

Mục lục

Kế hoạch định tuyến - ROUTING	1
Xe liên tỉnh - ICBUS	3
Lấy vàng — GOLD	4
Biển diễn đồ thị - GREP	5

Bài A. Kế hoạch định tuyến

File dữ liệu vào: `routing.inp`
File kết quả: `routing.out`
Hạn chế thời gian: 1 giây
Hạn chế bộ nhớ: 256 MB

Đảo Alpha là một hòn đảo đẹp như tranh vẽ và chỉ cho phép phương tiện giao thông duy nhất là xe đạp trên đảo. Do đó, có nhiều tuyến đường xe đạp kết nối các điểm chụp ảnh đẹp nhất trên đảo. Để giúp du khách có kế hoạch chuyến đi của họ tới hòn đảo này, ủy ban du lịch muốn thực hiện định tuyến cho đường xe đạp mà đi qua một số các điểm chụp ảnh đẹp nhất trên đảo.

Yêu cầu: Cho một bản đồ của tất cả những con đường xe đạp trên đảo và một danh sách những địa điểm chụp ảnh đẹp nhất trên mỗi tuyến đường quy hoạch (những điểm không được đề cập thì không xuất hiện trên tuyến đường đó), hãy giúp ủy ban du lịch lên kế hoạch định tuyến từng tuyến đường r để khoảng cách trên mỗi con đường là tối thiểu. Lưu ý rằng mỗi điểm chụp ảnh đẹp nhất chỉ có thể xuất hiện nhiều nhất một lần trên các tuyến đường.

Dữ liệu vào

Gồm hai phần. Phần đầu tiên cung cấp các thông tin của những tuyến đường xe đạp trên đảo. Dòng đầu tiên chứa hai số nguyên $n \leq 100$ và $r \leq 10$ là n điểm chụp ảnh đẹp nhất trên đảo và có r tuyến đường được lên kế hoạch.

n dòng tiếp theo (dòng 2 đến dòng thứ $n + 1$) chứa $n \times n$ số nguyên trong khoảng từ 0 đến 10 (n dòng với n số nguyên trên mỗi dòng), số nguyên thứ j trên dòng i là khoảng cách từ chỗ chụp ảnh đẹp nhất $i - 1$ đến chỗ chụp ảnh đẹp nhất j ; giá trị bằng 0 nghĩa là không có con đường một chiều đi từ chỗ chụp ảnh đẹp nhất $i - 1$ đến chỗ chụp ảnh đẹp nhất j .

Phần thứ hai gồm r dòng biểu thị r tuyến đường tham quan cần được quy hoạch. Mỗi dòng liệt kê các điểm chụp ảnh đẹp nhất cần đưa vào con đường tương ứng. Các số nguyên trên mỗi dòng biểu thị các địa điểm được đề nghị là các điểm chụp ảnh đẹp nhất trên tuyến đường tương ứng. Số nguyên đầu tiên trên dòng là điểm bắt đầu của tuyến đường và số nguyên cuối cùng là điểm dừng cuối cùng trên tuyến. Tuy nhiên, các điểm dừng ở giữa có thể được thăm theo thứ tự bất kì.

Kết quả

Đưa ra r số nguyên trên r dòng (mỗi số nguyên trên một dòng) là khoảng cách của từng tuyến đường được quy hoạch. Nếu có một tuyến đường không thể quy hoạch được thì in ra 0.

Ví dụ

routing.inp	routing.out
6 3 0 1 2 0 1 1 1 0 1 1 1 0 0 2 0 1 3 0 4 3 1 0 0 0 0 0 1 1 0 0 1 0 0 0 0 0 1 3 5 6 3 2 5 6 1 2 3 4 5	5 0 7

Bài B. Xe liên tỉnh

File dữ liệu vào: `icbus.inp`
File kết quả: `icbus.out`
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Quốc gia Backoi có N thành phố, mỗi thành phố có một hệ thống xe chạy liên tỉnh khác nhau. Một xe có thể chạy từ thành phố i sang thành phố j nếu như có đường nối trực tiếp giữa hai thành phố này. Các con đường ở đây đều là đường 2 chiều. Mỗi hệ thống xe liên tỉnh có một số luật như sau:

- Hành khách muốn sử dụng hệ thống xe của thành phố i thì bắt buộc phải bắt xe tại thành phố i .
- Giá vé xe của thành phố i là đồng hạng C_i bất kể quãng đường bao xa.
- Hệ thống xe của thành phố i chỉ cho phép chạy tối đa qua D_i thành phố.

Quân là một hành khách muốn đi từ thành phố 1 đến thành phố N . Hãy giúp Quân tìm cách đi sao cho tổng chi phí là thấp nhất.

