

最大流

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

const int oo=0x7fffffff;
int n,m,s,t,u,v,tot=1;
int head[520010],vis[520010],pre[520010],flag[2510][2510];
long long w,ans,dis[520010];
struct node{
    int to,net;
    long long val;
}e[520010];

inline void add(int u,int v,long long w){
    e[++tot].to=v;//正向边
    e[tot].val=w;
    e[tot].net=head[u];
    head[u]=tot;
    e[++tot].to=u;//反向边
    e[tot].val=0;
    e[tot].net=head[v];
    head[v]=tot;
}

inline int bfs(){
    for(int i=1;i<=n;i++){
        vis[i]=0;
    }
    queue<int>q;
    q.push(s);
    vis[s]=1;
    dis[s]=oo;
    while (!q.empty()){
        int x=q.front();
        q.pop();
        for(int i=head[x];i;i=e[i].net){
            if(e[i].val==0) continue;
            int v=e[i].to;
            if(vis[v]) continue;
            dis[v]=min(dis[x],e[i].val);
            pre[v]=i;
            q.push(v);
            vis[v]=1;
            if(v==t) return 1;
        }
    }
    return 0;
}

inline void update(){
    int x=t;
    while (x!=s){
        int v=pre[x];
```

```

        e[v].val-=dis[t];
        e[v^1].val+=dis[t];
        x=e[v^1].to;
    }
    ans+=dis[t];
}

int main(){
    ios::sync_with_stdio(false);
    cin.tie(0);
    cin >> n >> m >> s >> t;
    for(int i=1;i<=m;i++){
        cin >> u >> v >> w;
        if(flag[u][v]==0){
            add(u,v,w);
            flag[u][v]=tot;
        }
        else e[flag[u][v]-1].val+=w;
    }
    while (bfs()!=0){
        update();
    }
    cout << ans;
    return 0;
}

```