

TRÁO BÀI (SHUFFLE.*)

Cho bộ bài gồm n lá bài được xếp thành dãy thứ tự từ 1 tới n , đầu tiên người ta ghi vào mỗi lá bài một số nguyên là số thứ tự ban đầu của lá bài đó. Xét phép tráo $S(i, m, j)$: Lấy ra khỏi bộ bài m lá bài liên tiếp bắt đầu từ lá bài thứ i , sau đó chèn m lá bài này vào trước lá bài thứ j trong số $n - m$ lá bài còn lại $1 \leq i, j \leq n - m + 1$. Quy ước rằng nếu $j = n - m + 1$ thì m lá bài lấy ra sẽ được đưa vào cuối dãy.

Ví dụ với $n = 9$:

Bộ bài ban đầu: (1,2,3,4,5,6,7,8,9)

Thực hiện $S(1,5,2)$: (1,2,3,4,5, 6,7,8,9) \rightarrow (6, 1,2,3,4,5, 7,8,9)

Thực hiện tiếp $S(5,4,6)$: (6,1,2,3, 4,5,7,8, 9) \rightarrow (6,1,2,3,9, 4,5,7,8)

Thực hiện tiếp $S(8,2,1)$: (6,1,2,3,9,4,5, 7,8) \rightarrow (7,8, 6,1,2,3,9,4,5)

Yêu cầu: Hãy cho biết số ghi trên k lá bài đầu tiên của bộ bài ($k \leq n$) sau khi thực hiện x phép tráo bài cho trước.

Dữ liệu:

- ✿ Dòng 1: Chứa ba số nguyên dương n, k, x ($n \leq 10^5, k \leq 60, x \leq 10^5$)
- ✿ x dòng tiếp theo, mỗi dòng ghi ba số nguyên i, m, j tương ứng với một phép tráo $S(i, m, j)$

Kết quả: Ghi ra một dòng chứa k số nguyên, số thứ i là số ghi trên lá bài thứ i sau khi thực hiện x phép tráo đã cho.

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách.

Ví dụ:

Sample Input	Sample Output
9 2 3	7 8
1 5 2	
5 4 6	
8 2 1	

JOSEPHUS (JOSEPHUS.*)

Tương truyền rằng Josephus và 40 chiến sĩ bị người La Mã bao vây trong một hang động. Họ quyết định tự vẫn chứ không chịu bị bắt. 41 chiến sĩ đứng thành vòng tròn và bắt đầu đếm theo một chiều vòng tròn, cứ người nào đếm đến 3 thì phải tự vẫn và người kế tiếp bắt đầu đếm lại từ 1. Josephus không muốn chết và đã chọn được một vị trí mà ông ta cũng với một người nữa là hai người sống sót cuối cùng theo luật này. Hai người sống sót sau đó đã đầu hàng và gia nhập quân La Mã (Josephus sau đó chỉ nói rằng đó là sự may mắn, hay “bàn tay của Chúa” mới giúp ông và người kia sống sót)...

Có rất nhiều truyền thuyết và tên gọi khác nhau về bài toán Josephus, trong toán học người ta phát biểu bài toán dưới dạng một trò chơi: Cho n người đứng quanh vòng tròn theo chiều kim đồng hồ đánh số từ 0 tới $n - 1$. Từ một người xác định trước, họ bắt đầu đếm từ 1, người nào đếm đến m thì bị loại khỏi vòng và người kế tiếp bắt đầu đếm lại từ 1. Trò chơi tiếp diễn cho tới khi vòng tròn chỉ còn lại 1 người.

Yêu cầu:

- ✿ Cho p là số hiệu người đếm đầu tiên, tìm q là số hiệu người cuối cùng còn lại trên vòng tròn
- ✿ Cho y là số hiệu người cuối cùng còn lại trên vòng tròn, tìm x là số hiệu người đếm đầu tiên theo luật chơi

Dữ liệu:

- ✿ Dòng 1 chứa hai số nguyên dương $n, m \leq 10^7$.
- ✿ Dòng 2 chứa hai số nguyên dương p, y ($0 \leq p, y < n$)

Kết quả: Ghi ra một dòng hai số q, x tìm được

Các số trên một dòng của Input/Output files được/phải ghi cách nhau ít nhất một dấu cách

Ví dụ

Sample Input	Sample Output
7 3	3 6
0 2	

TRÒ CHƠI Ô SỐ (SWAPGAME.*)

Trò chơi ô số là một trò chơi trí tuệ được giáo sư X phát minh và phổ biến trong trường học. Trò chơi diễn ra trên một bảng chữ nhật 2 hàng, n cột được chia làm $2n$ ô vuông. Người ta điền các số nguyên từ 1 tới n vào các ô vuông, mỗi số điền 2 lần theo một trật tự ngẫu nhiên.

