!

Miếng bít tết có thể được biểu diễn dưới dạng một ma trận $n \times n$. Ban đầu, nhiệt độ của toàn bộ miếng bít tết là 0 độ. Trong quá trình nấu, nhiệt độ ở các phần khác nhau của miếng bít tết (tức là các ô khác nhau trong ma trận) sẽ tăng lên.

Để chuẩn bị một miếng bít tết tomahawk hoàn hảo, đầu bếp nhắc miếng bít tết lên và đặt nó trở lại vì nướng q lần. Mỗi lần, đầu bếp có thể đặt miếng bít tết lên một trong ba mặt: trái, phải, hoặc dưới. Sau đó, anh ấy sẽ để miếng bít tết nấu trên mặt đó trong x giây.

Khi miếng bít tết được nướng ở **mặt dưới**, nó có thể được nướng trong tối đa n giây. Sau đó, nhiệt độ của tất cả các ô ở hàng thứ n trong ma trận sẽ tăng thêm x , nhiệt độ của các ô ở hàng thứ $n - 1$ sẽ tăng thêm $x - 1$, nhiệt độ của các ô ở hàng thứ $n - 2$ sẽ tăng thêm $x - 2$, ..., nhiệt độ của các ô ở hàng số $n - x + 1$ sẽ tăng thêm 1.

Khi miếng bít tết tomahawk được nướng ở **mặt phải**, nó có thể được nướng trong tối đa $\lfloor \frac{n+1}{2} \rfloor$ giây. Sau đó, nhiệt độ của tất cả các ô ở cột thứ n trong ma trận sẽ tăng thêm x , nhiệt độ của các ô ở cột thứ $n - 1$ sẽ tăng thêm $x - 1$, nhiệt độ của các ô ở cột thứ $n - 2$ sẽ tăng thêm $x - 2$, ..., nhiệt độ của các ô ở cột số $n - x + 1$ sẽ tăng thêm 1.

Tương tự, khi miếng bít tết tomahawk được nướng ở **mặt trái**, nó có thể được nướng trong tối đa $\lfloor \frac{n+1}{2} \rfloor$ giây. Sau đó, nhiệt độ của tất cả các ô ở cột đầu tiên trong ma trận sẽ tăng thêm x , nhiệt độ của các ô ở cột thứ hai sẽ tăng thêm $x - 1$, nhiệt độ của các ô ở cột thứ ba sẽ tăng thêm $x - 2$, ..., nhiệt độ của các ô ở cột số x sẽ tăng thêm 1.

Chúng ta biết rằng độ ngon của miếng tomahawk liên quan chặt chẽ đến sự tương phản về nhiệt độ, đó là lý do tại sao đầu bếp muốn tìm ra sự chênh lệch giữa phần lạnh nhất và phần nóng nhất của miếng bít tết!

Dữ liệu vào

Trong dòng đầu tiên, có hai số tự nhiên n và q ($1 \leq n \leq 10^5, 1 \leq q \leq 10^5$), kích thước của miếng bít tết và số lần đầu bếp nhắc miếng bít tết lên.

Trong mỗi dòng trong q dòng tiếp theo, có một chữ cái đơn s và một số tự nhiên x . Chữ cái s cho biết mặt của miếng bít tết tomahawk: "L" – trái, "R" – phải, hoặc "D" – dưới. Nếu $s = "D"$, thì $1 \leq x \leq n$. Ngược lại, $1 \leq x \leq \lfloor \frac{n+1}{2} \rfloor$.

Dữ liệu ra

Trong dòng đầu tiên và duy nhất, in ra một số duy nhất – sự chênh lệch về độ giữa ô lạnh nhất và ô nóng nhất trên miếng bít tết.

Ví dụ

Dữ liệu vào	Dữ liệu ra	Giải thích
4 2 L 2 R 1	2	Giải thích test thứ nhất: $\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 1 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 2 & 1 & 0 & 0 \\ 2 & 1 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 2 & 1 & 0 & 1 \\ 2 & 1 & 0 & 1 \\ 2 & 1 & 0 & 1 \\ 2 & 1 & 0 & 1 \end{bmatrix}$
3 3 R 2 D 3 R 2	6	Giải thích test thứ hai: $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & 1 & 2 \\ 0 & 1 & 2 \\ 0 & 1 & 2 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \\ 3 & 4 & 5 \end{bmatrix} \rightarrow \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \\ 3 & 5 & 7 \end{bmatrix}$

Giới hạn:

Subtask 1: 10% số điểm của bài có $n, q \leq 20$

Subtask 2: 30% số điểm của bài có $n, q \leq 1000$

Subtask 3: 40% số điểm của bài có $n \leq 1000$

Subtask 4: 20% số điểm của bài không có ràng buộc gì thêm

Bài 2. Éch du hành

Chú ếch Gorf đang du hành qua vương quốc Đầm Lầy. Thật không may, sau một cú nhảy tồi, chú đã rơi xuống một cái giếng sâu n mét. Bây giờ Gorf đang ở đáy giếng và có một chặng đường dài để leo lên.

