

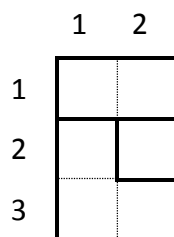
1. PHÁ TƯỜNG

Có một toà lâu đài hình chữ nhật kích thước $m \times n$ ô vuông đơn vị. Các dòng ô vuông được đánh số từ 1 tới m từ trên xuống dưới và trên mỗi dòng, các ô được đánh số theo thứ tự từ 1 tới n từ trái qua phải. Trong lâu đài có một số bức tường giữa các ô, định nghĩa một đường đi giữa hai ô là một cách di chuyển từ ô này tới ô kia qua các ô kề cạnh mà hai ô kề cạnh trong mỗi bước di chuyển không bị tường chắn ở giữa hai ô đó. Các bức tường phân lâu đài ra làm nhiều phòng, mỗi phòng gồm một số ô. Cách chia phòng thoả mãn hai điều kiện:

- Tồn tại đường đi giữa hai ô bất kỳ trong cùng một phòng
- Không tồn tại đường đi giữa hai ô thuộc hai phòng khác nhau

Giả thiết rằng quanh lâu đài có tường bao bọc.

Ví dụ dưới đây là một lâu đài 3×2 có 3 phòng.



Hãy cho biết lâu đài có bao nhiêu phòng, cho biết diện tích của phòng rộng nhất và tìm cách phá đi đúng một bức tường để được một phòng rộng nhất có thể

Dữ liệu: Vào từ file văn bản DWALL.INP

- Dòng 1: Ghi hai số m, n ($1 \leq m, n \leq 1000$)
- Các dòng tiếp theo, mỗi dòng ghi 4 số nguyên x_1, y_1, x_2, y_2 cho ta thông tin: có tường ngăn ô (x_1, y_1) và ô (x_2, y_2) . Có ít nhất một bức tường ngăn các ô trong lâu đài.

Kết quả: Ghi ra file văn bản DWALL.OUT

- Dòng 1: Ghi số phòng của lâu đài
- Dòng 2: Ghi diện tích của phòng rộng nhất
- Dòng 3: Ghi bốn số nguyên u_1, v_1, u_2, v_2 cho biết cần phá tường ngăn ô (u_1, v_1) và ô (u_2, v_2) . Bắt buộc phải phá đúng một bức tường.
- Dòng 4: Ghi diện tích của phòng rộng nhất thu được sau khi phá tường

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

Ví dụ:

DWALL.INP	DWALL.OUT
3 2	3
1 1 2 1	3
1 2 2 2	1 1 2 1
2 1 2 2	5
2 2 3 2	

2. ĐƯỜNG MỘT CHIỀU

Một hệ thống giao thông có n địa điểm đánh số từ 1 tới n và m tuyến đường đánh số từ 1 tới m . Tuyến đường thứ i nối từ địa điểm u_i tới địa điểm v_i , cho phép đi từ u_i tới v_i theo **một chiều**.

Ta nói địa điểm s có thể đi tới địa điểm t nếu tồn tại dãy $s = p_0, p_1, \dots, p_k = t$ sao cho có tuyến đường cho phép đi từ p_{i-1} tới p_i ($\forall i = 1, 2, \dots, k$)

Yêu cầu: Hãy xây dựng thêm một số ít nhất các tuyến đường một chiều để hệ thống giao thông đảm bảo được sự đi lại giữa hai địa điểm bất kỳ.

Dữ liệu: Vào từ file văn bản TRAFFIC.INP

- Dòng 1: Chứa hai số nguyên dương $n \leq 10^5$; $m \leq 2 \cdot 10^5$
- m dòng tiếp theo, dòng thứ i chứa hai số nguyên dương u_i, v_i

Kết quả: Ghi ra file văn bản TRAFFIC.OUT

- Dòng 1: Ghi số k là số tuyến đường cần xây dựng thêm
- k dòng tiếp theo, mỗi dòng ghi hai số x, y tương ứng với một tuyến đường cần xây dựng thêm để đi theo một chiều từ x tới y

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

Ví dụ

TRAFFIC.INP	TRAFFIC.OUT
7 7	3
1 2	4 5
2 1	7 3
2 3	3 2
2 4	
5 6	
6 7	
7 8	

