

Algorithm Codelet

TieWay59

October 12, 2019

Contents

1	测试一级标题	2	3
1.1	gym-101808-K LCA-ST.cpp	3	3

1 测试一级标题 2

1.1 gym-101808-K LCA-ST.cpp

```
1  /*
2   * src: https://codeforces.com/gym/101808/problem/K
3   * use: LCA + ST
4   *
5   */
6
7  #define _debug(x) cerr<<#x<<" = "<<(x)<<endl;fflush(stdout)
8
9  #include <bits/stdc++.h>
10
11 using namespace std;
12 typedef long long ll;
13 const int maxn = 100000 + 77;
14 const int inf = 1e9;
15 const int mo = 7901;
16 int rmq[maxn << 1];
17
18 struct ST {
19     int mm[maxn << 1];
20     int dp[maxn << 1][20];
21
22     void init(int n) {
23         mm[0] = -1;
24         for (int i = 1; i <= n; ++i) {
25             mm[i] = mm[i >> 1] + 1;
26             dp[i][0] = i;
27         }
28         for (int j = 1; j <= mm[n]; ++j) {
29             for (int i = 1; i + (1 << j) - 1 <= n; ++i) {
30                 dp[i][j] =
31                     rmq[dp[i][j - 1]] <
32                     rmq[dp[i + (1 << (j - 1))][j - 1]] ?
33                     dp[i][j - 1] :
34                     dp[i + (1 << (j - 1))][j - 1];
35             }
36         }
37     }
38
39     int query(int a, int b) {
40         if (a > b) swap(a, b);
41         int k = mm[b - a + 1];
42         return rmq[dp[a][k]] <=
43             rmq[dp[b - (1 << k) + 1][k]] ?
44             dp[a][k] :
45             dp[b - (1 << k) + 1][k];
46     }
47 };
48
49 struct Edge {
50     int to, nx;
51     ll w;
```

```

52 };
53 Edge edge[maxn << 1];
54 int totEd, head[maxn];
55
56 int F[maxn << 1];
57 int P[maxn];
58 int cntStp;
59
60 ST st;
61
62 void init() {
63     totEd = 0;
64     memset(head, -1, sizeof(head));
65 }
66
67 void addedge(int u, int v, int w) {
68     edge[totEd] = {v, head[u], w};
69     head[u] = totEd++;
70     edge[totEd] = {u, head[v], w};
71     head[v] = totEd++;
72 }
73
74 int CrcA, CrcB, CrcW;
75 bool vis[maxn];
76 ll dis[maxn];
77
78 void dfs(int u, int fa, int dep) {
79     F[++cntStp] = u;
80     rmq[cntStp] = dep;
81     P[u] = cntStp;
82     vis[u] = true;
83
84     for (int v = 0, i = head[u]; ~i; i = edge[i].nx) {
85         v = edge[i].to;
86
87         if (vis[v]) {
88             if (v != fa) {
89                 CrcA = u;
90                 CrcB = v;
91                 CrcW = edge[i].w;
92             }
93             continue;
94         }
95         // _debug(v);
96         dis[v] = dis[u] + edge[i].w;
97         dfs(v, u, dep + 1);
98         F[++cntStp] = u;
99         rmq[cntStp] = dep;
100     }
101 }
102
103 void LCA_init(int root, int node_num) {
104     cntStp = 0;
105     memset(vis, 0, sizeof vis);
106     memset(dis, 0, sizeof dis);
107     dfs(root, -1, 0);

```

```

108     st.init(2 * node_num - 1);
109 }
110
111 inline int LCA_query(int u, int v) {
112     return F[st.query(P[u], P[v])];
113 }
114
115 inline ll qryDis(int u, int v) {
116     return dis[u] + dis[v] - 2 * dis[LCA_query(u, v)];
117 }
118
119 int main() {
120     // ios::sync_with_stdio(false);
121     // cin.tie(nullptr);
122     // cout.tie(nullptr);
123
124     int Kase = 0, N = 0, Q = 0, root = 1, u = 0, v = 0;
125     // cin >> Kase;
126     scanf("%d", &Kase);
127     while (Kase--) {
128         scanf("%d %d", &N, &Q);
129         init();
130         for (int i = 1, w; i <= N; ++i) {
131             scanf("%d %d %d", &u, &v, &w);
132             addedge(u, v, w);
133         }
134         LCA_init(root, N);
135         while (Q--) {
136             scanf("%d%d", &u, &v);
137             ll res = qryDis(u, v);
138             res = min(res, qryDis(u, CrcA) + qryDis(v, CrcB) + CrcW);
139             res = min(res, qryDis(v, CrcA) + qryDis(u, CrcB) + CrcW);
140             printf("%lld\n", res);
141         }
142     }
143     return 0;
144 }
145
146
147
148 /*
149 3
150 3
151 1 2
152 2 222
153 3 22
154
155 */

```
