

Programming 3-3: 并查集

MA Jun

2018.09.10

1 Problem 3-3-A:Find them, Catch them

1.1 代码样例

```
1 #include<bits/stdc++.h>
2 #define maxn 100005
3 using namespace std;
4 int p[maxn], r[maxn], n, m, c1, c2;
5 char op[10];
6 int find(int x) {
7     if(x!=p[x]) {
8         int tmp = p[x];
9         p[x] = find(p[x]);
10        r[x] = (r[x]+r[tmp])%2;
11    }
12    return p[x];
13 }
14 void join(int x, int y) {
15     int px = find(x);
16     int py = find(y);
17
18     p[py] = px;
19     r[py] = (r[x]+r[y]+1)%2;
20 }
21 void work() {
22     scanf("%d %d", &n, &m);
23     getchar();
24     for(int i = 0; i < n; ++i) {
25         p[i] = i;
26         r[i] = 0;
27     }
28     for(int i = 0; i < m; ++i) {
29         scanf("%s %d %d", op, &c1, &c2);
30         getchar();
31         c1--;
32         c2--;
33         if(op[0]=='D') {
34             join(c1,c2);
35         } else {
36             if(find(c1)==find(c2)) {
37                 r[c1]==r[c2] ? printf("In the same gang.\n") : printf("In different gangs.\n");
38             } else {
```

```
39         printf("Not sure yet.\n");
40     }
41 }
42 }
43 }
44
45 int main() {
46     int T;
47     scanf("%d",&T);
48     while(T-->0)
49         work();
50     return 0;
51 }
52
53 /*****
54     Problem: 1542
55     User: 171860004
56     Language: C++
57     Result: 正确
58     Time:0 ms
59     Memory:2328 kb
60 *****/
```

Listing 1: A by 171860004

2 Problem 3-3-B: Happy PA

2.1 代码样例

```

1  #include<cstdio>
2  #include<algorithm>
3  using namespace std;
4  int n,k,m;
5  int fa[100003];
6  struct node {
7      int start;
8      int end;
9      int width;
10 }edges[200007];
11
12 void make_Set(int i) {
13     fa[i]=i;
14 }
15
16 int Find_Set(int i) {
17     if(fa[i]==i)
18         return i;
19     return fa[i]=Find_Set(fa[i]);
20 }
21 }
22
23 void Union_Set(int a, int b) {
24     a=Find_Set(a);
25     b=Find_Set(b);
26     if(a==b)
27         return;
28     fa[b]=a;
29 }
30
31 bool Same_Set(int a, int b) {
32     return Find_Set(a)==Find_Set(b);
33 }
34
35 bool cmp(node a, node b) {
36     return a.width<b.width;
37 }
38
39 int main(void) {
40     int x,y;
41     int res=0;
42     scanf("%d%d%d",&n,&k,&m);
43     int cnt=n;
44     for(int i=0;i<n;i++)
45         make_Set(i);
46     for(int i=0;i<k;i++) {
47         scanf("%d%d",&x,&y);
48         if(!Same_Set(x,y)) {
49             cnt--;
50             Union_Set(x,y);
51         }
52     }

```

```
53 printf("%d ",cnt);
54 for(int i=0;i<m;i++){
55     scanf("%d%d%d",&edges[i].start,&edges[i].end,&edges[i].width);
56 sort(edges,edges+m,cmp);
57 for(int i=0;i<m;i++){
58     if(cnt==1) break;
59     if(!Same_Set(edges[i].start,edges[i].end)) {
60         Union_Set(edges[i].start,edges[i].end);
61         res+=edges[i].width;
62         cnt--;
63     }
64 }
65 printf("%d\n",res);
66 }
67
68
69
70 /*****
71 Problem: 1543
72 User: 171860509
73 Language: C++
74 Result: 正确
75 Time:152 ms
76 Memory:3696 kb
77 *****/
```

