

最小生成树(Prim)

```
#include <bits/stdc++.h>
using namespace std;

int tot,n,m,cnt,sum,u,v,w,head[5005],dis[5005],vis[5005];

struct Edge{
    int v,w,next;
}ed[40005];

void add(int u,int v,int w){
    tot++;
    ed[tot].v=v;
    ed[tot].w=w;
    ed[tot].next=head[u];
    head[u]=tot;
}

typedef pair <int,int> pii;
priority_queue <pii,vector<pii>,greater<pii> >q;

void prim(){
    dis[1]=0;
    q.push(make_pair(0,1));
    while(!q.empty()&&cnt<n){
        int d=q.top().first,u=q.top().second;
        q.pop();
        if(vis[u]) continue;
        cnt++;
        sum+=d;
        vis[u]=1;
        for(int i=head[u];i;i=ed[i].next){
            if(ed[i].w<dis[ed[i].v]){
                dis[ed[i].v]=ed[i].w;
                q.push(make_pair(dis[ed[i].v],ed[i].v));
            }
        }
    }
}

int main(){
    memset(dis,0x7f,sizeof(dis));
    scanf("%d%d",&n,&m);
    for(int i=1;i<=m;i++){
        scanf("%d%d%d",&u,&v,&w);
        add(u,v,w);
        add(v,u,w);
    }
    prim();
    if (cnt==n) printf("%d",sum);
    else printf("orz");
}
```

