

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

# Thuật toán Kruskal tìm

## cây khung nhỏ nhất



03/08/2017

20:23



## Bài gốc

[SPOJ QBMST](#)

## Đề bài

Cho đơn đồ thị vô hướng liên thông  $G = (V, E)$  gồm  $n$  đỉnh và  $m$  cạnh, các đỉnh được đánh số từ 1 tới  $n$  và các cạnh được đánh số từ 1 tới  $m$ , mỗi cạnh có trọng số không âm  $c$ . Hãy tìm cây khung nhỏ nhất của đồ thị  $G$

Giới hạn:

- $1 \leq n \leq 10000$
- $1 \leq m \leq 15000$
- $0 \leq c \leq 10000$

## Định dạng test

Input:

- Dòng 1: Chứa hai số  $n, m$ .
- $M$  dòng tiếp theo, dòng thứ  $i$  có dạng ba số

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

nguyên  $u, v, c$ . Trong đó  $(u, v)$  là chỉ số hai đỉnh đầu mút của cạnh thứ  $i$  và  $c$  trọng số của cạnh đó.

Output:

- Gồm 1 dòng duy nhất: Ghi trọng số cây khung nhỏ nhất.

Ví dụ:

input:

6 9

1 2 1

1 3 1

2 4 1

2 3 2

2 5 1

3 5 1

3 6 1

4 5 2

5 6 2

---

output:

5

## Thuật toán

Kruskal.

## Source code

Ngôn ngữ: C++.

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

```
1  const int N = 1000
2  const int M = 1500
3  int n, m, lab[N];
4  pair<ll, ii> e[M];
5
6  int root(int u) {
7      return (lab[u]
8  ? u : lab[u] =
9  root(lab[u]));
10 }
11
12 void merge(int u,
13 {
14     u = root(u);
15     v = root(v);
16     if (u == v) re
17     if (lab[u] > l
18 swap(u, v);
19     lab[u] += lab[
20     lab[v] = u;
21 }
22
23 int main() {
24
25
26 ios::sync_with_std
27     cin.tie(0);
28 cout.tie(0);
29
30     // input
31     cin >> n >> m;
32     FOR(i,1,m)
```

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

```

33     cin >> e[i].se
34 e[i].se.se >> e[i]
35
36     // dsu init
37     memset(lab, -1
38 sizeof lab);
39
40     // sort edges
41     sort(e+1, e+m+
42
43     // build mst
44     ll cost = 0; /
45 minimum cost of ms
46     ll mste = 0; /
47 number of mst
48     FOR(i,1,m) {
49         if (mste =
50 1) break;
51
52         int u =
53 e[i].se.fi;
54         int v =
55 e[i].se.se;
56         ll c = e[i]
57
58         if (root(u
59 root(v)) {
60             cost +
61             mste++
62             merge(
63
64         }
65     }

```

Nhan Nguyen

nhannguyenmath95@gmail.com



Home



Algorithms



Problems

```
// output  
cout << cost;
```

```
return 0;
```

```
}
```

## Độ phức tạp

Xấp xỉ  $O(M \log M + M \log N)$

## Post Directory

[Bài gốc](#)[Đề bài](#)[Định dạng test](#)[Thuật toán](#)[Source code](#)[Độ phức tạp](#)