#include <bits/stdc++.h>  
using namespace std;  
  
const int N = 100005;  
  
struct DSU {  
  
 int Parent[ N ];  
 int Tot[ N ];  
  
 inline void Makeset( int k ) {  
 for( int i = 0; i <= k; i++ ) {  
 Parent[ i ] = i; Tot[ i ] = 1;  
 }  
 }  
 inline int Find( int x ) {  
 if( Parent[ x ] == x ) return x;  
 return Parent[ x ] = Find( Parent[ x ] );  
 }  
 inline void Union( int x , int y ) {  
 int u = Find( x ) , v = Find( y );  
 if( u != v ) Parent[ u ] = v , Tot[ v ] += Tot[ u ];  
 }  
} dsu;  
  
int main( int argc , char const \*argv[] ) {  
  
 dsu.Makeset( 50 );  
  
 return 0;  
}