

TỔNG QUAN VỀ BÀI THI

Tên bài	Tệp chương trình	Tệp dữ liệu vào	Tệp dữ liệu ra	Điểm
XORGAND	XORGAND.*	XORGAND.INP	XORGAND.OUT	6
SẮP XẾP	SAPXEP	SAPXEP.INP	SAPXEP.OUT	7
TÊN TRỘM	TENTROM	TENTROM.INP	TENTROM.OUT	7

Phần mở rộng của tệp chương trình được đặt tùy theo ngôn ngữ lập trình được sử dụng.

BÀI 1 (6 điểm): XORGAND

Một số M được gọi là số huyền bí cơ số X nếu $(M \wedge X) > (M \& X)$. Trong đó $\wedge$ là phép XOR bit, $\&$ là phép AND bit trong C++.

Bạn được cho 1 mảng N số nguyên và Q truy vấn. Mỗi truy vấn được cho bởi bộ 3 số L, R và X . Yêu cầu tìm trong đoạn từ L đến R xem có bao nhiêu số huyền bí cơ số X .

Dữ liệu vào:

Dòng đầu chứa số nguyên N .

Dòng tiếp theo chứa N số nguyên là các phần tử của mảng.

Dòng thứ ba chứa số nguyên Q là số truy vấn.

Q dòng cuối cùng, mỗi dòng chứa 3 số nguyên L, R và X biểu thị truy vấn.

Dữ liệu ra:

Q dòng là kết quả cho từng truy vấn.

Ví dụ:

XORGAND.INP	XORGAND.OUT
5	3
1 2 3 4 5	2
2	
1 5 4	
2 5 2	

Ghi chú:

Subtask 1: 50% số test có N và $Q \leq 2000$.

Subtask 2: 50% số test còn lại có N và $Q \leq 200000$.

BÀI 2 (7 điểm). SẮP XẾP

Nhận thấy quân địch đang trong tình trạng suy kiệt cả về số lượng lẫn chất lượng, Darth Vader đã ra lệnh cho đô đốc Sakerm lập tức chuẩn bị quân đội để tổng tấn công.

Hiện tại, đô đốc Sakerm đang gấp rút chuẩn bị quân đội. Quân đội của ông có N người lính, mỗi tên lính trong hàng ngũ đều có một chỉ số riêng biệt. Dãy chỉ số của N người lính là một hoán vị của dãy số nguyên từ 1 đến N . Để đoàn quân tinh nhuệ của ông được tối ưu sức mạnh, Sakerm phải chỉnh đốn lại hàng ngũ sao cho chỉ số của các người lính trong hàng tăng dần.

Trong quá khứ, ông thường dùng các thuật toán **Bubble Sort** để chỉnh đốn hàng ngũ. Tuy nhiên do đang gấp rút và thuật toán kia lại tốn quá nhiều thời gian nên buộc Sakerm phải tìm cách sắp xếp khác. May mắn cho ông, tất cả các lính tinh nhuệ của ông đều đã được rèn luyện bài bản, chỉ cần gọi tên thì ngay lập tức 1 chuỗi hành động sau sẽ được người lính vừa được gọi tên ấy thực hiện:

- Nếu người lính bên phải có chỉ số nhỏ hơn người lính được gọi thì 2 người lính này sẽ đổi chỗ cho nhau và quá trình này sẽ lặp đi lặp lại đến khi không còn người lính bên phải có chỉ số nhỏ hơn.
- Ngược lại, nếu người lính bên trái có chỉ số lớn hơn người lính được gọi thì 2 người lính này sẽ đổi chỗ cho nhau và quá trình này sẽ lặp đi lặp lại đến khi không còn người lính bên trái có chỉ số lớn hơn.

Mỗi người lính khi được gọi tên sẽ thực hiện chuỗi hành động trên chỉ tốn 1 giây.

Sakerm nhận thấy rằng không nhất thiết phải gọi tên toàn bộ N người lính mà chỉ cần gọi 1 số người trong đó thì toàn bộ hàng ngũ sẽ được sắp xếp. Sakerm không giỏi tính toán nên ông rất muốn biết xem số lượng tên lính tối thiểu ông cần gọi tên là bao nhiêu. Ngoài ra, có thể có nhiều tập thỏa mãn có cùng số lượng phần tử. Là 1 người ham học hỏi, Sakerm tò mò muốn biết tập bé thứ K trong thứ tự từ điển là tập nào. Bạn hãy giúp Sakerm sắp xếp quân ngũ nhằm chuẩn bị tổng phản công nhé. (Dữ liệu đảm bảo luôn có nhiều hơn K tập thỏa mãn).

