

Bài 1. Rửa nhà kho

Vào những ngày đẹp trời trước đây, Farmer John phục vụ một kiểu ăn buồn chán cho N con bò chỉ bằng một loại thức ăn ($1 \leq N \leq 40000$). Thời gian trôi đi và các chú bò ngày càng buồn chán cách phục vụ kiểu này. Hôm nay, anh sẽ phục vụ đàn bò bằng M loại thức ăn khác nhau ($1 \leq M \leq N$), các loại thức ăn được đánh số từ 1 đến M .

Được vậy, các chú bò lại kén cá chọn canh. Con bò thứ i có một loại thức ăn khoái khẩu P_i ($1 \leq P_i \leq M$) và nó chỉ ăn loại thức ăn khoái khẩu đó.

FJ chuyển nhà kho đến một tiệm ăn trang nhã. Mỗi ngày đến giờ ăn, những chú bò xếp hàng bên ngoài để vào tiệm ăn theo một thứ tự đã định sẵn. Thật không may, với càng nhiều loại thức ăn, việc rửa nhà kho lại càng lâu. Nếu Farmer John phục vụ K loại thức ăn thì anh ta cần $K \cdot K$ đơn vị thời gian để rửa nhà kho.

Để tiếp kiệm thời gian, FJ phục vụ các con bò theo nhóm liên tiếp theo thứ tự xếp hàng. Sau mỗi nhóm, anh ta sẽ rửa nhà kho và phục vụ nhóm tiếp theo. Xác định tổng thời gian nhỏ nhất mà FJ cần sử dụng để rửa nhà kho. Mỗi nhóm gồm ít nhất một con bò được xếp hàng liên tiếp. Mỗi con bò thuộc đúng một nhóm. Nhà kho luôn được rửa sau mỗi nhóm.

Dữ liệu vào từ tệp: cleanup.in

- Dòng đầu tiên ghi số N và M
- Mỗi dòng trong N dòng tiếp theo, dòng thứ i ghi một số nguyên dương thể hiện loại thức ăn mà con bò xếp thứ i trong hàng đó thích.

Dữ liệu ra vào tệp: cleanup.out

- Ghi một số duy nhất là số

cleanup.in	cleanup.out
13 4	11
1	
2	<i>Giải thích:</i>
1	▪ Bốn nhóm đầu chỉ gồm một con.
3	▪ Nhóm 5 gồm 2 con
2	▪ Nhóm 6 gồm 4 con 3, 4, 3, 4, 3
2	▪ Nhóm 7 gồm 1 con
3	▪ Nhóm cuối gồm 1 con
4	<i>Tổng thời gian:</i>
3	$1+1+1+1+1+2^2+1+1=11$

4	
3	
1	
4	

Bài 2. Chia đội

Bạn cần phải làm chia một số người thành hai đội, theo cách sau:

- Mỗi người thuộc một trong hai đội.
- Mỗi team có ít nhất một người.
- Mỗi người trong một đội thì biết tất cả những người còn lại trong đội
- Hai đội có số lượng càng gần nhau càng tốt.

Input: Teamup.inp

Để đơn giản, tất cả mọi người được gán một số hiệu nằm trong khoảng từ 1 đến N ($2 \leq N \leq 100$).

Dòng đầu tiên chứa số N – Tổng số lượng người cần chia vào hai đội. Theo sau là N dòng – mỗi dòng là thông tin của một người theo thứ tự tăng dần. Mỗi dòng chứa một danh sách các số phân biệt A_{ij} ($1 \leq A_{ij} \leq N$, $A_{ij} \neq i$) cách nhau một dấu cách. Danh sách này là những người mà người thứ I biết, kết thúc bởi số 0. Dữ liệu đảm bảo là nếu i biết j thì j cũng biết i.

Output: Teamup.out

Nếu không tồn tại cách chia, in ra 1 dòng “No solution” (không có dấu nháy). Ngược lại in ra trị tuyệt đối của chênh lệch số lượng thành viên của hai đội

Ví dụ:

Teamup.inp	Teamup.out
5 2 3 5 0 1 4 5 3 0 1 2 5 0 1 2 3 0 4 3 2 1 0	1

Bài 3. Cửa hồi môn

Công chúa con vua xứ Flatland chuẩn bị làm đám cưới với một hoàng tử xinh đẹp của nước láng giềng. Vua cha dự định sẽ dành cho con gái yêu một món quà hồi môn ấn tượng, đó là một phần độ sưu tập đá quý của mình. Nhà vua có bộ sưu tập n viên đá quý, đánh số từ 1 đến n . Viên thứ i có trọng lượng w_i và giá trị v_i . Nhà vua muốn món quà càng có giá trị càng tốt nhưng trọng lượng cũng phải ở mức hợp lý – chỉ trong khoảng từ L đến R .

Yêu cầu: Hãy chỉ ra cách chọn các viên đá quý thỏa mãn điều kiện của nhà vua.

Dữ liệu: Vào từ file văn bản DOWRY.INP:

- Dòng đầu tiên chứa 3 số nguyên n, L và R ($1 \leq n \leq 32, 0 \leq L \leq R \leq 10^{18}$),
- Dòng thứ i trong n dòng sau chứa 2 số nguyên w_i và v_i ($1 \leq w_i, v_i \leq 10^{15}$).

Kết quả: Đưa ra file văn bản DOWRY.OUT:

- Dòng đầu tiên đưa ra số nguyên k – số viên đá được chọn,
- Mỗi dòng trong k dòng sau chứa số nguyên trong phạm vi từ 1 đến n xác định viên đá được chọn.

Nếu không thể chọn được món quà đạt yêu cầu thì đưa ra một dòng chứa số 0.

Ví dụ:

DOWRY.INP
3 6 8
3 10
7 3
8 2

DOWRY.OUT
1
2

Bài 4. Bookcase

Tom có n quyển sách, quyển thứ i có chiều cao h_i , chiều rộng t_i . Tom muốn làm một giá sách gồm có 3 tầng để có thể chứa hết tất cả n quyển sách.

Giả sử n quyển sách được phân thành 3 tập không rỗng

S_1, S_2, S_3 (các quyển sách thuộc tập S_i được xếp vào tầng i) thì cần giá sách chiếm diện tích bằng:

$$\left(\sum_{j=1}^3 \max_{i \in S_j} h_i \right) \times \left(\max_{j=1}^3 \sum_{i \in S_j} t_i \right)$$

Yêu cầu: Cần tìm cách phân n quyển sách thành 3 tập khác rỗng để giá sách chiếm diện tích nhỏ nhất.

Input

- Dòng 1: số nguyên dương T ($T \leq 20$) là số bộ dữ liệu
- T nhóm dòng sau, mỗi nhóm dòng có dạng:
 - o Dòng đầu là số nguyên dương n ($n \leq 70$)
 - o n dòng tiếp theo mỗi dòng 2 số h_i, t_i ($150 \leq h_i \leq 300, 5 \leq t_i \leq 30$)

Output

- Gồm n dòng, mỗi dòng ghi một số là diện tích nhỏ nhất của giá sách.

bookcase.inp	bookcase.out
1	18000
4	
220 29	
195 20	
200 9	
180 30	