

MÁY BÁN HÀNG TỰ ĐỘNG

Có một chiếc máy bán hàng tự động không được trang bị cơ chế trả lại tiền thừa, nó bắt buộc đã mua món hàng nào phải trả đúng giá bán, không thừa không thiếu.

Trong ví của giáo sư X có n tờ tiền mệnh giá hoàn toàn phân biệt. Ông muốn nhẩm tính với những tờ tiền này thì không thể mua món hàng từ máy với giá thấp nhất là bao nhiêu nếu máy có bán món hàng với giá đó.

Dữ liệu: Vào từ file văn bản VENDING.INP

- ✿ Dòng 1 chứa số nguyên n ($0 \leq n \leq 1000$)
- ✿ Dòng 2 chứa n số nguyên dương theo thứ tự tăng dần là mệnh giá các tờ tiền trong ví giáo sư X (mệnh giá mỗi tờ tiền không vượt quá 10^9)

Kết quả: Ghi ra file văn bản VENDING.OUT một số nguyên duy nhất là giá tiền thấp nhất mà giáo sư X không thể trả cho máy để mua hàng

Ví dụ:

VENDING.INP	VENDING.OUT
5 1 2 3 8 9	7
8 1 2 4 8 16 32 64 128	256

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: Vào từ file văn bản JOSEPHUS.INP

- ✿ 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 file văn bản JOSEPHUS.OUT một dòng ghi 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

JOSEPHUS.INP	JOSEPHUS.OUT
7 3	3 6
0 2	

LỊCH SỬA CHỮA Ô TÔ

Một cơ sở sửa chữa ô tô có nhận n chiếc xe để sửa, các xe được đánh số từ 0 tới $n - 1$. Do các nhân viên làm việc quá lười nhác nên đã đến hạn trả cho khách hàng mà vẫn chưa tiến hành sửa được chiếc xe nào. Theo hợp đồng đã ký kết từ trước, nếu bàn giao xe thứ i quá hạn ngày nào thì sẽ phải trả thêm một khoản tiền phạt là a_i .

Ông chủ cơ sở sửa chữa quyết định sa thải toàn bộ công nhân và thuê nhân công mới. Với lực lượng mới này, ông ta dự định rằng để sửa chiếc xe thứ i sẽ cần b_i ngày. Vấn đề đặt ra đối với ông là phải lập lịch sửa tuần tự các chiếc xe sao cho tổng số tiền bị phạt là ít nhất.

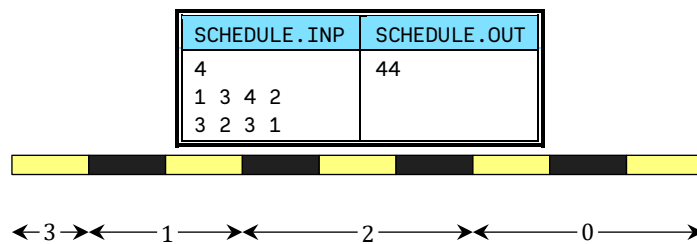
Yêu cầu: Hãy lập lịch sửa xe giúp cho ông chủ cơ sở sửa chữa ô tô sao cho tổng số tiền bị phạt là ít nhất.

Dữ liệu: Vào từ file văn bản SCHEDULE.INP

- Dòng 1: Chứa số nguyên dương $n \leq 10^5$
- Dòng 2: Chứa n số nguyên dương a_0, a_1, \dots, a_{n-1} , $\forall i: a_i \leq 1000$
- Dòng 3: Chứa n số nguyên dương b_0, b_1, \dots, b_{n-1} , $\forall i: b_i \leq 1000$

Kết quả: Ghi ra file văn bản SCHEDULE.OUT một số nguyên duy nhất là số tiền phạt tối thiểu theo phương án tìm được.

Ví dụ



Phương án tối ưu là sửa các xe theo thứ tự 3, 1, 2, 0. Tiền phạt:

Xe 3: Muộn 1 (ngày) $\times 2 = 2$

Xe 1: Muộn 3 (ngày) $\times 3 = 9$

Xe 2: Muộn 6 (ngày) $\times 4 = 24$

Xe 0: Muộn 9 (ngày) $\times 1 = 9$

Tổng cộng = 44

Nếu sửa theo thứ tự 0, 1, 2, 3 thì:

Xe 0: Muộn 3 (ngày) $\times 1 = 3$

Xe 1: Muộn 5 (ngày) $\times 3 = 15$

Xe 2: Muộn 8 (ngày) $\times 4 = 32$

Xe 3: Muộn 9 (ngày) $\times 2 = 18$

Tổng cộng = 68

CHỤP ẢNH

Lễ khai mạc thể vận hội năm 2112 có n vận động viên đánh số từ 1 tới n đứng xếp hàng ngang để chụp ảnh, ban tổ chức đã sắp xếp họ theo một thứ tự mà họ cho là đẹp nhất gọi là **thứ tự chuẩn**.

