# 1232 畅通工程

### 一、题目

#### 问题描述

某省调查城镇交通状况，得到现有城镇道路统计表，表中列出了每条道路直接连通的城镇。省政府“畅通工程”的目标是使全省任何两个城镇间都可以实现交通（但不一定有直接的道路相连，只要互相间接通过道路可达即可）。问最少还需要建设多少条道路？

#### 输入数据

测试输入包含若干测试用例。每个测试用例的第1行给出两个正整数，分别是城镇数目N ( < 1000 )和道路数目M；随后的M行对应M条道路，每行给出一对正整数，分别是该条道路直接连通的两个城镇的编号。为简单起见，城镇从1到N编号。   
注意:两个城市之间可以有多条道路相通,也就是说  
3 3  
1 2  
1 2  
2 1  
这种输入也是合法的  
当N为0时，输入结束，该用例不被处理。

#### 输出数据

对每个测试用例，在1行里输出最少还需要建设的道路数目。

#### 输入样例

4 2 1 3 4 3 3 3 1 2 1 3 2 3 5 2 1 2 3 5 999 0 0

#### 输出样例

1 0 2 998

#### 题目来源

HDU 1232 http://acm.hdu.edu.cn/showproblem.php?pid=1232

### 二、题解

#### 解题思路

并查集入门题目，同学们可以通过此题了解并查集这一数据结构类型并初步掌握应用方法。 相互连接的城市构成一个集合，只需要判断集合个数即可知道要修多少条路。 集合个数的判断也可以根据每个集合只有一个根节点的特征，找n个数里有几个根节点，并减去1。

#### 参考程序

#include <stdio.h>  
#define MAX 1000  
int father[MAX];  
//初始化函数  
void Init(int n)  
{  
 int i;  
 for(i=1;i<=n;i++)  
 father[i]=i;  
}  
//查找函数  
int Find(int x)  
{  
 while(father[x]!=x)  
 x=father[x];  
 return x;  
}  
//合并函数  
void combine(int a,int b)  
{  
 int temp\_a,temp\_b;  
 temp\_a=Find(a);  
 temp\_b=Find(b);  
 if(temp\_a!=temp\_b)  
 father[temp\_a]=temp\_b;  
}  
//确定连通分量个数  
int find\_ans(int n)  
{  
 int i,sum=0;  
 for(i=1;i<=n;++i)  
 if(father[i]==i)  
 ++sum;  
 return sum;  
}  
int main()  
{  
 int i,n,m,a,b;  
 while(scanf("%d",&n)!=EOF)  
 {  
 if(!n) break;  
 Init(n);  
 scanf("%d",&m);  
 for(i=0;i<m;++i)  
 {  
 scanf("%d%d",&a,&b);  
 combine(a,b);  
 }  
 printf("%d\n",find\_ans(n)-1);  
 }  
 return 0;  
}

#### 复杂度分析

无

#### 编程技巧

无