Listing 2: B by 171860509

3 Problem 3-3-C: 旌旗十万诛神

3.1 代码样例

```

1  #include<stdio.h>
2  int pre[110000];
3
4  int find(int x){
5      if (pre[x]==-1 || pre[x]==x){
6          return pre[x];
7      }
8      else {
9          pre[x]=find(pre[x]);
10         return pre[x];
11     }
12 }
13
14 int main(){
15     int n,m,x;
16     int i;
17     char str[20];
18     scanf("%d%d",&n,&m);
19     n--;
20     for (i=0;i<=n;i++){
21         pre[i]=i;
22     }
23     for (i=1;i<=m;i++){
24         scanf("%s%d",&str,&x);
25
26         if (str[0]=='K'){
27             if (x<n){
28                 pre[x]=find(x+1);
29             }
30             else{
31                 pre[x]=-1;
32             }
33         }
34         else{
35             x=find(x);
36             printf("%d\n",x);
37         }
38     }
39 }
40 /*****
41     Problem: 1544
42     User: 171860609
43     Language: C
44     Result: 正确
45     Time:72 ms
46     Memory:1388 kb
47 *****/

```

Listing 3: C by 171860609

4 Problem 3-3-D: 幻想大学开学季

4.1 代码样例

```

1  #include <bits/stdc++.h>
2  using namespace std;
3
4  int k = 0, cnt = 0, op = 0;
5  map<string, int> mp;
6  vector<int> pa;
7  vector<int> gp;
8  vector<bool> ne;
9  /* -2-unknown,
10 * -1-super,
11 * 0-same as parent,
12 * 1-diss parent,
13 * 2-dissed by parent */
14
15 int getID(string &s) {
16     if (mp.find(s) == mp.end()) {
17         mp[s] = cnt;
18         pa.push_back(cnt);
19         gp.push_back(0); //unknown at first
20         ne.push_back(true);
21         return cnt++;
22     } else {
23         return mp.find(s)->second;
24     }
25 }
26
27 int getNewID(string s) {
28     mp[s] = cnt;
29     pa.push_back(cnt);
30     gp.push_back(0); //unknown at first
31     ne.push_back(true);
32     return cnt++;
33 }
34
35 int findSet(int n) {
36     if (pa[n] == n) {
37         return n;
38     } else {
39         int npa = findSet(pa[n]);
40         gp[n] = (gp[n] + gp[pa[n]]) % 3;
41         return pa[n] = npa;
42     }
43 }
44
45 void unionSet(int op, int a, int b) {
46     ne[a] = ne[b] = false;
47     int fa = findSet(a), fb = findSet(b);
48     pa[fa] = fb;
49     if (op == 0) {
50         gp[fa] = (gp[b] - gp[a] + 3) % 3;
51     } else {

```

```
52     gp[fa] = (gp[b] - gp[a] + 4) % 3;
53 }
54 findSet(a); // fuck!
55 findSet(b);
56 }
57
58 bool judge(int op, int a, int b, string &sa, string &sb) {
59     switch (op) {
60     case 0: {
61         if (a == b) {
62             return true;
63         } else {
64             int fa = findSet(a), fb = findSet(b);
65             if (fa != fb) {
66                 unionSet(op, a, b);
67             }
68             return gp[a] == gp[b];
69         }
70     }
71     case 1: {
72         if (a == b) {
73             return false;
74         } else {
75             int fa = findSet(a), fb = findSet(b);
76             if (fa != fb) {
77                 unionSet(op, a, b);
78             }
79             return gp[a] == (gp[b] + 1) % 3;
80         }
81     }
82     case 2: {
83         findSet(a);
84         if (gp[a] == -1 || ne[a]) {
85             return false;
86         } else {
87             getNewID(sa);
88             return true;
89         }
90     }
91     case 3: {
92         findSet(a);
93         return gp[a] < 0 || ne[a];
94     }
95 }
96 }
97
98 void init() {
99     cnt = 0;
100     mp.clear();
101     pa.clear();
102     gp.clear();
103     ne.clear();
104 }
105
106 int main() {
107     ios::sync_with_stdio(false);
108     int T = 0;
```

```
109 cin >> T;
110 while (T--) {
111     init();
112     cin >> k;
113     int ida = 0, idb = 0, ans = 0;
114     string sa, sb;
115     for (int i = 0; i < k; ++i) {
116         cin >> op >> sa;
117         ida = getID(sa);
118         if (op < 2) {
119             cin >> sb;
120             idb = getID(sb);
121         } else {
122             idb = 0;
123         }
124         if (!judge(op, ida, idb, sa, sb)) {
125             ans++;
126             //cout << i + 1 << "-th is bad." << endl;
127         }
128     }
129     cout << ans << endl;
130 }
131 return 0;
132 }
133 /*****
134     Problem: 1545
135     User: 171860508
136     Language: C++
137     Result: 正确
138     Time:216 ms
139     Memory:1828 kb
140 *****/
```

Listing 4: D by 171860508