Tuy nhiên khi người thợ chụp ảnh quay lại để bấm máy, một số vận động viên đã tự ý rời hàng để bắt tay khán giả (những vận động viên khác giữ nguyên vị trí). Trọng tài cảnh cáo những vận động viên tự ý rời hàng và yêu cầu quay lại hàng ngũ, tuy nhiên những vận động viên vừa bị cảnh cáo khi quay lại hàng lại có thể chen vào những vị trí mới làm mất đi thứ tự chuẩn, tấm ảnh chụp được không được như ý.

Ban tổ chức sắp xếp lại các vận động viên theo thứ tự chuẩn nhưng mọi việc diễn ra tương tự như trên. Sau 5 lần và thu được 5 tấm ảnh, Ban tổ chức đành bỏ cuộc và gửi 5 tấm ảnh cho chuyên gia Photoshop cắt dán lại theo thứ tự chuẩn.

Vấn đề đặt ra là Ban tổ chức đã quên mất thứ tự chuẩn, bạn cần dựa vào thứ tự trong 5 bức ảnh để xác định thứ tự chuẩn của Ban tổ chức. Biết rằng **không có vận động viên nào bị cảnh cáo nhiều hơn 1 lần**.

Ví dụ với $n = 9$, thứ tự chuẩn là (1,3,5,7,9,2,4,6,8)

Lần 1 (vận động viên 3 và 8) rời vị trí: (1, 3, 5, 7, 9, 2, 4, 6, 8) \rightarrow (1, 5, 7, 8, 9, 2, 4, 3, 6)

Lần 2 (vận động viên 2, 4 và 6) rời vị trí: (1, 3, 5, 7, 9, 2, 4, 6, 8) \rightarrow (2, 4, 6, 1, 3, 5, 7, 9, 8)

Lần 3 (vận động viên 1) rời vị trí: (1, 3, 5, 7, 9, 2, 4, 6, 8) \rightarrow (3, 5, 7, 9, 2, 4, 6, 8, 1)

Lần 4 (vận động viên 5) rời vị trí: (1, 3, 5, 7, 9, 2, 4, 6, 8) \rightarrow (1, 3, 7, 5, 9, 2, 4, 6, 8)

Lần 5 (vận động viên 7) rời vị trí: (1, 3, 5, 7, 9, 2, 4, 6, 8) \rightarrow (1, 7, 3, 5, 9, 2, 4, 6, 8)

Dữ liệu: Vào từ file văn bản PHOTO.INP

- ✿ Dòng 1 chứa số nguyên dương $n \leq 10^5$
- ✿ 5 dòng tiếp theo, dòng thứ i chứa n số nguyên là số hiệu các vận động viên trong bức ảnh thứ i theo đúng thứ tự trong ảnh

Kết quả: Ghi ra file văn bản PHOTO.OUT n số nguyên là số hiệu các vận động viên theo đúng thứ tự chuẩn muốn chụp

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

Ví dụ

PHOTO.INP	PHOTO.OUT
9	1 3 5 7 9 2 4 6 8
1 5 7 8 9 2 4 3 6	
2 4 6 1 3 5 7 9 8	
3 5 7 9 2 4 6 8 1	
1 3 7 5 9 2 4 6 8	
1 7 3 5 9 2 4 6 8	

RẢI SỎI

Người ta vẽ ra n ô trống đánh số từ 1 tới n sau đó đặt đúng m viên sỏi vào trong ô trống thứ k . Bạn được thực hiện thao tác: lấy một viên sỏi ở ô thứ k và chuyển nó sang một ô khác bất kỳ.

Yêu cầu: Hãy dùng ít thao tác nhất để rải các viên sỏi vào các ô, sao cho khi bạn kết thúc các thao tác, số sỏi trong hai ô liên tiếp bất kỳ chênh lệch nhau không quá 1. Cho biết số thao tác cần sử dụng

Dữ liệu: Vào từ file văn bản ARRANGE.INP

- ✿ Dòng 1 chứa số nguyên dương $T \leq 10^5$ là số test
- ✿ T dòng tiếp theo, mỗi dòng chứa một test là ba số nguyên dương n, m, k cách nhau bởi dấu cách ($n \leq 10^9; m \leq 10^9; 1 \leq k \leq n$)

Kết quả: Ghi ra file văn bản ARRANGE.OUT

Ứng với mỗi test, ghi ra số thao tác cần sử dụng theo phương án tìm được

Ví dụ

ARRANGE.INP	ARRANGE.OUT	Giải thích
2	5	Dãy kết quả:
4 8 3	3	0 0 8 0 \rightarrow 1 2 3 2
5 6 5		0 0 0 0 6 \rightarrow 0 0 1 2 3

TRÁO BÀI

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: Vào từ file văn bản SHUFFLE.INP

✿ 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 file văn bản SHUFFLE.OUT 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:

SHUFFLE.INP	SHUFFLE.OUT
9 2 3	7 8
1 5 2	
5 4 6	
8 2 1	