题意必须是环里的点度都为2才算一个环

题目大意：给定一个n表示图的顶点个数，一个m表示图的边的个数，然后是m对数x y，表示x和y节点之间有一条边。求该图有几个部分组成，这些组成部分中又有几个是环。

思想：并查集判断部分的个数，用点的度来判断是否是环，点构成环的充要条件是该环中每个点的度皆为2；

 1、初始化每个点的度为0，每个点所属集合（根节点）为其自身，标记值flag=1；

 2、对于每组输入数据，相应的点的度加1；查找两个点的根节点，如果不相等则合并之；

 3、遍历每个点，如果其度不为2，找到其根节点，将其根节点标记flag=0，表示该集合不可能为环；

 4、遍历查找答案，看per[i]与i是否相等，若相等，则部分数加1，再判断其标记值是否为1，若是则环数加1；

**如果一个并查集里点的度都是2它肯定是个环**

#include <cstdio>

#include <cstring>

#include<iostream>

#include<algorithm>

using namespace std;

typedef long long ll;

const int mod=1000000007;

int fa[1000010],num[1000010];

int degree[100010];

int n,m;

bool b[100010];

int Find(int x)

{

if(fa[x]==x)return x;

return fa[x]=Find(fa[x]);//是否需要带路径压缩还要考虑一下

}

void Merge(int x,int y)

{

x=Find(x);

y=Find(y);

if(x!=y)

fa[y]=x;

}

int main()

{

//freopen("input.txt","r",stdin);

int x,y;

while(~scanf("%d%d",&n,&m))

{

if(n==0 && m==0)return 0;

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

{

fa[i]=i;

degree[i]=0;

b[i]=1;

}

for(int i=1;i<=m;i++)

{

scanf("%d%d",&x,&y);

degree[x]++;degree[y]++;

Merge (x,y);

}

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

if(degree[i]!=2)

b[Find(i)]=0;

int num(0),numc(0);

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

if(fa[i]==i)

{

num++;

if(b[i])

numc++;

}

printf("%d %d\n",num,numc);

}

return 0;

}