

# 最小生成树计数

## 题意

给出了一个简单无向加权图。你不满足于求出这个图的最小生成树，而希望知道这个图中有多少个不同的

最小生成树。（如果两颗最小生成树中至少有一条边不同，则这两个最小生成树就是不同的）。由于不同的最小生

成树可能很多，所以你只需要输出方案数对31011的模就可以了。

## 思路

这道题关键的地方理解不够透彻

姑且当一个模板题来记录

一棵生成树，通过增加一条边后，会形成一个回路，删除回路上另外一条边，就得到了另外一棵生成树

如果一个最小生成树，任意加一条边后，形成的环，其他边一定都不大于加的这条边，所以如果仍然要是一棵最小生成树，换掉的边一定是和加的这条边相等的

所以可以得出结论：所有的最小生成树，每种边权的个数不变

我们可以枚举不同的边权，用深搜来找出有多少种选择方案（实现不甚理解）

最后通过乘法原理，得到我们要的答案。

## 代码

```
1 #include<bits/stdc++.h>
2 using namespace std;
3 typedef long long LL;
4 struct edge{
5     int x,y,w;
```

```

6
7 }e[1005];
8 inline bool cmp(edge a,edge b)
9 {
10     return a.w<b.w;
11 }
12 map<int,int> be;
13 map<int,int> en;
14 map<int,int> choose_num;
15 int fa[105];
16 int getf(int a)
17 {
18     if(a == fa[a])
19         return a;
20     return getf(fa[a]);
21 }
22
23 int sum;
24 void dfs(int bian,int now,int k)
25 {
26     if(now == en[bian]+1)
27     {
28         if(k == choose_num[bian])
29         {
30             sum++;
31         }
32         return ;
33     }
34     int p =getf(e[now].x);
35     int q = getf(e[now].y);
36     if(p!=q)
37     {
38         fa[p] = q;
39         dfs(bian,now+1,k+1);
40         fa[p] = p;
41         fa[q] = q;
42     }
43     dfs(bian,now+1,k);
44 }
45 int main()
46 {
47     ios::sync_with_stdio(false);
48     int n,m;
49     cin>>n>>m;
50     for(int i=1;i<=m;i++)
51     {

```

```

52     cin>>e[i].x>>e[i].y>>e[i].w;
53 }
54 for(int i=1;i<=n;i++)
55 {
56     fa[i] = i;
57 }
58 sort(e+1,e+m+1,cmp);
59 e[0].w= 0;
60 int tot = 0;
61 for(int i=1;i<=m;i++)
62 {
63     if(e[i].w!=e[i-1].w)
64     {
65         be[e[i].w] = i;
66         if(i!=1)
67             en[e[i-1].w] = i-1;
68     }
69     int p = getf(e[i].x);
70     int q = getf(e[i].y);
71     if(p!=q)
72     {
73         fa[p] = q;
74         choose_num[e[i].w]++;
75         tot++;
76     }
77 }
78 en[e[m].w] = m;
79 if(tot<n-1) {
80     cout<<"0"<<endl;
81     return 0;
82 }
83 for(int i=1;i<=n;i++)
84 {
85     fa[i] = i;
86 }
87 int ans = 1;
88 for(map<int,int>::iterator it =
choose_num.begin();it!=choose_num.end();it++)
89 {
90     int bian = it->first;
91     sum = 0;
92     dfs(bian,be[bian],0);
93     ans*=sum;
94     ans%=31011;
95     for(int i = be[bian];i<=en[bian];i++)
96     {

```

```
97         int p=getf(e[i].x),q=getf(e[i].y);
98         if(p!=q)fa[p]=q;
99     }
100 }
101 cout<<ans<<endl;
102 }
```