Input:

- Dòng đầu chứa 2 số nguyên $N \leq 10^5$ và $K \leq 10^{18}$
- Dòng thứ hai chứa N số nguyên là hoán vị của dãy số từ 1 đến N .

Output:

- Dòng đầu chứa số nguyên x là số lượng người lính cần gọi tên ít nhất.
- x dòng sau, mỗi dòng chứa 1 số nguyên thuộc tập bé thứ K , các số sắp xếp theo thứ tự từ điển cần tìm (liệt kê theo thứ tự tăng dần).

Ví dụ:

SAPXEP.INP	SAPXEP.OUT
5 1	3
3 5 1 4 2	1
	2
	4

Subtask

- 20% số test có $N \leq 6$ và $K = 1$.
- 30% số test tiếp theo có $N \leq 10^5$ và $K = 1$
- 50% số test còn lại không có điều kiện gì thêm

Giải thích:

- Sau khi gọi tên người lính có chỉ số 1 (tức là ở vị trí 3 trong dãy ban đầu), dãy trở thành **1 3 5 4 2**.
- Sau khi gọi tên người lính có chỉ số 2, dãy trở thành **1 2 3 5 4**.
- Sau khi gọi tên người lính có chỉ số 4, dãy trở thành **1 2 3 4 5**.

Như vậy, sau 3 lần gọi tên, quân đoàn đã được sắp xếp, và không có cách nào có số lần gọi bé hơn cũng như thứ tự từ điển bé hơn **1 2 4**.

Bài 3 (7 điểm). Tên trộm

Giới hạn thời gian: 2 giây, Bộ nhớ 512Mb

Tên trộm OBI thâm nhập vào trung tâm điều khiển của thị trấn A. Trung tâm điều khiển này có dạng 1 đồ thị N đỉnh vô hướng liên thông không có chu trình. Mỗi đỉnh tương ứng là 1 phòng và các cạnh là đường hầm nối 2 phòng đó. Những phòng chỉ có duy nhất 1 đường hầm nối với nó sẽ có 1 lối ra (vào) – được coi là nút lá trên đồ thị.

Biết có tên trộm đột nhập, giám đốc trung tâm điều khiển điều động các lính canh tại các lối thoát hiểm nhằm truy bắt tên trộm. Tên trộm cố gắng di chuyển ra một nút lá nào đó để thoát thân; các lính canh cố gắng di chuyển để chặn được tên trộm.

Bạn hãy tính giúp ông giám đốc xem với mỗi phòng từ i đến N , nếu tên trộm dịch chuyển ra phòng i thì cần tối thiểu bao nhiêu tên lính để chặn đứng tên trộm. Giả sử tên trộm và các lính canh đều di chuyển tối ưu: tên trộm di chuyển tối ưu để cực tiểu hóa số tên lính mình phải đối đầu. Còn lính canh luôn di chuyển về phía tên trộm. Mỗi 1 bước di chuyển sẽ đi qua được 1 cạnh. Các lính canh sẽ được đi vào qua các nút lá.

INPUT:

- Dòng đầu chứa số nguyên N ($N \leq 7 \cdot 10^4$)
- $N-1$ dòng sau, mỗi dòng chứa 2 số nguyên u, v thể hiện có đường hầm nối giữa 2 phòng u và v ($u, v \leq n$).

OUTPUT:

Ghi ra n dòng, dòng thứ i là số tên lính tối thiểu cần để chặn đứng tên trộm nếu hãn dịch chuyển ra phòng thứ i .

Ví dụ:

Tentrom.inp	Tentrom.out
6	2
1 2	2
1 3	2
2 4	1
3 5	2
5 6	1

Ghi chú:

Subtask 1: 20% số test có $N \leq 6$

Subtask 2: 20% số test có $N \leq 2000$

Subtask 3: 60% số test có $N \leq 7 \cdot 10^4$

----- Hết -----

Ghi chú: Cán bộ coi thi không giải thích gì thêm

Họ và tên thí sinh: Số báo danh:

Họ và tên giám thị số 1: Chữ ký:

Họ và tên giám thị số 2: Chữ ký: