

Programming 3-4: 最小生成树

MA Jun

2018.10.31

1 Problem 3-4-A: Agri-Net

1.1 代码样例

```
1 #include <cstdio>
2 #include <algorithm>
3 #include <cmath>
4 using namespace std;
5
6 const int MAXN = 105;
7
8 int dad[MAXN];
9 void init(int n){
10     for (int i=1;i<=n;i++) dad[i] = i;
11 }
12 int find(int x){
13     if (dad[x]==x)
14         return x;
15     return find(dad[x]);
16 }
17 int uni(int x,int y,int enact = 1){
18     x = find(x);
19     y = find(y);
20     if (x==y)
21         return 0;
22     if (enact) dad[y] = x;
23     return 1;
24 }
25
26 struct edge{
27     int u,v,w;
28 };
29 bool operator<(const edge& x,const edge& y){
30     return x.w<y.w;
31 }
32 edge E[MAXN*MAXN];
33
34 int n,m,T;
35
36 int main(){
37
38     T = 1; while (T--){
```

```
39     scanf("%d",&n);
40     m = 0;
41     init(n);
42     for (int i=1;i<=n;i++)
43         for (int j=1;j<=n;j++){
44             int x;
45             scanf("%d",&x);
46             if (j<=i) continue;
47             m++;
48             E[m].u = i;
49             E[m].v = j;
50             E[m].w = x;
51         }
52     sort(E+1,E+1+m);
53
54     long long ans = 0;
55     for (int i=1;i<=m;i++) ans += uni(E[i].u,E[i].v)*E[i].w;
56     printf("%lld\n",ans);
57 }
58 return 0;
59 }
60 /*****
61 Problem: 1547
62 User: 171860696
63 Language: C++
64 Result: 正确
65 Time:0 ms
66 Memory:1088 kb
67 *****/
```

Listing 1: A by 171860696

2 Problem 3-4-B: NJU 送“温暖”

2.1 代码样例

```

1  #include<stdio.h>
2  #include<cstring>
3  #include<algorithm>
4  using namespace std;
5  int t,n,m;
6  int p[25];
7  struct Node{
8      int x,y,w;
9      bool vis;
10 }a[205];
11 int sum;
12
13 bool cmp(Node a, Node b){
14     return a.w<b.w;
15 }
16
17 int find_set(int x){
18     if(p[x]!=x)
19         p[x]=find_set(p[x]);
20     return p[x];
21 }
22
23 void init(){
24     scanf("%d%d",&n,&m);
25     if(n==1){
26         printf("0\n");
27         return;
28     }
29     for(int i=1; i<=m; i++){
30         scanf("%d%d%d",&a[i].x,&a[i].y,&a[i].w);
31     }
32     sort(a+1,a+m+1,cmp);
33 }
34
35 int kruskal(int k){
36     int tmp_sum = 0;
37     for(int i=1; i<=n; i++){
38         p[i]=i;
39     }
40     for(int i=1; i<=m; i++){
41         if(i==k){
42             continue;
43         }
44         int root1 = find_set(a[i].x);
45         int root2 = find_set(a[i].y);
46         if(root1!=root2){
47             p[root1]=root2;
48             tmp_sum += a[i].w;
49             if(k==0){
50                 a[i].vis = 1;
51             }

```

```

52     }
53 }
54 return tmp_sum;
55 }
56
57 void solve(){
58     init();
59     if(n==1){
60         return;
61     }
62     sum = kruskal(0);
63     int cnt=0;
64     for(int i=1; i<=m; i++){
65         if(a[i].vis==1&&ca[i].w!=0){
66             if(sum==kruskal(i))
67                 cnt++;
68         }
69         if(cnt>=1){
70             printf("Peter can find other ways\n");
71             return;
72         }
73     }
74     printf("%d\n",sum);
75 }
76
77 int main(){
78     scanf("%d",&t);
79     for(int i=1; i<=t; i++){
80         solve();
81     }
82 }
83 /*****
84 Problem: 1548
85 User: 171240511
86 Language: C++
87 Result: 正确
88 Time:0 ms
89 Memory:968 kb
90 *****/

