RBU, Nagpur
CSE III Sem
PRACTICAL NO. 8-CP Task

Name:	**Kishan Saini**
Sec-Batch-Rollno:	**A4-B4-55**

Aim: Competitive Programming Code Submission

Q) You are given an undirected graph consisting of V vertices and E edges represented by a list edges[][], along with an integer m. Your task is to determine whether it is possible to color the graph using at most m different colors such that no two adjacent vertices share the same color.

Return true if the graph can be colored with at most m colors, otherwise return false.

Note: The graph is indexed with 0-based indexing.

Code:class Solution:

```
    def graphColoring(self, v, edges, m): g =
        [[] for _ in range(v)]
        for u, w in edges:
            g[u].append(w)
            g[w].append(u) color

        = [0] * v

        def isSafe(node, c): for
            nei in g[node]:
                if color[nei] == c:
                    return False
            return True
```

```
def solve(node):
    if node == v:
        return True
    for c in range(1, m + 1): if
        isSafe(node, c):
            color[node] = c
            if solve(node + 1):
                return True
            color[node] = 0
    return False

return solve(0)
```

Submission