

STATUE

Với phương hướng phát triển thành phố Free Contest trở thành địa điểm du lịch ăn khách, nhiều công trình kiến trúc mới đã được xây dựng trong thời gian gần đây. Công trình nổi tiếng nhất trong số đó là những bức tượng được đặt tại trung tâm thành phố.

Các bức tượng được xếp ngay ngắn thành một bảng chữ nhật gồm 2 hàng và n cột, mỗi ô gồm đúng một bức tượng (các cột được đánh số từ 0 đến $n - 1$). Với cột thứ i , bức tượng ở hàng trên có chiều cao là a_i , và bức tượng ở hàng dưới có chiều cao là b_i . Người dân thành phố cho rằng, tổng chênh lệch chiều cao giữa các bức tượng ở hàng trên với từng bức tượng ở hàng dưới càng thấp thì công trình càng có tính thẩm mỹ cao. Nói cách khác, độ kém thẩm mỹ có thể được định nghĩa bởi giá trị S như sau:

$$S = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} |a_i - b_j|$$

Để tăng tính thẩm mỹ cho công trình, Ủy ban Nhân dân (UBND) thành phố Free Contest dự định sẽ đổi chỗ một số bức tượng với nhau. Nhưng do kinh phí có hạn, UBND chỉ có thể tiến hành đổi chỗ hai bức tượng nằm cùng cột với nhau. Nói cách khác, với mỗi cột i từ 0 đến $n - 1$, UBND có thể quyết định đổi chỗ hai bức tượng a_i và b_i cho nhau, hoặc không làm gì cả.

Hãy tìm một kế hoạch đổi chỗ các bức tượng sao cho độ kém thẩm mỹ S của công trình là nhỏ nhất có thể. Nếu có nhiều phương án dẫn đến giá trị S nhỏ nhất, bạn có thể đưa ra bất kỳ phương án nào.

Chi tiết cài đặt

Bạn cần cài đặt hàm sau:

```
int[] swap_statues(int[] a, int[] b)
```

- a, b : hai dãy số có độ dài n . Với $0 \leq i < n$, a_i và b_i lần lượt là chiều cao của bức tượng nằm ở hàng trên và hàng dưới cột i .
- Hàm cần trả về một dãy số độ dài n (gọi dãy này là t) biểu diễn một kế hoạch đổi chỗ các bức tượng. Với $0 \leq i < n$, t_i cần có giá trị 1 nếu cần đổi chỗ hai bức tượng trong cột i , ngược lại t_i cần có giá trị 0.
- Hàm này được gọi duy nhất một lần.

Ràng buộc

- $1 \leq n \leq 10^5$
- $1 \leq a_i, b_i \leq 10^8$ (với mọi $0 \leq i < n$)

Ví dụ

IOI Practice Contest 2022

Ví dụ 1

Xét lời gọi hàm sau:

```
swap_states([1, 3], [4, 5])
```

Ban đầu, $a = [1, 3]$ và $b = [4, 5]$. Độ kém thắm mĩ là:

$$S = |1 - 4| + |1 - 5| + |3 - 4| + |3 - 5| = 3 + 4 + 1 + 2 = 10$$

Một kế hoạch tối ưu là đổi chỗ hai bức tượng ở cột 0. Sau khi đổi chỗ, $a = [4, 3]$ và $b = [1, 5]$. Độ kém thắm mĩ trở thành:

$$S = |4 - 1| + |4 - 5| + |3 - 1| + |3 - 5| = 3 + 1 + 2 + 2 = 8$$

Đây cũng là độ kém thắm mĩ nhỏ nhất có thể đối với ví dụ này. Do đó, hàm `swap_states` có thể trả về giá trị $[1, 0]$.

Một kế hoạch hợp lệ khác mà hàm có thể trả về là $[0, 1]$.

Ví dụ 2

Xét lời gọi hàm sau:

```
swap_states([7, 9, 5], [8, 2, 4])
```

Một kế hoạch tối ưu là đổi chỗ hai bức tượng ở cột 0 và hai bức tượng ở cột 1. Sau khi đổi chỗ, $a = [8, 2, 5]$ và $b = [7, 9, 4]$, và độ kém thắm mĩ sẽ trở thành 27. Do đó, hàm `swap_states` có thể trả về giá trị $[1, 1, 0]$.

Một kế hoạch hợp lệ khác mà hàm có thể trả về là $[0, 0, 1]$.

Ví dụ 3

Xét lời gọi hàm sau:

```
swap_states([1, 2, 3, 4], [1, 2, 3, 4])
```

Trong trường hợp này, với từng cột thì hai bức tượng đều có độ cao bằng nhau, do đó độ kém thắm mĩ sẽ không đổi dù ta đổi chỗ tượng theo bất kì cách nào. Do đó, hàm `swap_states` có thể trả về bất kì dãy t nào hợp lệ (ví dụ như $[1, 1, 0, 1]$).

Subtasks

- (10 điểm) $n \leq 16$
 - (20 điểm) $a_{i-1} \leq a_i, b_{i-1} \geq b_i$ với mọi $1 \leq i < n; a_{n-1} \leq b_{n-1}$
 - (70 điểm) Không có ràng buộc gì thêm.
-

Trình chấm mẫu

Trình chấm mẫu đọc dữ liệu vào theo định dạng sau:

- dòng 1: n
- dòng 2: $a_0 a_1 \dots a_{n-1}$
- dòng 3: $b_0 b_1 \dots b_{n-1}$

Kết quả đầu ra của trình chấm mẫu theo định dạng sau:

- dòng 1: $t_0 t_1 \dots t_{n-1}$
-