题意：给你一个不完全的图，问你最少移动多少节点可以使图变成树

题目思路：

　　【并查集】【模拟】

　　用并查集把成环的归在一起(类似强连通分量)，然后统计分量数并修改。

第一个出现的当作根，其余的每一块连通分量都去掉一条边改为连接到根上。

就算是有环，读入的时候经过合并处理了之后，每个环也都有一个顶点了，也就是每个环中也都有一个点fa[i]==i了

对于并查集的每一个集合，只修改他们的根节点的指向就可以了，其他的不同管，它们最终还是指向根节点

#include <cstdio>

#include <cstring>

#include<iostream>

#include<algorithm>

using namespace std;

int fa[200010],a[200010];

int n,m,x,y;

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

//return Find(f[x]); //不路径压缩版

}

int main()

{

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

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

{

memset(fa,0,sizeof(fa));

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

{

fa[i]=i;

}

int ans(0);

int root=0;

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

{

scanf("%d",&a[i]);

if(a[i]==i)

root=i;

int x=Find(i),y=Find(a[i]);

if(x!=y)

fa[x]=y;

}

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

cout<<fa[i]<<' ';cout<<endl;

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

if(fa[i]==i)

{

if(root==0)//有环

{

root=i;

ans++;

a[i]=i;

}

else if(i!=root)

{

ans++;

a[i]=root;

}

}

printf("%d\n",ans);

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

printf("%d ",a[i]);

printf("%d\n",a[n]);

}

return 0;

}