**Usman Siddiqui(9042)**

using System;

class Discrete

{

static int V = 4;

void TransitiveClosure(int [,]matrix)

{

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

int i, j, k;

for (i = 0; i < V; i++)

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

reach[i, j] = matrix[i, j];

for (k = 0; k < V; k++)

{

for (i = 0; i < V; i++)

{

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

{

path[i, j] = (path[i, j] != 0) ||

((path[i, k] != 0) &&

(path[k, j] != 0)) ? 1 : 0;

}

}

}

printSolution(path);

}

void printSolution(int [,]path)

{

Console.WriteLine("Following matrix is transitive" +

" closure of the given graph matrix");

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

{

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

Console.Write(path[i, j] + " ");

Console.WriteLine();

}

}

public static void Main (String[] args)

{

int [,]matrix = new int[,]{{0, 0, 0, 1},

{1, 0, 1, 0},

{1, 0, 0, 1},

{0, 0, 1, 0}};

Discrete d = new Discrete();

d.TransitiveClosure(matrix);

}

}