**#include <iostream>**

最大流

**#include <cstdio>**

**#include <cstring>**

**#include <vector>**

**#include <queue>**

**using namespace std;**

**#define MAXN 30**

**#define INF 1e8**

**struct Edge**

**{**

**int from,to,cap,flow;**

**Edge(int fr,int t,int c,int f):from(fr),to(t),cap(c),flow(f){}**

**};**

**int n,real\_m,m; //n个点，m条边**

**vector<Edge> edges; //所有的边**

**vector<int> G[MAXN]; //G[i][j]表示起点 i 的第 j 条边的序号**

**int add[MAXN]; //可增量**

**int pre[MAXN]; //前驱**

**void Init(){**

**edges.clear();**

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

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

**}**

**int AddEdge(int from,int to,int cap){**

**edges.push\_back(Edge(from,to,cap,0));**

**edges.push\_back(Edge(to,from,0,0));**

**m=edges.size();**

**G[from].push\_back(m-2);**

**G[to].push\_back(m-1);**

**return 0;**

**}**

**int MaxFlow(int s,int t)**

**{**

**int flow=0;**

**while(1)**

**{**

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

**add[s]=INF;**

**queue<int> Q;**

**Q.push(s);**

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

**{**

**int u = Q.front(); Q.pop();**

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

**{**

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

**if(!add[e.to]&&e.cap>e.flow)//如果没去过并且可增**

**{**

**add[e.to]=min(add[u],e.cap-e.flow);**

**pre[e.to]=G[u][i];**

**Q.push(e.to);**

**}**

**if(add[t])**

**break;**

**}**

**}**

**if (!add[t]) //不能增了**

**break;**

**for(int u=t;u!=s;u=edges[pre[u]].from)**

**{**

**int num=pre[u];//那条边的编号**

**edges[num].flow +=add[t];**

**edges[num^1].flow-=add[t];**

**}**

**flow+=add[t];**

**}**

**return flow;**

**}**

**int main(){**

**while(1){**

**cout<<"输入顶点个数:"; cin >> n;**

**Init();**

**int from , to ,cap;**

**cout<<"输入边个数:"; cin>>real\_m;**

**cout<<"起点 终点 容量"<<endl;**

**for(int i=0;i<real\_m;i++)**

**{**

**scanf("%d%d%d",&from,&to,&cap);**

**AddEdge(from,to,cap);**

**}**

**int s,t;**

**cout<<"起点:"; cin >> s;**

**cout<<"终点:"; cin >> t;**

**cout<<"最大流量为:"<<MaxFlow(s,t)<<endl;**

**}**

**return 0;**

**}**