|  |
| --- |
| #include <iostream> |
|  | #include <fstream> |
|  | using namespace std; |
|  |  |
|  | int szamol(int v[100][100],int n,int x){ |
|  |  |
|  | int szamlalo =0; |
|  | //megszamoljuk hogy hany 1es van az xedik sorban; |
|  | for(int i=0;i<n;i++){ |
|  | if(v[x][i]==1){ |
|  | szamlalo++; |
|  | } |
|  | } |
|  | return szamlalo; |
|  | } |
|  |  |
|  | int main() |
|  | { |
|  | /\* |
|  | n- csomopontok szama |
|  | m- osszekottetesek szama |
|  | \*/ |
|  | ifstream cin("input.txt"); |
|  | int n,m,v[100][100]; |
|  | //cout <<"hany csomopont van a grafban ? "; |
|  | cin >> n; |
|  | //cout << "ezek kozzul hany van osszekotve ? "; |
|  | cin >> m; |
|  |  |
|  | //lenullazuk a szomszedossagi matrixot |
|  | for(int i=0;i<n;i++){ |
|  | for(int j=0;j<n;j++){ |
|  | v[i][j]=0; |
|  | } |
|  | } |
|  | //beolvasuk az osszekotott pontokat es ennek fuggvenyeben irjuk at a szomszedossagi matrixot |
|  | for(int i=0;i<m;i++){ |
|  | int x,y; |
|  | //cout <<"add meg a ket csomopontot amelyik ossze van kotve : "; |
|  | cin >> x >> y; |
|  | //hogy szimetrikus legyen a matrix es ne legyen semmi problema mindket oldalrol megkozzelitve irjuik at az erteket 1esre |
|  | //pl ha x1 es y2 akkor az 0. sor 1. elemet is es a 1. sor 0. elemet is atcsereljuk 1esre |
|  | v[x-1][y-1]=1; |
|  | v[y-1][x-1]=1; |
|  | } |
|  |  |
|  | //1. alpont - beolvasuk a keresendo csomopontot |
|  |  |
|  | int x,y; |
|  | //cout << "adj meg ket csomopontot : "; |
|  | cin >> x >> y; |
|  | // megnezzuk hogy az x-1. sor y-1. tagja 1es e? ha igen akkor kiirjuk hogy a ket pont ossze van kotve.ha nem akkor kiirjuk hogy nincs |
|  | if(v[x-1][y-1]==1){ |
|  | cout << "a ket pont ossze van kotve!"<<endl; |
|  | }else{ |
|  | cout << "a ket pont nincs osszekotve!"<<endl; |
|  | } |
|  |  |
|  | //2. feladat -fuggvenyt irunk a felhasznalasra mert a 3. alpontnal is felhasznaljuk; |
|  | //cout << "add meg a csomopontot : "; |
|  | cin >> x; |
|  | cout << szamol(v,n,x-1)<< " osszekottetes van a csomopont fele"<<endl; |
|  |  |
|  | //3. feladat - keszitunk egy vektort amely n sort es ket oszlopor,egyik a csomopontot masik az oszekottetesek szamat tartalmazza |
|  | int csomopont[100][2]; |
|  | for(int i=0;i<n;i++){ |
|  | csomopont[i][0]=i+1; |
|  | csomopont[i][1]=szamol(v,n,i); |
|  | } |
|  | //rendezzuk a tombot a 2. oszlop szerint |
|  | for (int i=0; i<n-1; i++) |
|  | { |
|  | for (int j=i+1; j<n; j++) |
|  | { |
|  |  |
|  |  |
|  |  |
|  | if ( csomopont[i][2]> csomopont[j][2]) |
|  | { |
|  | swap(csomopont[i][2],csomopont[j][2]); |
|  | swap(csomopont[i][1],csomopont[j][1]); |
|  |  |
|  | } |
|  |  |
|  | } |
|  |  |
|  | } |
|  | for(int i=0;i<n;i++){ |
|  | cout << csomopont[i][0]+1 <<" ("<<csomopont[i][1]<<" kapcsolat) "<<endl; |
|  | } |
|  | return 0; |
|  | } |