



ĐỀ CHÍNH THỨC

(Đề thi gồm 04 trang)

TỔNG QUAN ĐỀ THI

Bài	Tên bài	File chương trình	Input/Output	Điểm
1	Sao chép ảnh	pics.*	Thiết bị chuẩn	100
2	Di chuyển robot	robot.*	Thiết bị chuẩn	100
3	Hành trình du lịch	dtour.*	Thiết bị chuẩn	100

Dấu * được thay bởi PAS, CPP, PY của ngôn ngữ lập trình tương ứng Pascal, C++ hoặc Python

Bài 1. Sao chép ảnh

Trong buổi họp lớp, Alice đã chụp được n bức ảnh, các bức ảnh được đánh số từ 1 đến n . Bức ảnh thứ i ($1 \leq i \leq n$) có kích thước s_i . Có m bạn trong lớp muốn nhờ Alice sao chép các bức ảnh, bạn thứ k ($1 \leq k \leq m$) sẽ đưa cho Alice hai ổ đĩa, ổ đĩa thứ nhất có sức chứa a_k , ổ thứ hai có sức chứa b_k và một danh sách L_k là các bức ảnh không cần sao chép. Bạn thứ k mong muốn có thể sao chép được nhiều bức ảnh nhất vào hai ổ đĩa gồm các bức ảnh không thuộc danh sách L_k . Alice không chắc chắn có thể sao chép được tối đa các bức ảnh theo cách tối ưu. Tuy nhiên, bạn thứ k vẫn vui vẻ nếu số lượng bức ảnh sao chép được không ít hơn $(g_k - 1)$, trong đó g_k là số lượng tối đa các bức ảnh có thể sao chép được theo cách tối ưu.

Yêu cầu: Hãy giúp Alice đưa ra kế hoạch sao chép các bức ảnh cho các bạn. Giả sử t_k là số lượng bức ảnh mà Alice có thể sao chép cho bạn thứ k ($1 \leq k \leq m$), giá trị này được chấp nhận nếu $(g_k - 1) \leq t_k \leq g_k$.

Dữ liệu: Vào từ thiết bị vào chuẩn (bàn phím) có khuôn dạng:

- Dòng đầu chứa hai số nguyên dương n, m ($n, m \leq 10^5$);
- Dòng thứ hai chứa n số nguyên dương $s_1, s_2, \dots, s_n$ ($s_i \leq 10^9$);
- Dòng thứ k ($1 \leq k \leq m$) trong m dòng sau, mỗi dòng có dạng:
 - o Hai số đầu tiên là a_k, b_k ($a_k, b_k \leq 10^9$);
 - o Số thứ ba là số $|L_k|$ là số bức ảnh mà bạn k không thích;
 - o Tiếp theo là $|L_k|$ số là chỉ số các bức ảnh thuộc danh sách L_k .

Tổng số lượng phần tử trong m danh sách không vượt quá 10^5 ($\sum_{k=1}^m |L_k| \leq 10^5$).

Kết quả: Ghi ra thiết bị ra chuẩn (màn hình) có khuôn dạng:

- Dòng đầu chứa số nguyên t_1 là số bức ảnh có thể sao chép cho bạn thứ nhất;
- Dòng thứ hai là một xâu độ dài n chỉ gồm các kí tự '0', '1', '2' mô tả cách sao chép các bức ảnh cho bạn thứ nhất, trong đó kí tự thứ i ($1 \leq i \leq n$) bằng '0' cho biết bức ảnh thứ i không được sao chép, bằng '1' hoặc '2' cho biết bức ảnh i được sao chép vào đĩa 1 hoặc đĩa 2;
- Do dữ liệu ghi ra quá lớn nên với các bạn còn lại chỉ cần đưa ra số lượng bức ảnh có thể sao chép được. Cụ thể, dòng thứ ba chứa $m - 1$ số $t_2, t_3, \dots, t_m$.