Dữ liệu vào

Dòng đầu tiên chứa hai số nguyên dương N và K ($2 \leq N \leq 5000$; $N - 1 \leq K \leq 10000$).

N dòng tiếp theo, dòng thứ i chứa 2 số nguyên dương C_i và D_i ($1 \leq C_i \leq 10000$; $1 \leq D_i \leq N$) là 2 thông tin của hệ thống xe của thành phố i .

K dòng tiếp theo mỗi dòng ghi hai số i và j ($1 \leq i < j \leq N$) biểu thị giữa 2 thành phố i và j có đường nối trực tiếp.

Kết quả

Ghi ra duy nhất một số là chi phí Quân phải trả để đi từ thành phố 1 đến thành phố N . Dữ liệu đảm bảo luôn có cách đi từ thành phố 1 đến thành phố N .

Ví dụ

<code>icbus.inp</code>	<code>icbus.out</code>
6 6 400 2 200 1 500 3 900 1 400 4 200 5 1 2 1 5 2 3 2 4 3 6 4 6	800

Giải thích

Quân sử dụng lần lượt hệ thống xe của thành phố 1 rồi thành phố 5.

Bài C. Lầy vàng

File dữ liệu vào: gold.inp
File kết quả: gold.out
Hạn chế thời gian: 0.1 giây
Hạn chế bộ nhớ: 256 MB

Ông vua ở một vương quốc nọ có n kho vàng nằm trên một đường thẳng được đánh số theo thứ tự là $1, 2, \dots, n$, trong đó kho vàng i có trữ lượng vàng là a_i (a_i là số nguyên không âm) và đặt tại toạ độ i ($\forall i = 1, \dots, n$). Nhà vua mở cuộc thi tài cho những người tìm vàng xem ai là người có thể lựa chọn được dãy các kho vàng có tổng trữ lượng lớn nhất với điều kiện khoảng cách giữa 2 kho liên tiếp trong dãy các kho được chọn ra phải lớn hơn hoặc bằng L_1 và nhỏ hơn hoặc bằng L_2 , nghĩa là, nếu dãy kho được chọn là i_1, i_2, \dots, i_k ($i_1 < i_2 < \dots < i_k$), thì ta phải có $L_1 \leq i_j - i_{j-1} \leq L_2$, $j = 2, 3, \dots, k$.

Dữ liệu vào

Dòng 1 ghi giá trị n, L_1, L_2 cách nhau bởi dấu cách ($1 \leq n \leq 10^6, 1 \leq L_1 \leq L_2 \leq n$)

Dòng 2 ghi n giá trị nguyên a_1, a_2, \dots, a_n cách nhau bởi dấu cách.

Kết quả

Ghi ra duy nhất một số là tổng trữ lượng của các kho vàng được lựa chọn.

Ví dụ

gold.inp	gold.out
6 2 2	19
3 5 9 6 7 4	

Giải thích

Phương án tối ưu là lựa chọn dãy 3 kho vàng 1, 3 và 5 với tổng dự trữ là $3 + 9 + 7 = 19$

Bài D. Biểu diễn đồ thị

File dữ liệu vào: `grep.inp`

File kết quả: `grep.out`

Hạn chế thời gian: 0.1 giây

Hạn chế bộ nhớ: 256 MB

Bài toán yêu cầu chuyển đổi giữa các cách biểu diễn đồ thị và kiểm tra các đỉnh kề.

Dữ liệu vào

- Dòng đầu chứa ba số nguyên dương $n \leq 10^5$, $m \leq 10^5$ và $k \leq 10^5$;
- m dòng sau mỗi dòng chứa hai số nguyên dương u và v tương ứng với một cung (u, v) ;
- Dòng tiếp theo chứa k số nguyên dương w tương ứng với k yêu cầu liệt kê các đỉnh đầu của cung có đỉnh cuối là w theo thứ tự tăng dần chỉ số đỉnh.

Kết quả

- Dòng đầu chứa hai số n và m ;
- Dòng 2 chứa mảng Head gồm n giá trị: Head[i] là vị trí đầu của khoảng chứa thông tin các đỉnh kề với i trong mảng danh sách kề Last. Nếu đỉnh i không có đỉnh kề thì Head[i]=Head[$i + 1$];
- Dòng 3 in ra mảng Last gồm m số là thông tin các đỉnh kề trong danh sách kề;
- k dòng tiếp theo mỗi dòng in các đỉnh có chỉ số tăng dần là các đỉnh đầu của cung có đỉnh cuối là w tương ứng với yêu cầu trong file dữ liệu vào. Nếu không có cung đến w thì để dòng trống.

Ví dụ

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