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

```
#include <cstdio>
#include <algorithm>
#include <cmath>
using namespace std;

const int MAXN = 105;

int dad[MAXN];
void init(int n){
    for (int i=1;i<=n;i++) dad[i] = i;
}
int find(int x){
    if (dad[x]==x)
        return x;
    return find(dad[x]);
}
int uni(int x,int y,int enact = 1){
    x = find(x);
    y = find(y);
    if (x==y)
        return 0;
    if (enact) dad[y] = x;
    return 1;
}

struct edge{
    int u,v,w;
};
bool operator<(const edge& x,const edge& y){
    return x.w<y.w;
}
edge E[MAXN*MAXN];

int n,m,T;

int main(){

    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

```
#include<stdio.h>
#include<cstring>
#include<algorithm>
using namespace std;
int t,n,m;
int p[25];
struct Node{
    int x,y,w;
    bool vis;
}a[205];
int sum;

bool cmp(Node a, Node b){
    return a.w<b.w;
}

int find_set(int x){
    if(p[x]!=x)
        p[x]=find_set(p[x]);
    return p[x];
}

void init(){
    scanf("%d%d",&n,&m);
    if(n==1){
        printf("0\n");
        return;
    }
    for(int i=1; i<=m; i++){
        scanf("%d%d%d",&a[i].x,&a[i].y,&a[i].w);
    }
    sort(a+1,a+m+1,cmp);
}

int kruskal(int k){
    int tmp_sum = 0;
    for(int i=1; i<=n; i++){
        p[i]=i;
    }
    for(int i=1; i<=m; i++){
        if(i==k){
            continue;
        }
        int root1 = find_set(a[i].x);
        int root2 = find_set(a[i].y);
        if(root1!=root2){
            p[root1]=root2;
            tmp_sum += a[i].w;
            if(k==0){
                a[i].vis = 1;
            }
```

```
        }
    }
    return tmp_sum;
}

void solve(){
    init();
    if(n==1){
        return;
    }
    sum = kruskal(0);
    int cnt=0;
    for(int i=1; i<=m; i++){
        if(a[i].vis==1&&a[i].w!=0){
            if(sum==kruskal(i))
                cnt++;
        }
        if(cnt>=1){
            printf("Peter can find other ways\n");
            return;
        }
    }
    printf("%d\n",sum);
}

int main(){
    scanf("%d",&t);
    for(int i=1; i<=t; i++){
        solve();
    }
}
/**************************************************************
    Problem: 1548
    User: 171240511
    Language: C++
    Result: 正确
    Time:0 ms
    Memory:968 kb
**************************************************************/
```

Listing 2: B by 171240511

3 Problem 3-4-C: 脸盲得治

3.1 代码样例

```
#include<cstdio>
#include<algorithm>
using namespace std;

#define MAXSIZE 105

int n, k;
int a[MAXSIZE];
int b[MAXSIZE];
int uset[MAXSIZE];
int w[MAXSIZE*MAXSIZE];

void make_set(int size){
    for (int i = 0; i<size; i++){
        uset[i] = i;
    }
}

int find_set(int x){
    if (x != uset[x]){
        uset[x] = find_set(uset[x]);
    }
    return uset[x];
}

void union_set(int x, int y){
    int fx = find_set(x);
    int fy = find_set(y);

    if (fx != fy)
        uset[fx] = fy;
}

int main(){
    scanf("%d%d", &n, &k);
    for (int i = 0; i < n; i++){
        scanf("%d%d", &a[i], &b[i]);
    }
    for (int i = 0; i < n; i++){
        for (int j = i; j < n; j++){
            int da = a[i] - a[j];
            int db = b[i] - b[j];
            if (da < 0)
                da = -da;
            if (db < 0)
                db = -db;
            w[i*n + j] = w[j*n + i] = da + db;
        }
    }
    int t = n - k;
    int kth_max = 0;
```

```
    make_set(n);
    while (t--){
        int e = min_element(w, w + n * n) - w;

        if (find_set(e / n) == find_set(e % n)) {
            w[e] = 0x7fffffff;
            t++;
        }
        else{
            union_set(e / n, e % n);
            kth_max = max(kth_max, w[e]);
            w[e] = 0x7fffffff;
        }
    }

    printf("%d", kth_max);

    return 0;
}



/**************************************************************
    Problem: 1549
    User: 171860695
    Language: C++
    Result: 正确
    Time:4 ms
    Memory:1004 kb
****************************************************************/
```

Listing 3: C by 171860695

4 Problem 3-4-D: 抽奖

4.1 代码样例

```
#include<stdio.h>
#include<cstring>
#include<algorithm>
#include<math.h>
#define zero(x) ((x>0?x:-x)<1e-10)
using namespace std;
int t,n,m;
double matrix[15][15];
bool vis[15][15];

void init(){
    scanf("%d%d",&n,&m);
    int x, y;
    for(int i=1; i<=n; i++){
        for(int j=1; j<=n; j++){
            vis[i][j]=0;
            matrix[i][j]=0;
        }
    }
    for(int i=1; i<=m; i++){
        scanf("%d%d",&x,&y);
        if(vis[x][y]==0&&vis[y][x]==0&&x!=y){
            matrix[x][x]+=1;
            matrix[y][y]+=1;
            matrix[x][y] = matrix[y][x] = -1;
            vis[x][y]=1;
        }
    }
}

void gauss(){
    for(int i=1; i<n; i++){
        if(zero(matrix[i][i])){
            int j;
            for(j=i+1; j<n; j++){
                if(!zero(matrix[j][i])){
                    break;
                }
            }
            if(j<n){
                for(int k=1; k<n; k++){
                    double tmp = matrix[i][k];
                    matrix[i][k] = matrix[j][k];
                    matrix[j][k] = tmp;
                }
            }else{
                return;
            }
        }
        for(int j=i+1; j<n; j++){
            double tmp = matrix[j][i]/matrix[i][i];
            for(int k=i; k<n; k++){
```

```
                matrix[j][k]=matrix[j][k]-(matrix[i][k]*tmp);
            }
        }
    }
}

void solve(){
    init();
    gauss();
    double ans=1;
    for(int i=1; i<n; i++){
        ans*=matrix[i][i];
    }
    if(ans<0){
        ans = -ans;
    }
    printf("%.0lf\n",ans);//一定要先转 int，不然 printf 会输出 0；
}

int main(){
    scanf("%d",&t);
    for(int i=1; i<=t; i++){
        solve();
    }
}
/**************************************************************
    Problem: 1550
    User: 171240511
    Language: C++
    Result: 正确
    Time:0 ms
    Memory:960 kb
**************************************************************/
```

Listing 4: D by 171240511