

糖果

一. 考察内容:

图论 最长路径 差分约束

二. 题目分析:

[题目大意]

给 n 个人分糖果，每个人至少分一个，要满足一些相对关系的限制条件，求出最少需要多少糖。

[写题思路]

该题需要满足一些元素值之间大小关系，考虑用差分约束实现。

首先，对于 $a=b$ 的关系，我们建一条无向边 $a \rightarrow b$ ， $v=0$ ，表示这两个节点权值必须相等，对于 $a < b$ 的关系，建一条有向边 $a \rightarrow b$ ， $v=1$ ，表示 b 至少需要比 a 大1；对于 $a \leq b$ 的关系，建一条有向边 $a \rightarrow b$ ， $v=0$ ，表示 b 不能比 a 小，最后将所有节点初始值设为1（每人至少分一个糖果），最后求出最长路径即可。

三. 代码实现:

```
#define _CRT_SECURE_NO_DEPRECATE
/*****
*创建时间: 2018 09 08
*文件类型: 源代码文件
*题目来源: BZOJ
*当前状态: 已通过
*备忘录: 图论 最短路径 差分约束
*作者: HtBest
*****/
#include <stdio.h>
#include <iostream>
#include <string>
#include <stdlib.h>
#include <string.h>
#include <algorithm>
#include <queue>
#include <bitset>
// #include <sys/wait.h>
// #include <sys/types.h>
// #include <unistd.h>
using namespace std;
#define MAXN 100010
#define MAXM 1000010
int n,m,head[MAXN],_edge,d[MAXN],times[MAXN];
struct EDGE
{
    int a,b,v,next;
    EDGE(int a=0,int b=0,int v=0,int next=0):a(a),b(b),v(v),next(next){}
}edge[MAXM];
/* Variable explain:

*/
void adde(int a,int b,int v)
{
    edge[++_edge]=EDGE(a,b,v,head[a]);
    head[a]=_edge;
}
void read()
{
    int ls1,ls2,ls3;
    scanf("%d%d",&n,&m);
    for(int i=1;i<=m;++i)
```

```

{
    scanf("%d%d%d",&ls1,&ls2,&ls3);
    switch(ls1)
    {
        case 1:
            adde(ls2,ls3,0);
            adde(ls3,ls2,0);
            break;
        case 2:
            adde(ls2,ls3,1);
            break;
        case 3:
            adde(ls3,ls2,0);
            break;
        case 4:
            adde(ls3,ls2,1);
            break;
        case 5:
            adde(ls2,ls3,0);
    }
    }
    return;
}
void spfa()
{
    deque<int> q;
    bitset<MAXN> vis;
    for(int i=1;i<=n;++i)vis[i]=d[i]=1,q.push_back(i);
    while(!q.empty())
    {
        int a=q.front();
        q.pop_front();
        vis[a]=0;
        ++times[a];
        if(times[a]>n)printf("-1"),exit(0);
        for(int i=head[a];i;i=edge[i].next)
        {
            int b=edge[i].b,v=edge[i].v;
            if(d[b]<d[a]+v)
            {
                d[b]=d[a]+v;
                if(!vis[b])vis[b]=1,q.push_back(b);
            }
        }
    }
}
int main()
{
    // freopen(".in","r",stdin);
    // freopen(".out","w",stdout);
    read();
    spfa();
    long long ans=0;
    for(int i=1;i<=n;++i)ans+=d[i];
    printf("%lld\n",ans);
    return 0;
}

```

[<题目跳转>](#) [<查看代码>](#)