

原创

Marriage Match IV (双向SPFA+最大流) 求对短路个数

2019-07-26 11:45:00 _Y-_-Y-_ 阅读数 30 更多

编辑

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/97376637

题目链接：<https://cn.vjudge.net/problem/HDU-3416>

```
1  /*
2  Marriage Match IV
3  HDU - 3416
4  https://cn.vjudge.net/problem/HDU-3416
5  题意： 找出所有最短路的个数
6  输入：
7      T组样例
8      点 边
9      u->v w
10     B E
11 样例输入：
12     2
13     7 8
14     1 2 1
15     1 3 1
16     2 4 1
17     3 4 1
18     4 5 1
19     4 6 1
20     5 7 1
21     6 7 1
22     1 7
23
24     2 2
25     1 2 1
26     1 2 2
27     1 2
28 样例输出：
29     2
30     1
31 解法：
32 双向SPFA枚举每条边找出最短路的连通图
33 最大流求最短路的个数
34 */
35 #include<cstdio>
36 #include<cstring>
37 #include<algorithm>
38 #include<queue>
39 #include<iostream>
40 using namespace std;
41 #define INF 0x3f3f3f
42 #define maxn 100010
43 struct node{
44     int u, v, w;
45     int next;
46 }edge[maxn], uedge[maxn], wedge[maxn*2];
47 int head[maxn], uhead[maxn], whead[maxn];
48 int cnt, ucnt, wcnt;
49 int dis[maxn], udis[maxn];
50 int d[maxn];
51 int t,n,B,E;
52 void init(){
53     memset(head, -1, sizeof head);
54     memset(uhead, -1, sizeof uhead);
55     memset(whead, -1, sizeof whead);
56     memset(dis, INF, sizeof dis);
57     memset(udis, INF, sizeof udis);
```

```

58     cnt = ucnt = wcnt = 0;
59 }
60 void add(int u, int v, int w, node edge[], int &cnt, int *head){
61     edge[cnt].w=w;
62     edge[cnt].u=u;
63     edge[cnt].v=v;
64     edge[cnt].next=head[u];
65     head[u]=cnt++;
66 }
67 void add_dinic(int u,int v,int w){
68     wedge[wcnt].w=w;
69     wedge[wcnt].u=u;
70     wedge[wcnt].v=v;
71     wedge[wcnt].next=whead[u];
72     whead[u]=wcnt++;
73
74     wedge[wcnt].w=0;
75     wedge[wcnt].u=v;
76     wedge[wcnt].v=u;
77     wedge[wcnt].next=whead[v];
78     whead[v]=wcnt++;
79 }
80 void spfa(int st, int *dis, int *head, node edge[]){
81     queue<int> q;
82     bool color[maxn];
83     memset(color,true,sizeof color);
84     q.push(st);
85     dis[st]=0;
86     color[st]=false;
87     while(!q.empty()){
88         int u=q.front();
89         q.pop();
90         color[u]=true;
91         for(int i=head[u];~i;i=edge[i].next){
92             if(dis[u]+edge[i].w<dis[edge[i].v]){
93                 dis[edge[i].v]=dis[u]+edge[i].w;
94                 if(color[edge[i].v]){
95                     q.push(edge[i].v);
96                     color[edge[i].v]=false;
97                 }
98             }
99         }
100     }
101 }
102 bool bfs(int B,int E){
103     memset(d,-1,sizeof d);
104     queue<int> q;
105     d[B]=0;
106     q.push(B);
107     while(!q.empty()){
108         int st=q.front();
109         q.pop();
110         for(int i=whead[st];~i;i=wedge[i].next){
111             if(d[wedge[i].v]==-1&&wedge[i].w>0){
112                 d[wedge[i].v]=d[st]+1;
113                 q.push(wedge[i].v);
114             }
115         }
116     }
117     return d[E]!=-1;
118 }
119 int dfs(int a,int b){
120     int r=0;
121     if(a==E) return b;
122     for(int i=whead[a];~i;i=wedge[i].next){
123         if(wedge[i].w>0&&d[wedge[i].v]==d[a]+1){
124             int x=min(wedge[i].w,b-r);
125             x=dfs(wedge[i].v,x);
126             r+=x;
127             wedge[i].w-=x;
128             wedge[i^1].w+=x;

```

```
129     }
130 }
131 if(!r) d[a]=-2;
132 return r;
133 }
134 int dinic(int B,int E){
135     int ans=0;
136     int t;
137     while(bfs(B,E)){
138         while(t=dfs(B,INF)) ans+=t;
139     }
140     return ans;
141 }
142 int main(){
143     int T;
144     scanf("%d", &T);
145     while(T--){
146         scanf("%d %d", &n, &t);
147         init();
148         for(int i=0;i<t;i++){
149             int u,v,w;
150             scanf("%d %d %d", &u, &v, &w);
151             add(u, v, w, edge, cnt, head);
152             add(v, u, w, uedge, ucnt, uhead);
153         }
154         scanf("%d %d", &B, &E);
155         spfa(B, dis, head, edge);
156         if(dis[E]==INF){
157             printf("0\n");
158             continue;
159         }
160         long long tmp = dis[E];
161         spfa(E, udis, uhead, uedge);
162         for(int i=0;i<t;i++){
163             int u=edge[i].u,v=edge[i].v;
164             if(edge[i].w&&dis[u]+udis[v]+edge[i].w==tmp){
165                 add_dinic(u,v,1);
166             }
167         }
168         printf("%d\n", dinic(B,E));
169     }
170     return 0;
171 }
172 }
```

有 0 个人打赏

文章最后发布于: 2019-07-26 11:45:00

©2019 CSDN 皮肤主题: 大白 设计师: CSDN官方博客