

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

MIỀN LỚN NHẤT

Cho đồ thị vô hướng G gồm n đỉnh đánh số từ 1 tới n và m cạnh đánh số từ 1 tới m . Cạnh thứ i nối giữa hai đỉnh u_i, v_i . Mỗi đỉnh được tô một trong n màu đánh số từ 1 tới n

Ta gọi tập đỉnh S là một miền nếu nó thỏa mãn hai điều kiện:

- ✿ Các đỉnh thuộc S phải cùng màu
- ✿ Giữa hai đỉnh bất kỳ thuộc S có đường đi chỉ qua các đỉnh $\in S$

Bạn được phép đổi màu một đỉnh nếu muốn. Hãy tìm cách để thu được miền lớn nhất (tính theo số đỉnh), cho biết số đỉnh của miền lớn nhất đó.

Dữ liệu: Vào từ file văn bản RECOLOR.INP

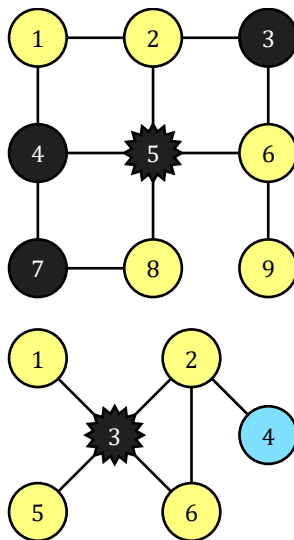
- ✿ Dòng đầu tiên chứa số nguyên dương $T \leq 10$ là số test
- ✿ T nhóm dòng tiếp theo mỗi nhóm dòng chứa dữ liệu một test theo khuôn dạng
 - ✿ Dòng 1 chứa hai số nguyên dương $n \leq 10^5; m \leq 2 \cdot 10^5$
 - ✿ Dòng 2 chứa n số nguyên, số thứ i là màu của đỉnh i .
 - ✿ m dòng tiếp theo, dòng thứ j chứa hai số nguyên u_j, v_j ứng với một cạnh của đồ thị

Các số trên một dòng của input file được ghi cách nhau bởi dấu cách

Kết quả: Ghi ra file văn bản RECOLOR.OUT một số nguyên duy nhất là số đỉnh của miền lớn nhất thu được.

Ví dụ

RECOLOR.INP	RECOLOR.OUT
2	6
9 11	5
1 1 2 2 2 1 2 1 1	
1 2	
1 4	
2 3	
2 5	
3 6	
4 5	
4 7	
5 6	
5 8	
6 9	
7 8	
6 6	
1 1 2 3 1 1	
1 3	
2 3	
2 4	
2 6	
3 5	
3 6	



CHIA ĐỘI

Kì thi Hackathon CSP sắp tới, Ban tổ chức muốn chia các thí sinh thành các đội chơi theo nguyện vọng của chính các thí sinh. Biết rằng, có n thí sinh tham gia cuộc thi. Thí sinh thứ i mong muốn đội của mình không ít hơn a_i người.

Để cuộc thi thêm nhiều gay cấn, Ban tổ chức mong muốn chia n thí sinh thành nhiều nhất các đội sao cho:

- ✿ Mỗi thí sinh chỉ thuộc một đội.
- ✿ Số lượng thí sinh thuộc mỗi đội không nhỏ hơn nguyện vọng của từng thí sinh thuộc đội đó
- ✿ Số lượng đội là nhiều nhất có thể. Nếu có nhiều phương án chia thành nhiều đội nhất, hãy chọn cách chia để số thí sinh thuộc đội nhiều nhất là ít nhất có thể.

Yêu cầu: Biết nguyện vọng của từng thí sinh, hãy cho biết Ban tổ chức có thể chia các bạn thí sinh thành tối đa bao nhiêu đội.

Dữ liệu: Vào từ file văn bản TEAMS.INP gồm:

- ✿ Dòng đầu ghi số nguyên dương n ($1 \leq n \leq 10^6$).
- ✿ Dòng tiếp theo ghi n số nguyên dương a_1, a_2, \dots, a_n ($1 \leq a_i \leq n$) là nguyện vọng của từng thí sinh

Kết quả: Ghi ra file văn bản TEAMS.OUT gồm:

- ✿ Dòng đầu tiên ghi số nguyên dương k – số đội lớn nhất có thể chia được thỏa mãn nguyện vọng của tất cả các thí sinh
- ✿ k dòng tiếp theo, mỗi dòng cho biết một đội: Đầu dòng là số lượng thí sinh trong đội, tiếp theo là các chỉ số của các thí sinh trong đội.

Ví dụ:

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

HACKATHON

SuperCoders là đội gồm 3 thành viên: A, B, C dự cuộc thi Hackathon của trường mầm non SuperKids. Theo thể thức của cuộc thi, mỗi đội chỉ được giao duy nhất một máy tính để làm k bài thi đánh số từ 1 tới k . Thời gian làm bài không hạn chế, nhưng phải làm xong hết k bài mới được xếp hạng. Đội nào xong càng sớm sẽ có thứ hạng càng cao.

Với luật thi như vậy, việc phân phối công việc kết hợp với nghỉ ngơi là điều hết sức quan trọng. Sau khi hội ý, các thành viên quyết định rằng mỗi bài sẽ chỉ giao cho một trong ba người làm và các bài sẽ được giải quyết một cách tuần tự trên máy tính duy nhất của đội:

- ✿ A sẽ làm đúng m bài, nếu A làm bài thứ i sẽ mất a_i giây
- ✿ B sẽ làm đúng n bài, nếu B làm bài thứ i sẽ mất b_i giây
- ✿ C sẽ làm đúng p bài, nếu C làm bài thứ i sẽ mất c_i giây

Ở đây $m + n + p = k$

Yêu cầu: Hãy giúp đội SuperCoders tìm ra cách phân công để mỗi người làm đúng số bài đã định sao cho tổng thời gian làm cả k bài là ít nhất.

Dữ liệu: Vào từ file văn bản HACKATHON.INP

- ✿ Dòng 1 chứa ba số nguyên dương $m, n, p \leq 10^5$
- ✿ $m + n + p$ dòng tiếp theo, dòng thứ i chứa ba số nguyên dương $a_i, b_i, c_i \leq 10^6$

Các số trên cùng một dòng của input được ghi cách nhau bởi dấu cách

Kết quả: Ghi ra file văn bản HACKATHON.OUT một số nguyên duy nhất là tổng thời gian làm tất cả các bài theo phương án phân công tối ưu tìm được.

Ví dụ

HACKATHON.INP	HACKATHON.OUT
2 3 4	36
1 3 8	
1 4 5	
1 5 6	
9 3 4	
9 4 5	
9 5 6	
6 6 6	
6 6 6	
6 6 6	

