

## 原创 POJ - 3164 Command Network (最小树形图)

2019-10-04 14:07:59 \_Y-\_Y- 阅读数 3 更多

第

版权声明：本文为博主原创文章，遵循 CC 4.0 BY-SA 版权协议，转载请附上原文出处链接和本声明。

本文链接：[https://blog.csdn.net/weixin\\_44410512/article/details/102069721](https://blog.csdn.net/weixin_44410512/article/details/102069721)

### Command Network

题意：在有向图中找出最小生成树（最小树形图）

```

1  /*
2  Command Network
3  POJ - 3164
4  https://vjudge.net/problem/POJ-3164
5  题意：在有向图中找出最小生成树（最小树形图）
6  解法：朱, 刘算法
7  算法思路：
8  1: 确定一个根
9  2: 找到除根外每一个点的最小入边，若这些边构成了环，则缩环成点，并将环内的每一个点的其他入边都减去环内的入边
10 3: 重复步骤2直到没有环出现（构成了树）。
11 */
12 #include <iostream>
13 #include <cstdio>
14 #include <cstring>
15 #include <cmath>
16 using namespace std;
17 #define maxn 10000
18 #define INF 0x3f3f3f3f
19 int X[maxn],Y[maxn];
20 double IN[maxn];
21 int PRE[maxn];
22 int ID[maxn];
23 int VIS[maxn];
24 int n,m,cnt;
25 struct node{
26     int u,v;
27     double w;
28 }edge[maxn];
29 double dtc(int a,int b){
30     return sqrt((double)((X[a]-X[b])*(X[a]-X[b])+(Y[a]-Y[b])*(Y[a]-Y[b])));
31 }
32 void add(int u,int v){
33     edge[cnt].u=u;
34     edge[cnt].v=v;
35     edge[cnt++].w=dtc(u,v);
36 }
37 double zhuliu(int root){
38     double ans=0;
39     while(1){
40         for(int i=1;i<=n;i++){
41             IN[i]=INF;
42         }
43         for(int i=0;i<cnt;i++){
44             if(edge[i].v==edge[i].u) continue;
45             else if(IN[edge[i].v]>edge[i].w){
46                 PRE[edge[i].v]=edge[i].u;//记录每个点的最小入边
47                 IN[edge[i].v]=edge[i].w;//记录每个点的最小入边的边权
48             }
49         }
50         for(int i=1;i<=n;i++){
51             if(i!=root&&IN[i]==INF) return -1;
52         }
53         int num=0;
54         memset(ID,0,sizeof ID);
55         memset(VIS,0,sizeof VIS);
56         //ID数组的初始化要与cnt记录的环数相匹配，这里ID用0初始化，则cnt从1开始记录，也可以用-1初始化，让cnt从0开始记录
57         //初始化要注意，因为这里的节点是从1开始编号的，所以VIS数组中不会出现0，可以用0来初始化，若节点是从0开始编号，则VIS中会有0，不能用0初始化
58         IN[root]=0;

```

猿衣酷

专属于程序员的卫衣

关闭

```
59     for(int i=1;i<=n;i++){
60         ans+=IN[i];
61         int v=i;
62         while(!ID[v]&&v!=root&&VIS[v]!=i){
63             VIS[v]=i;
64             v=PRE[v];
65         }
66         if(v!=root&&!ID[v]){
67             ID[v]=++num;
68             for(int j=PRE[v];j!=v;j=PRE[j]){
69                 ID[j]=num;
70             }
71         }
72     }
73     if(!num) break;//没有环，树成立
74     for(int i=1;i<=n;i++){
75         if(!ID[i]){
76             ID[i]=++num;
77         }
78     }
79     for(int i=0;i<m;i++){
80         int x=edge[i].u,y=edge[i].v;
81         edge[i].u=ID[x];
82         edge[i].v=ID[y];
83         if(ID[x]!=ID[y]){
84             edge[i].w-=IN[y];
85         }
86     }
87     n=num;
88     root=ID[root];//维护新的点数目和根
89 }
90 return ans;
91 }
92 int main(){
93     int u,v;
94     while(~scanf("%d %d", &n, &m)){
95         cnt=0;
96         for(int i=1;i<=n;i++){
97             scanf("%d%d", &X[i], &Y[i]);
98         }
99         for(int i=0;i<m;i++){
100             scanf("%d%d", &u, &v);
101             if(u!=v) add(u,v);
102         }
103         double ans=zhuiliu(1);
104         if (ans == -1) printf("poor snoopy\n");
105         else printf("%.2f\n", ans);
106     }
107     return 0;
108 }
109 }
```

参考博客：

算法学习——最小树形图

有 0 个人打赏

文章最后发布于: 2019-10-17 10:06:06

©2019 CSDN 皮肤主题: 大白 设计师: CSDN官方博客

猿衣酷

专属于程序员的卫衣

关闭