Bề mặt tường giếng có chất lượng khác nhau: có chỗ trơn trượt, nhưng có chỗ lại có gờ đá thuận tiện. Nói cách khác, nếu Gorf đang ở độ sâu x mét so với mặt đất, thì trong một cú nhảy, chú có thể nhảy lên một khoảng cách nguyên bất kỳ từ 0 đến a_x mét (lưu ý rằng Gorf không thể nhảy xuống, chỉ có thể nhảy lên).

Thật không may, Gorf phải nghỉ ngơi sau mỗi cú nhảy (kể cả cú nhảy 0 mét). Và sau khi nhảy lên vị trí x mét dưới mặt đất, chú sẽ bị trượt xuống đúng b_x mét trong khi nghỉ ngơi.

Hãy tính số lần nhảy tối thiểu Gorf cần để chạm tới mặt đất.

Dữ liệu vào:

Dòng đầu tiên chứa một số nguyên duy nhất n ($1 \leq n \leq 300\,000$) — độ sâu của giếng.

Dòng thứ hai chứa n số nguyên $a_1, a_2, \dots, a_n$ ($0 \leq a_i \leq i$), trong đó a_i là chiều cao tối đa Gorf có thể nhảy từ độ sâu i mét dưới mặt đất.

Dòng thứ ba chứa n số nguyên $b_1, b_2, \dots, b_n$ ($0 \leq b_i \leq n - i$), trong đó b_i là khoảng cách Gorf sẽ bị trượt xuống nếu chú nghỉ ngơi ở độ sâu i mét dưới mặt đất.

Dữ liệu ra:

Nếu Gorf không thể chạm tới mặt đất, in ra -1 . Ngược lại, hãy in ra số nguyên duy nhất k — số lần nhảy tối thiểu có thể.

Ví dụ

Dữ liệu vào	Dữ liệu ra	Giải thích
3 0 2 2 1 1 0	2	Gorf đang ở đáy giếng và nhảy lên độ cao 1 mét dưới mặt đất. Sau đó, chú bị trượt xuống 1 mét và dừng lại ở độ cao 2 mét dưới mặt đất. Bây giờ, từ đây, chú có thể chạm tới mặt đất trong một cú nhảy.
2 1 1 1 0	-1	Gorf có thể nhảy lên độ cao 1 mét dưới mặt đất, nhưng sẽ bị trượt trở lại đáy giếng. Đó là lý do chú không thể chạm tới mặt đất.
10 0 1 2 3 5 5 6 7 8 5 9 8 7 1 5 4 3 2 0 0	3	Gorf có thể chạm tới mặt đất chỉ từ độ cao 5 mét dưới mặt đất. Và Gorf có thể đạt được độ cao này bằng một chuỗi các cú nhảy $10 \Rightarrow 9 \rightarrow 9 \Rightarrow 4 \rightarrow 5$ trong đó $\Rightarrow$ là cú nhảy và $\rightarrow$ là trượt trong khi nghỉ.

Giới hạn:

Subtask 1: 50% số điểm của bài có $n \leq 1000$;

Subtask 2: 20% số điểm của bài có $b_i = 0 \forall i = 1, 2, \dots, n$;

Subtask 3: 30% số điểm của bài không có ràng buộc gì thêm.

Bài 3. Gốc khác biệt

Cho một cây với n đỉnh. Mỗi đỉnh i có một giá trị a_i đi kèm với nó.

Giả sử ta chọn một đỉnh v làm gốc của cây. Đỉnh v được gọi là một **gốc khác biệt** nếu điều sau đây được thỏa mãn: trong tất cả các đường đi bắt đầu tại v và kết thúc tại một nút khác bất kỳ, tất cả các giá trị gặp trên đường đi đó đều phân biệt. Hai đường đi khác nhau có thể có các giá trị chung nhưng trên một đường đi đơn lẻ thì tất cả các giá trị phải phân biệt.

Hãy tìm số lượng **gốc khác biệt** trong cây.

Dữ liệu vào:

Dòng đầu tiên của đầu vào chứa một số nguyên duy nhất n ($1 \leq n \leq 2 \cdot 10^5$) — số lượng đỉnh trong cây.

Dòng tiếp theo chứa n số nguyên cách nhau bởi dấu cách $a_1, a_2, \dots, a_n$ ($1 \leq a_i \leq 10^9$).

$n - 1$ dòng tiếp theo, mỗi dòng chứa hai số nguyên cách nhau bởi dấu cách u và v ($1 \leq u, v \leq n$), biểu thị một cạnh nối từ u đến v .

Đảm bảo rằng các cạnh tạo thành một cây.

Dữ liệu ra:

In ra một số nguyên duy nhất — số lượng **gốc khác biệt** trong cây.

Ví dụ

Dữ liệu vào	Dữ liệu ra	Giải thích
5 2 5 1 1 4 1 2 1 3 2 4 2 5	3	1, 2 và 5 là các gốc khác biệt
5 2 1 1 1 4 1 2 1 3 2 4 2 5	0	

Giới hạn:

Subtask 1: 30% số điểm của bài có $n \leq 2000$;

Subtask 2: 20% số điểm của bài có cây là một đường thẳng;

Subtask 3: 30% số điểm của bài có $a_i \leq 2$;

Subtask 4: 20% số điểm của bài không có ràng buộc gì thêm.

-----Hết-----