



Graph Connectivity

Link submit:

https://uva.onlinejudge.org/index.php?option=com_onlinejudge&Itemid=8&page=show_problem&problem=400

Solution:

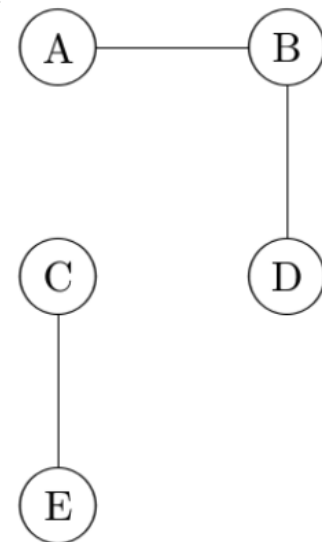
C++	https://ideone.com/67zr6q
Java	https://ideone.com/e141cd
Python	https://ideone.com/l1pEMi

Tóm tắt đề:

Một đồ thị gọi là liên thông nếu từ một đỉnh bất kì có thể đi đến tất cả các đỉnh còn lại trong đồ thị.

Một đồ thị con liên thông là một tập các đỉnh và các cạnh nối giữa các đỉnh trong tập con đó và đồng thời đồ thị con đó cũng liên thông. Ví dụ như hình bên ta có các tập $\{A\}$, $\{B\}$, $\{C\}$, $\{D\}$, $\{E\}$, $\{A, B\}$, $\{B, D\}$, $\{C, E\}$, $\{A, B, D\}$.

Một đồ thị con liên thông là cực đại nếu như nó là một đồ thị con liên thông và ta không thể thêm các đỉnh và cạnh từ đồ thị ban đầu vào đồ thị con này mà vẫn giữ nguyên tính liên thông của nó. Ví dụ như hình bên ta có $\{A, B, D\}$ và $\{C, E\}$ là hai đồ thị con cực đại, còn đồ thị con $\{A, B\}$ không thỏa vì có thể thêm đỉnh D vào tập, hay $\{A, C\}$ không thỏa vì chúng không liên thông.



Yêu cầu: cho một đồ thị vô hướng, đếm số đồ thị con liên thông cực đại của nó.

Input:

Dòng đầu gồm một số nguyên dương T là số lượng bộ test. Mỗi bộ test có định dạng như sau:

- Dòng đầu tiên là một kí tự alphabet in hoa – là nhãn của đỉnh có thứ tự từ điển lớn nhất trong đồ thị.
- Tiếp theo là một số dòng, mỗi dòng là một cặp hai kí tự thể hiện cạnh nối giữa hai đỉnh có nhãn tương ứng.
- Dữ liệu của test case kết thúc khi gặp dòng trống.

Output:

Với mỗi test case in ra trên một dòng là số lượng đồ thị con liên thông cực đại.

Ví dụ:

1	2
E	
AB	
CE	
DB	
EC	

Giải thích ví dụ:

Hình ảnh của đồ thị là ví dụ đề bài.

Hướng dẫn giải:

Sử dụng DSU, khi gặp các cạnh thì ta thực hiện gộp hai tập hợp chứa hai đỉnh của cạnh lại với nhau thành một thành phần liên thông lớn. Đến cuối thì ta duyệt qua danh sách đỉnh, nếu đỉnh nào có parent là chính nó (hoặc -1, tùy giá trị khởi tạo) thì nghĩa là đây là đỉnh đại diện của một tập hợp mới (một thành phần liên thông cực đại). Vậy kết quả của ta chính là số lượng đỉnh như vậy.

Độ phức tạp: $O(T*N*M)$ trong trường hợp sử dụng DSU cơ bản, nếu sử dụng DSU nâng cao (update by rank, path compression, by size) thì sẽ là $O(T*M*\log N)$ với T là số bộ test, N là số đỉnh của đồ thị và M là số cạnh tối đa của đồ thị.