**Assignment**

**Name: Umair Mehdi**

**COURSE: DM**

**S.ID: 61880**

**C.ID: 103347**

**Code:**

using System;

class GFG

{

    static int V = 4;

    void transitiveClosure(int [,]graph)

    {

        int [,]reach = new int[V, V];

        int i, j, k;

        for (i = 0; i < V; i++)

            for (j = 0; j < V; j++)

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

        for (k = 0; k < V; k++)

        {

            for (i = 0; i < V; i++)

            {

              for (j = 0; j < V; j++)

                {

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

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

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

                }

            }

        }

        printSolution(reach);

    }

    void printSolution(int [,]reach)

    {

        Console.WriteLine("Matrix is transitive" +

                          " closure of the graph");

        for (int i = 0; i < V; i++)

        {

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

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

            Console.WriteLine();

        }

    }

    public static void Main (String[] args)

    {

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

                                  {0, 1, 1, 0},

                                  {0, 0, 1, 1},

                                  {0, 0, 0, 1}};

        GFG g = new GFG();

        g.transitiveClosure(graph);

    }

}

**Output:**

