

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

Thuật toán Tarjan tìm

thành phần liên thông



04/08/2017

mạnh

11:13



Bài gốc

[SPOJ TJALG](#)

Đề bài

Cho đồ thị $G(V,E)$ có hướng n ($1 \leq n \leq 10^4$) đỉnh m ($1 \leq m \leq 10^5$) cung, Hãy đếm số thành phần liên thông mạnh của G .

Giới hạn:

- $1 \leq n \leq 10000$
- $1 \leq m \leq 100000$

Định dạng test

Input:

- Dòng đầu tiên là n, m .
- M dòng tiếp theo mô tả một cung của G .

Output:

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

- Gồm một dòng duy nhất là số TPLT mạnh.

Ví dụ:

input:

3 2

1 2

2 3

output:

3

Thuật toán

Tarjan.

Source code

Ngôn ngữ: C++.

```
1  #include
2  <iostream>
3  #include <cstring>
4  #include <vector>
5  using namespace
6  std;
7
8  int n, m,
9  num[10005],
10 low[10005],
11 cnt=0,
12 connect[10005],
13 numSCC=0;
```

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

```
14 vector<int>
15 a[10005], S;
16
17 void dfs(int u) {
18     low[u] =
19     num[u] = cnt++;
20
21     S.push_back(u);
22     connect[u] =
23     1;
24     for(int v :
25     a[u]) {
26         if (num[v]
27         == -1) dfs(v);
28         if
29         (connect[v])
30         low[u] =
31         min(low[u],
32         low[v]);
33     }
34
35     if (num[u] ==
36     low[u]) {
37         numSCC++;
38         while(1) {
39             int v
40             = S.back();
41             S.pop_back();
42
43             connect[v] = 0;
44             if (u
45             == v) break;
```

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

```
45         }  
46     }  
    }  
  
    int main() {  
  
        cin >> n >> m;  
        for(int i = 0;  
i < m; i++) {  
            int u, v;  
            cin >> u >> v;  
  
            a[u].push_back(v);  
        }  
  
        RESET(num,  
-1);  
        RESET(low, 0);  
        RESET(connect,  
0);  
        for(int u = 1;  
u <= n; u++)  
            if (num[u]  
== -1) dfs(u);  
  
        cout <<  
numSCC;  
  
        return 0;  
    }
```

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

Tính chất và ứng dụng

- $num[u]$: là thứ tự duyệt dfs đến đỉnh u
- $low[u]$: là num nhỏ nhất trong tập những đỉnh mà u có thể đi đến.

Khởi tạo thì $low[u] = num[u]$, $low[u]$ sẽ thay đổi khi có một cạnh tạo nên chu trình trong đồ thị.

- Có 2 mảng để đánh dấu, $num[]$ và $connect[]$.
 - Mảng $num[]$ vừa để lưu thứ tự duyệt, vừa để kiểm tra xem một đỉnh u đã được duyệt đến hay chưa
 - Mảng $connect[]$ dùng để kiểm tra xem đỉnh v có còn được “kết nối” trong đồ thị hay không? Nếu phát hiện ra một thành phần liên thông mạnh, và một đỉnh v có trong thành phần liên thông mạnh đó, thì ta loại đỉnh v này

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

ra khỏi đồ thị bằng
câu lệnh `connect[v] = 0`, điều này là quan trọng vì để tránh gây ảnh hưởng đến việc nén mảng `low[]` của những đỉnh khác vẫn còn nằm trong đồ thị.

Độ phức tạp

 $O(M + N)$

Post Directory

[Bài gốc](#)[Đề bài](#)[Định dạng test](#)[Thuật toán](#)[Source code](#)[Tính chất và ứng dụng](#)[Độ phức tạp](#)