

## Name: Implementation of Backtracking to solve Map Coloring Problem.

In this problem you are going to solve the Map Coloring problem using backtracking.

Here the input map will be presented as a graph. Nodes of the graph will present unique cities. In the first line there will be given two integer variables, V and E denoting the number of nodes of the input graph and the adjacency information respectively. In each of the following E lines, you will be given two integer variables u ( $1 \leq u \leq V$ ) and v ( $1 \leq v \leq V$ ) denoting which nodes are adjacent to each other. After that, you will be given a single integer variable K denoting the maximum number of colors that can be used to color the graph.

You need to write a program that will calculate if it is possible to color the map using at most K colors, maintaining the constraint that the adjacent nodes will always have different colors. Print a valid configuration if it is possible else print a single string "NO" (without quotes). A valid configuration means a valid assignment of the colors to the nodes denoting V values for V nodes and  $i^{\text{th}}$  color will denote the assignment for the  $i^{\text{th}}$  node.

Please look at the input output section for more clarification.

Test Cases:

Input	Output
7 9 1 2 1 3 2 3 2 4 3 4 3 5 3 6 4 5 5 6 4	2 1 0 2 1 2 0
8 9 1 2 1 3 1 4 1 5 3 4 2 6 6 7 7 8 5 8 7	0 1 1 2 1 0 1 0

89 12 13 14 15 34 26 67 78 58 2	NO
---	----