class Program

{

static void Main()

{

int V = 8;

int[] visited = new int[V];

int[,] graph = new int[V, V];

Dictionary<int, int> komp\_sv = new Dictionary<int, int>();

List<(int, int)> tree = new List<(int, int)>();

komp\_sv[0] = 1;

graph[0, 1] = 1;

graph[0, 2] = 1;

graph[1, 2] = 1;

graph[0, 4] = 1;

graph[2, 4] = 1;

graph[1, 4] = 1;

graph[3, 6] = 1;

graph[3, 7] = 1;

graph[6, 7] = 1;

graph[3, 5] = 1;

graph[5, 7] = 1;

for (int j = 0; j < V; j++)

{

Dfs(graph, j, visited, V, komp\_sv);

}

Console.WriteLine("Вершина \t Компонента связности");

foreach (var d in komp\_sv)

{

Console.WriteLine(d.Key + "\t\t\t" + d.Value);

}

}

static int FindMaxValue(Dictionary<int, int> D)

{

int max = 0;

foreach (var d in D)

{

if (d.Value > max) max = d.Value;

}

return max;

}

static void Dfs(int[,] graph, int tek, int[] visited, int V, Dictionary<int, int> D)

{

if (!D.ContainsKey(tek)) D[tek] = FindMaxValue(D) + 1;

visited[tek] = 1;

for (int to = 0; to < V; to++)

{

if ((graph[tek, to] == 1) && (visited[to] != 1))

{

D[to] = D[tek];

Dfs(graph, to, visited, V, D);

}

}

}

}