Người chơi được phép thực hiện các phép ĐẢO: hoán đổi giá trị hai số ở cùng cột với mục đích làm cho hai hàng của bảng trở thành hai hoán vị của dãy số $(1, 2, \dots, n)$ (cấu hình hoàn hảo).

3	2	1	2	3
4	5	5	1	4

 →

3	5	1	2	4
4	2	5	1	3

Một người chuyên nghiệp trong trò chơi này (Cell Swapping Professional - CSP) có thể trả lời rất nhanh hai câu hỏi sau đối với một cấu hình ban đầu của trò chơi:

- ✿ Có bao nhiêu cấu hình hoàn hảo khác nhau có thể tạo thành từ cấu hình ban đầu (hai cấu hình hoàn hảo gọi là khác nhau nếu nó có một vị trí ô mang giá trị khác nhau trên hai cấu hình)
- ✿ Số lần thực hiện phép ĐẢO ít nhất là bao nhiêu để thu được một cấu hình hoàn hảo

Bạn có thể không phải người chơi chuyên nghiệp nhưng hoàn toàn có thể giúp máy tính của bạn trở thành máy chơi chuyên nghiệp, hãy thực hiện điều đó

Dữ liệu:

- ✿ Dòng 1 chứa số nguyên dương $n \leq 10^5$
- ✿ Dòng 2 chứa n số nguyên dương ghi trên hàng 1 của bảng ban đầu
- ✿ Dòng 3 chứa n số nguyên dương ghi trên hàng 2 của bảng ban đầu

Dữ liệu vào được cho đúng đắn, tức là bảng chứa đầy đủ các số từ 1 tới n , mỗi số xuất hiện 2 lần.

Kết quả:

- ✿ Dòng 1 ghi số cấu hình hoàn hảo có thể tạo thành
- ✿ Trong trường hợp dòng 1 chứa số khác 0, dòng 2 ghi số phép biến đổi ít nhất để đưa bảng về cấu hình hoàn hảo

Ví dụ

Sample Input	Sample Output
5 3 2 1 2 3 4 5 5 1 4	4 2

CHUYỂN NƯỚC (WATERMOV.*)

Các bé học sinh trường mầm non SuperKids tỏ ra say mê với các trò chơi đòi hỏi tư duy thuật toán chuyên nghiệp. Nhân dịp đến thăm trường, giáo sư X bày ra một trò chơi cho các bạn nhỏ tại đây.

Ban đầu, người chơi được cho n thùng nước đánh số từ 1 tới n . Thùng thứ i có a_i lít nước. Người chơi được quyền mức một lượng nước bất kỳ từ một thùng chuyển sang thùng liền sau (chuyển từ thùng i sang thùng $i + 1$ với i tùy chọn thỏa mãn $1 \leq i < n$). Năng lượng tiêu tốn cho thao tác này đúng bằng lượng nước được chuyển (có thể không phải là số nguyên)

Nhiệm vụ của người chơi là phải làm cho lượng nước trong các thùng sắp xếp thứ tự không giảm, tức là:

$$a_1 \leq a_2 \leq \dots \leq a_n$$

Yêu cầu: Hãy tìm phương án chơi sao cho tổng năng lượng tiêu tốn là ít nhất

Dữ liệu:

- ✿ Dòng 1 chứa số nguyên dương $n \leq 10^6$
- ✿ Dòng 2 chứa n số nguyên không âm a_1, a_2, \dots, a_n ($\forall i: a_i \leq 10^6$) cách nhau bởi dấu cách

Kết quả: Ghi ra một số thực duy nhất với 1 chữ số sau dấu chấm thập phân là tổng năng lượng tiêu tốn nếu các bé chơi theo phương án của bạn

Ví dụ

Sample Input	Sample Output
6 1 3 0 0 3 0	4.5

Giải thích:

Ta sẽ chuyển nước để được lượng nước trong các thùng là 1.0,1.0,1.0,1.0,1.5,1.5

Chuyển 2 lít từ thùng 2 sang thùng 3

Chuyển 1 lít từ thùng 3 sang thùng 4

Chuyển 1.5 lít từ thùng 5 sang thùng 6