**#include <iostream>**

**Dijstra(m\*log2n)**

**#include <stdio.h>**

**#include <string.h>**

**#include <queue>**

**#include <vector>**

**using namespace std;**

**#define MX 1010**

**#define INF 0x3f3f3f3f**

**struct Edge**

**{**

**int to;**

**int w;**

**bool operator <(const Edge & b)const**

**{**

**return w>b.w;**

**}**

**};**

**int n,m;**

**vector<Edge> G[MX];**

**int dis[MX];**

**int vis[MX];**

**void Init()**

**{**

**for (int i=0;i<=n;i++)**

**G[i].clear();**

**}**

**void Dijstra(int s,int t)**

**{**

**memset(vis,0,sizeof(vis));**

**for (int i=0;i<=n;i++) dis[i]=INF;**

**priority\_queue<Edge> Q;**

**dis[s]=0;**

**Q.push((Edge){s,0});**

**while (!Q.empty())**

**{**

**Edge x = Q.top();Q.pop();**

**int u = x.to;**

**if (vis[u]) continue;**

**vis[u]=1;**

**for (int i=0;i<G[u].size();i++)**

**{**

**Edge &e = G[u][i];**

**if(dis[u]+e.w<dis[e.to])**

**{**

**dis[e.to]=dis[u]+e.w;**

**Q.push((Edge){e.to,dis[e.to]});**

**}**

**}**

**}**

**}**

**int main()**

**{**

**while (scanf("%d%d",&n,&m)&&(n+m))**

**{**

**Init();**

**for (int i=0;i<m;i++)**

**{**

**int u,v,w;**

**scanf("%d%d%d",&u,&v,&w);**

**G[u].push\_back((Edge){v,w});**

**G[v].push\_back((Edge){u,w});**

**}**

**int s,t;**

**scanf("%d%d",&s,&t);**

**Dijstra(s,t);**

**printf("%d\n",dis[t]);**

**}**

**return 0;**

**}**