Ví dụ:

Input	Output	Giải thích
6 3 2 1 1 3 4 1 5 5 0 5 5 2 2 6 5 7 0	5 111022 4 5	Với bạn thứ 1, tối đa các bức ảnh có thể sao chép được là 5 (ví dụ, đĩa 1 chứa bức ảnh 1, 2, 3; đĩa 2 chứa bức ảnh 5, 6). Kết quả đưa ra bằng 4 hoặc 5 đều được chấp nhận. Với bạn thứ 2, tối đa các bức ảnh có thể sao chép được là 4 (ví dụ, đĩa 1 chứa bức ảnh 1, 4; đĩa 2 chứa bức ảnh 3, 5). Kết quả đưa ra bằng 3 hoặc 4 đều được chấp nhận. Với bạn thứ 3, tối đa các bức ảnh có thể sao chép được là 6 (ví dụ, đĩa 1 chứa bức ảnh 1, 4; đĩa 2 chứa bức ảnh 2, 3, 5, 6). Kết quả bằng 5 hoặc 6 đều được chấp nhận.

Giới hạn:

- **Subtask 1 (25%):** $n \leq 10; m \leq 3; a_k, b_k \leq 1000; |L_k| = 0$.
- **Subtask 2 (25%):** $n \leq 50; m \leq 3; a_k, b_k \leq 1000$.
- **Subtask 3 (25%):** $|L_k| = 0$.
- **Subtask 4 (25%):** Không có ràng buộc nào thêm.

Bài 2. Di chuyển robot

Alice thiết kế trò chơi điều khiển robot như sau: Một robot đặt trên một sân được biểu diễn như một lưới ô vuông kích thước $n \times m$. Các dòng được đánh số từ 0 đến $n - 1$, các cột được đánh số từ 0 đến $m - 1$. Ô nằm giao giữa hàng i cột j được gọi là ô (i, j) , một số ô của lưới là tường, các ô còn lại là ô tự do. Người chơi điều khiển robot bằng bốn loại lệnh: U, D, L, R, giả sử robot đang đứng tại ô (x, y) , robot sẽ di chuyển tương ứng như sau:

- Lệnh U: Nếu $x = 0$ hoặc $(x - 1, y)$ là tường, robot sẽ không di chuyển. Ngược lại, robot sẽ di chuyển đến $(x - 1, y)$

- Lệnh D: Nếu $x = n - 1$ hoặc $(x + 1, y)$ là tường, robot sẽ không di chuyển. Ngược lại, robot sẽ di chuyển đến $(x + 1, y)$
- Lệnh L: Nếu $y = 0$ hoặc $(x, y - 1)$ là tường, robot sẽ không di chuyển. Ngược lại, robot sẽ di chuyển đến $(x, y - 1)$
- Lệnh R: Nếu $y = m - 1$ hoặc $(x, y + 1)$ là tường, robot sẽ không di chuyển. Ngược lại, robot sẽ di chuyển đến $(x, y + 1)$.

Người chơi không được biết chính xác vị trí ban đầu của robot, chỉ biết rằng robot có thể đang ở một trong k ô tự do $(u_1, v_1), (u_2, v_2), \dots, (u_k, v_k)$.

Yêu cầu: Hãy tìm một dãy lệnh điều khiển robot để robot luôn kết thúc tại vị trí $(0, 0)$.

Dữ liệu: Vào từ thiết bị vào chuẩn (bàn phím) có khuôn dạng:

- Dòng đầu chứa ba số nguyên dương n, m, k ($n, m \leq 200; k \leq 10$);
- Dòng thứ i trong n dòng tiếp theo chứa m số mô tả lưới, số thứ j bằng 0 (hoặc 1) tương ứng ô (i, j) là không có tường (hoặc có tường);
- Dòng thứ t trong k dòng tiếp theo, mỗi dòng chứa hai số u_t, v_t .

Dữ liệu đảm bảo bài toán luôn có cách di chuyển thỏa mãn.

