

UCV2013H - Slick

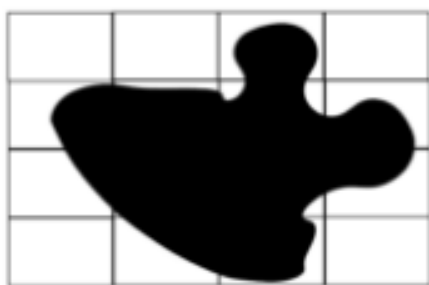
Link submit: <http://www.spoj.com/problems/UCV2013H/>

Solution:

C++	https://ideone.com/rl2DPR
Java	https://ideone.com/TpMwJN
Python	https://ideone.com/ZHOn3c

Tóm tắt đề:

Cho một hình ảnh thu được từ vệ tinh, hình này chuyển thành 0 là nước, 1 là dầu như hình.



(a)

0	0	1	0
1	1	1	1
1	1	1	1
0	1	1	0

(b)

Cho ma trận nhị phân, công việc của bạn là đếm có bao nhiêu vũng dầu (slick) và kích thước tương ứng của chúng. Hai ô lân cận được coi là trong cùng một slick nếu chúng nằm trong cùng một hàng hoặc cùng một cột.

Input:

Có nhiều bộ test, mỗi bộ test ứng với một hình ảnh thu được từ vệ tinh đã chuyển thành 0, 1, bắt đầu bằng hai số n, m ($1 \leq n, m \leq 250$) thể hiện số dòng và số cột trong hình.

n dòng tiếp theo, mỗi dòng có m số chứa thông tin của hình.

Dữ liệu kết thúc bằng 2 số 0 0.

Output:

Đối với mỗi hình, xuất số lượng slicks trên biển. Đồng thời, xuất kích thước của mỗi slick theo thứ tự tăng dần và số slick của kích thước đó.

Ví dụ:

10 10	7
1 1 1 1 1 1 1 1 1 1	1 2
1 1 1 1 0 0 0 0 0 0	2 1
1 1 1 0 0 0 0 1 1 1	6 1
1 1 0 0 1 0 0 1 1 1	10 2
1 0 1 0 0 1 1 0 0 0	20 1
0 0 0 0 0 0 0 0 0 0	
0 0 0 0 0 0 0 0 0 0	
1 1 1 1 1 1 1 1 1 1	
0 0 0 0 0 0 0 0 0 0	
1 1 1 1 1 1 1 1 1 1	
0 0	

Hướng dẫn giải:

Ta có thể thấy rằng hai ô (x, y) và ô (i, j) sẽ đi được với nhau nếu chúng có chung cạnh. Do đó, nếu có thể xem mỗi ô trên ma trận là một đỉnh của một đồ thị, thì hai đỉnh trên đồ thị sẽ có một cạnh nối với nhau nếu chúng thỏa mãn ba tính chất sau đây:

- Hai ô được biểu diễn bởi hai đỉnh của đồ thị phải nằm trong phạm vi ma trận $N \times M$.
- Hai ô, không ô nào được phép biểu diễn bởi số 0 (cả hai ô đều là số 1).
- Hai ô phải chung cạnh với nhau.

Vì số lượng ô tối đa của một vùng dầu là $N \times M$, nên để sắp xếp kích thước các slick theo thứ tự tăng dần và số slick của kích thước đó, ta chỉ cần tạo một mảng đếm $N \times M$ phần tử, sau đó chỉ cần duyệt từ 1 đến $N \times M$ và in số lượng vùng dầu có kích thước tương ứng.

Như vậy, sau khi ta xây dựng được một đồ thị vô hướng, ý tưởng của ta sẽ như sau:

- Xác định vị trí của ô số 1, giả sử là ô (sx, sy) .
- Với mỗi ô (i, j) trong ma trận mà được biểu diễn bằng số 1, ta sẽ sử dụng kỹ thuật duyệt DFS hoặc BFS từ ô (i, j) để xem có thể đến được ô (sx, sy) hay không, nếu có thì ta sẽ cập nhật kích thước của vùng dầu đó lên 1. Sau khi duyệt hết tất cả ô liền kề có số 1, ta cập nhật số lượng vùng dầu có kích thước tương ứng lên 1.

Độ phức tạp:

Time Complexity: $O(N * M)$ với N và M lần lượt là độ dài của 2 cạnh của bảng.

Space Complexity: $O(N * M)$.