Graph Coloring Problem - backtracking

Code:

In [6]:

```
V = 4
m = 3
graph = [[0, 1, 1, 1],
         [1, 0, 1, 0],
         [1, 1, 0, 1],
         [1, 0, 1, 0]]

def isSafe(v, colour, c):
    for i in range(V):
        if graph[v][i] == 1 and colour[i] == c:
            return False
    return True

def graphColourUtil(m, colour, v):
    if v == V:
        return True

    for c in range(1, m+1):
        if isSafe(v, colour, c) == True:
            colour[v] = c
            if graphColourUtil(m, colour, v + 1) == True:
                return True
            colour[v] = 0

    return False

def graphColouring(m):
    colour = [0] * V
    if graphColourUtil(m, colour, 0) == False:
        return False
    print("Solution exist and Following are the assigned colours:")
    for c in colour:
        print(c, end=' ')
    return True

graphColouring(m)

#Output
```

Out[6]:

```
False
```