Kết quả: Ghi ra thiết bị ra chuẩn (màn hình) một dòng chứa một xâu chỉ gồm các kí tự L, R, U, D mô tả dãy lệnh.

Ví dụ:

Input	Output	Giải thích
2 2 2 0 0 1 0 0 1 1 1	UL	Nếu ban đầu robot ở ô $(0, 1)$ thì vị trí robot sau mỗi lệnh tương ứng như sau: $(0,1) \rightarrow (0,1) \rightarrow (0, 0)$ Nếu ban đầu robot ở ô $(1, 1)$ thì vị trí robot sau mỗi lệnh tương ứng như sau: $(1,1) \rightarrow (0,1) \rightarrow (0, 0)$

Giới hạn:

- **Subtask 1 (15%):** $k = 1$ và tất cả các ô đều là ô tự do.
- **Subtask 2 (25%):** $k = 1$.
- **Subtask 3 (30%):** $m, n \leq 10; k \leq 3$.
- **Subtask 4 (30%):** Không có ràng buộc nào thêm.

Bài 3. Hành trình du lịch

Đất nước của Alice gồm có n thành phố và $n - 1$ con đường hai chiều giữa các thành phố. Hệ thống đường đảm bảo từ thành phố bất kì có thể đi đến được thành phố bất kì khác. Để phát triển du lịch, chính phủ đã thực hiện phương án như sau:

- Một danh sách gồm k cặp thành phố $(u_1, v_1), (u_2, v_2), \dots, (u_k, v_k)$ được công bố;
- Định nghĩa một hành trình du lịch được gọi là chứa cặp thành phố (u, v) nếu hành trình đó thăm tất cả các thành phố nằm trên đường đi ngắn nhất giữa hai thành phố

u, v (bao gồm cả hai thành phố u, v). Nếu một hành trình du lịch chứa ít nhất một cặp trong k cặp thành phố đã công bố thì hành trình du lịch đó sẽ được hỗ trợ giá 50%.

Alice đang xây dựng một hành trình du lịch xuất phát tại thành phố s , đi dọc theo các con đường và mong muốn thăm mỗi thành phố không quá một lần nhưng phải chứa ít nhất một cặp thành phố đã công bố để được hỗ trợ giá.

Yêu cầu: Cho q giả định, với một giả định mô tả bằng thành phố s , hãy cho biết có bao nhiêu thành phố f ($f \neq s$) mà hành trình du lịch là đường đi từ thành phố s đến thành phố f chứa ít nhất một cặp thành phố đã công bố và không có thành phố nào đi quá một lần.

Dữ liệu: Vào từ thiết bị vào chuẩn (bàn phím) có khuôn dạng:

- Dòng đầu chứa ba số nguyên dương n, k, q ($n, k, q \leq 2 \times 10^5$);
- Tiếp theo là $n - 1$ dòng, mỗi dòng chứa hai số i, j mô tả con đường giữa hai thành phố i, j ;
- Dòng thứ t ($1 \leq t \leq k$) trong k dòng sau chứa hai số nguyên dương u_t, v_t ($u_t \neq v_t$) mô tả cặp thành phố được công bố;
- Dòng tiếp theo chứa q số, mỗi số mô tả một giả định.

Kết quả: Ghi ra thiết bị ra chuẩn (màn hình) một dòng gồm q số là câu trả lời cho q giả định.

Ví dụ:

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

Giới hạn:

- **Subtask 1 (15%):** $n, k, q \leq 200$ và hệ thống đường có dạng đường thẳng với các thành phố lần lượt từ 1 đến n (thành phố i có đường nối đến thành phố $i + 1$).
- **Subtask 2 (20%):** $n, k, q \leq 200$.
- **Subtask 3 (20%):** Hệ thống đường có dạng đường thẳng.
- **Subtask 4 (45%):** Không có ràng buộc nào thêm.

----- HẾT -----

Họ và tên thí sinh: Số báo danh:

Lưu ý: Thí sinh **không** được sử dụng tài liệu; Cán bộ coi thi **không** giải thích gì thêm.