```

Listing 2: B by 171240511

3 Problem 3-4-C: 脸盲得治

3.1 代码样例

```

1  #include<cstdio>
2  #include<algorithm>
3  using namespace std;
4
5  #define MAXSIZE 105
6
7  int n, k;
8  int a[MAXSIZE];
9  int b[MAXSIZE];
10 int uset[MAXSIZE];
11 int w[MAXSIZE*MAXSIZE];
12
13 void make_set(int size){
14     for (int i = 0; i<size; i++){
15         uset[i] = i;
16     }
17 }
18
19 int find_set(int x){
20     if (x != uset[x]){
21         uset[x] = find_set(uset[x]);
22     }
23     return uset[x];
24 }
25
26 void union_set(int x, int y){
27     int fx = find_set(x);
28     int fy = find_set(y);
29
30     if (fx != fy)
31         uset[fx] = fy;
32 }
33
34 int main(){
35     scanf("%d%d", &n, &k);
36     for (int i = 0; i < n; i++){
37         scanf("%d%d", &a[i], &b[i]);
38     }
39     for (int i = 0; i < n; i++){
40         for (int j = i; j < n; j++){
41             int da = a[i] - a[j];
42             int db = b[i] - b[j];
43             if (da < 0)
44                 da = -da;
45             if (db < 0)
46                 db = -db;
47             w[i*n + j] = w[j*n + i] = da + db;
48         }
49     }
50     int t = n - k;
51     int kth_max = 0;

```

```
52 make_set(n);
53 while (t--){
54     int e = min_element(w, w + n * n) - w;
55
56     if (find_set(e / n) == find_set(e % n)) {
57         w[e] = 0x7ffffff;
58         t++;
59     }
60     else{
61         union_set(e / n, e % n);
62         kth_max = max(kth_max, w[e]);
63         w[e] = 0x7ffffff;
64     }
65 }
66
67 printf("%d", kth_max);
68
69 return 0;
70 }
71
72
73
74 /*****
75 Problem: 1549
76 User: 171860695
77 Language: C++
78 Result: 正确
79 Time:4 ms
80 Memory:1004 kb
81 *****/
```

Listing 3: C by 171860695

4 Problem 3-4-D: 抽奖

4.1 代码样例

```

1  #include<stdio.h>
2  #include<cstring>
3  #include<algorithm>
4  #include<math.h>
5  #define zero(x) ((x>0?x:-x)<1e-10)
6  using namespace std;
7  int t,n,m;
8  double matrix[15][15];
9  bool vis[15][15];
10
11 void init(){
12     scanf("%d%d",&n,&m);
13     int x, y;
14     for(int i=1; i<=n; i++){
15         for(int j=1; j<=n; j++){
16             vis[i][j]=0;
17             matrix[i][j]=0;
18         }
19     }
20     for(int i=1; i<=m; i++){
21         scanf("%d%d",&x,&y);
22         if(vis[x][y]==0&&vis[y][x]==0&&x!=y){
23             matrix[x][x]+=1;
24             matrix[y][y]+=1;
25             matrix[x][y] = matrix[y][x] = -1;
26             vis[x][y]=1;
27         }
28     }
29 }
30
31 void gauss(){
32     for(int i=1; i<n; i++){
33         if(zero(matrix[i][i])){
34             int j;
35             for(j=i+1; j<n; j++){
36                 if(!zero(matrix[j][i])){
37                     break;
38                 }
39             }
40             if(j<n){
41                 for(int k=1; k<n; k++){
42                     double tmp = matrix[i][k];
43                     matrix[i][k] = matrix[j][k];
44                     matrix[j][k] = tmp;
45                 }
46             }else{
47                 return;
48             }
49         }
50         for(int j=i+1; j<n; j++){
51             double tmp = matrix[j][i]/matrix[i][i];
52             for(int k=i; k<n; k++){

```

```
53         matrix[j][k]=matrix[j][k]-(matrix[i][k]*tmp);
54     }
55 }
56 }
57 }
58
59 void solve(){
60     init();
61     gauss();
62     double ans=1;
63     for(int i=1; i<n; i++){
64         ans*=matrix[i][i];
65     }
66     if(ans<0){
67         ans = -ans;
68     }
69     printf("%.01f\n",ans);//一定要先转 int, 不然 printf 会输出 0;
70 }
71
72 int main(){
73     scanf("%d",&t);
74     for(int i=1; i<=t; i++){
75         solve();
76     }
77 }
78 /*****
79 Problem: 1550
80 User: 171240511
81 Language: C++
82 Result: 正确
83 Time:0 ms
84 Memory:960 kb
85 *****/
```

Listing 4: D by 171240511