

转载 Path（双向SPFA+最小割，最短路会爆 int）

2019-07-25 14:33:24 _Y-_Y_ 阅读数 16 文章标签： 双向SPFA 最小割 更多

编辑

参考：<https://blog.csdn.net/chenshibo17/article/details/97156743>

Path

<https://cn.vjudge.net/problem/HDU-6582>

思路

1. 先算最短路，要是没有最短路就直接输出0（用SPFA找最短路）
2. 要是有的话新建一个图存所有的最短路（分别存从1到所有的和n到所有点的最短路，然后枚举每条边判断这条边是不是在最短路上，注意会爆int）
3. 在新建的图上跑一遍最小割（最大流等于最小割）

AC code

```
1  /*
2  Path
3  HDU - 6582
4  https://cn.vjudge.net/problem/HDU-6582
5  题意：割断所有的最短路所需要的代价
6  输入：
7      T组样例
8      点 边
9      u->v w
10  样例输入：
11      1
12      3 4
13      1 2 1
14      2 3 1
15      1 3 2
16      1 3 3
17  样例输出：
18      3
19  解法：
20  双向SPFA 枚举每条边找出最短路的连通图
21  最小割来计算代价
22  */
23  #include<cstdio>
24  #include<cstring>
25  #include<algorithm>
26  #include<queue>
27  #include<iostream>
28  using namespace std;
29  #define INF 0x3f3f3f
30  #define maxn 100100
31  #define ll long long
32  struct node{
33      int u, v, w;
34      int next;
35  }edge[maxn], uedge[maxn], wedge[maxn];
36  int head[maxn], uhead[maxn], whead[maxn];
37  int cnt, ucnt, wcnt;
38  ll dis[maxn], udis[maxn];
39  int d[maxn];
40  int t,n;
41  void init(){
42      memset(head, -1, sizeof head);
43      memset(uhead, -1, sizeof uhead);
44      memset(whead, -1, sizeof whead);
45      memset(dis, INF, sizeof dis);
46      memset(udis, INF, sizeof udis);
47      cnt = ucnt = wcnt = 0;
48  }
49  void add(int u, int v, int w, node edge[], int &cnt, int *head){
```

```

50     edge[cnt].w=w;
51     edge[cnt].u=u;
52     edge[cnt].v=v;
53     edge[cnt].next=head[u];
54     head[u]=cnt++;
55 }
56 void add_dinic(int u,int v,int w){
57     wedge[wcnt].w=w;
58     wedge[wcnt].u=u;
59     wedge[wcnt].v=v;
60     wedge[wcnt].next=whead[u];
61     whead[u]=wcnt++;
62
63     wedge[wcnt].w=0;
64     wedge[wcnt].u=v;
65     wedge[wcnt].v=u;
66     wedge[wcnt].next=whead[v];
67     whead[v]=wcnt++;
68 }
69 void spfa(int st, ll *dis, int *head, node edge[]){
70     dis[st]=0;
71     queue<int> q;
72     q.push(st);
73     while(!q.empty()){
74         int u=q.front();
75         q.pop();
76         for(int i = head[u]; ~i; i = edge[i].next ){
77             int v=edge[i].v;
78             if(dis[v] > dis[u] + (ll)edge[i].w){
79                 dis[v] = dis[u] + (ll)edge[i].w;
80                 q.push(v);
81             }
82         }
83     }
84     return ;
85 }
86 bool bfs(){
87     memset(d,-1,sizeof d);
88     queue<int> q;
89     d[1]=0;
90     q.push(1);
91     while(!q.empty()){
92         int st=q.front();
93         q.pop();
94         for(int i=whead[st];~i;i=wedge[i].next){
95             if(d[wedge[i].v]==-1&&wedge[i].w>0){
96                 d[wedge[i].v]=d[st]+1;
97                 q.push(wedge[i].v);
98             }
99         }
100     }
101     return d[n]!=-1;
102 }
103 int dfs(int a,int b){
104     int r=0;
105     if(a==n) return b;
106     for(int i=whead[a];~i;i=wedge[i].next){
107         if(wedge[i].w>0&&d[wedge[i].v]==d[a]+1){
108             int x=min(wedge[i].w,b-r);
109             x=dfs(wedge[i].v,x);
110             r+=x;
111             wedge[i].w-=x;
112             wedge[i^1].w+=x;
113         }
114     }
115     if(!r) d[a]=-2;
116     return r;
117 }
118 ll dinic(){
119     ll ans=0;
120     int t;

```

```
121 while(bfs()){
122     while(t=dfs(1,INF)) ans+=(ll)t;
123 }
124 return ans;
125 }
126 int main(){
127     int T;
128     scanf("%d", &T);
129     while(T--){
130         scanf("%d %d", &n, &t);
131         init();
132         for(int i=0;i<t;i++){
133             int u,v,w;
134             scanf("%d %d %d", &u, &v, &w);
135             add(u, v, w, edge, cnt, head);
136             add(v, u, w, uedge, ucnt, uhead);
137         }
138         spfa(1, dis, head, edge);
139         if(dis[n]==INF){
140             printf("0\n");
141             continue;
142         }
143         long long tmp = dis[n];
144         spfa(n, udis, uhead, uedge);
145         for(int i=0;i<t;i++){
146             int u=edge[i].u,v=edge[i].v;
147             if(dis[u]+udis[v]+edge[i].w==tmp){
148                 add_dinic(u,v,edge[i].w);
149             }
150         }
151         printf("%lld\n", dinic());
152     }
153     return 0;
154 }
155
156
```

有 0 个人打赏

文章最后发布于: 2019-07-25 14:33:24

©2019 CSDN 皮肤主题: 大白 设计师: CSDN官方博客