GRAPH COLOURING PROBLEM

```
import java.util.*;
public class GraphColoring {
   static int V; // Number of vertices
   static int[] colors; // Colors assigned to vertices
   static boolean isSafe(int v, int[][] graph, int color, int[] colors) {
      for (int i = 0; i < V; i++)
         if (graph[v][i] == 1 && color == colors[i])
            return false;
      return true;
   }
   static boolean graphColoringUtil(int[][] graph, int m, int v, int[] colors) {
      if (v == V)
         return true;
      for (int c = 1; c <= m; c++) {
         if (isSafe(v, graph, c, colors)) {
            colors[v] = c;
            if (graphColoringUtil(graph, m, v + 1, colors))
               return true;
            colors[v] = 0; // Backtrack
         }
      }
      return false;
   }
   static void printSolution(int[] colors) {
      System.out.println("Vertex   Color");
      for (int i = 0; i < V; i++)
```

```
            System.out.println("  " + i + "      " + colors[i]);
    }
    static boolean graphColoring(int[][] graph, int m) {
        colors = new int[V];
        Arrays.fill(colors, 0);
        if (!graphColoringUtil(graph, m, 0, colors)) {
            System.out.println("Solution does not exist");
            return false;
        }
        printSolution(colors);
        return true;
    }
    public static void main(String[] args) {
        V = 4; // Number of vertices
        int[][] graph = {{0, 1, 1, 1},
                         {1, 0, 1, 0},
                         {1, 1, 0, 1},
                         {1, 0, 1, 0}};
        int m = 3; // Number of colors
        graphColoring(graph, m);
    }
}
```

OUTPUT: