

# 地震损坏

## 一. 考察内容:

贪心 图论 连通性 图的遍历

## 二. 题目分析:

[题目大意]

一个无向图，图上有一些节点是损坏的，不能通过，给出一些限制条件例如x，表示x是好的，但是x到1之间的道路是不通的，求出至少有多少点是与1点不通的。

[写题思路]

这个题其实很简单，但是思路很有意思（其实也很好想）。

我们考虑对于每个限制条件，在他们到1的路径上断掉一个点，究竟断掉哪个点呢？我们贪心的认为，断掉与该点直接连通的所有点即可，因为这些点一定是不能与1连通的（如果可以连通，那么该限制点也可以连通了），于是我们为这些需要断掉的点打上标记，从1节点开始遍历整个图，只能访问没有标记的点，访问到的节点都是可以互通的，其余节点都不能互通。

## 三. 代码实现:

```
#define _CRT_SECURE_NO_DEPRECATE
/*****
*创建时间: 2018 09 04
*文件类型: 源代码文件
*题目来源: BZOJ
*当前状态: 已通过
*备忘录: 图的遍历 连通分量
*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 30001
#define MAXM 100001
int n,m,k,head[MAXN],_edge;
bitset <MAXN> vis;
struct EDGE
{
    int a,b,next;
    EDGE(int a=0,int b=0,int next=0):a(a),b(b),next(next){}
}edge[2*MAXM];
/* Variable explain:
n: 节点数
m: 边数
k:message数
head[i]: 节点i的第一条出边
edge[i]: 邻接表
_edge: 邻接表标记
vis: 保存节点是否阻塞 | 是否访问过
```

```
*/
void adde(int a,int b)
{
    edge[++_edge]=EDGE(a,b,head[a]);
    head[a]=_edge;
}
void read()
{
    int ls1,ls2;
    scanf("%d%d%d",&n,&m,&k);
    for(int i=1;i<=m;++i)scanf("%d%d",&ls1,&ls2),adde(ls1,ls2),adde(ls2,ls1);
    return;
}
int dfs(int a)
{
    int ans=0;
    for(int i=head[a];i;i=edge[i].next)
    {
        int b=edge[i].b;
        if(vis[b])continue;
        vis[b]=1;
        ans+=dfs(b);
    }
    return ans+1;
}
int main()
{
    // freopen(".in","r",stdin);
    // freopen(".out","w",stdout);
    read();
    int ls1;
    while(k--)
    {
        scanf("%d",&ls1);
        vis[ls1]=1;
        for(int i=head[ls1];i;i=edge[i].next)
            vis[edge[i].b]=1;
    }
    vis[1]=1;
    printf("%d\n",n-dfs(1));
    return 0;
}
```

[<题目跳转>](#) [<查看代码>](#)