**Name:** Rohan Arun Nalawade

**Roll No:** 231012

**PRN:** 22310407  
**SY IT A**

**Assignment 7 A**

**Code implementation:**

#include <iostream>

#include <vector>

using namespace std;

bool isSafe(int vertex, const vector<vector<int>>& graph, const vector<int>& color, int c, int V) {

    for (int i = 0; i < V; i++) {

        if (graph[vertex][i] == 1 && color[i] == c) {

            return false;

        }

    }

    return true;

}

bool graphColoringUtil(const vector<vector<int>>& graph, int m, vector<int>& color, int vertex, int V)

{

    if (vertex == V) {

        return true;

    }

    for (int c = 1; c <= m; c++) {

        if (isSafe(vertex, graph, color, c, V)) {

            color[vertex] = c;

            if (graphColoringUtil(graph, m, color, vertex + 1, V)) {

                return true;

            }

            color[vertex] = 0;

        }

    }

    return false;

}

void printSolution(const vector<int>& color) {

    cout << "Solution Exists: Following are the assigned colors:\n";

    for (int c : color) {

        cout << c << " ";

    }

    cout << endl;

}

int main() {

    int V;

    cout << "Enter number of vertices: ";

    cin >> V;

    vector<vector<int>> graph(V, vector<int>(V));

    cout << "Enter the adjacency matrix:\n";

    for (int i = 0; i < V; i++) {

        for (int j = 0; j < V; j++) {

            cin >> graph[i][j];

        }

    }

    int m;

    cout << "Enter the number of colors: ";

    cin >> m;

    vector<int> color(V, 0);

    if (graphColoringUtil(graph, m, color, 0, V)) {

        printSolution(color);

} else {

cout << "No solution exists." << endl;

}

return 0;

}

**Output:**

