AI EX 2

EDGE COLOURING

AIM: An edge colouring of a graph, is a colouring of the edges of such that adjacent edges(or the edges bounding different regions) receive different colours. An edge colouring containing the smallest possible number of colours for a given graph is known as minimum edge colouring

ALGORITHM :

Initialize:

Color first vertex with first color.

Loop for remaining V-1 vertices.:

1. Consider the currently picked vertex and color it with the lowest numbered color that has not been used on any previously colored vertices adjacent to it.

2. If all previously used colors appear on vertices adjacent to v, assign a new color to it.

3. Repeat the following for all edges.

4. Index of color used is the chromatic number.

OPTIMIZATION TECHNIQUE:

Graph coloring problem is to assign colors to certain elements of a graph subject to certain constraints.

Vertex coloring is the most common graph coloring problem. The problem is, given m colors, find a way of coloring the

vertices of a graph such that no two adjacent vertices are colored using the same color.

The other graph coloring problems like Edge Coloring (No vertex is incident to two edges of same color) and Face Coloring (Geographical Map Coloring) can be transformed into vertex coloring.

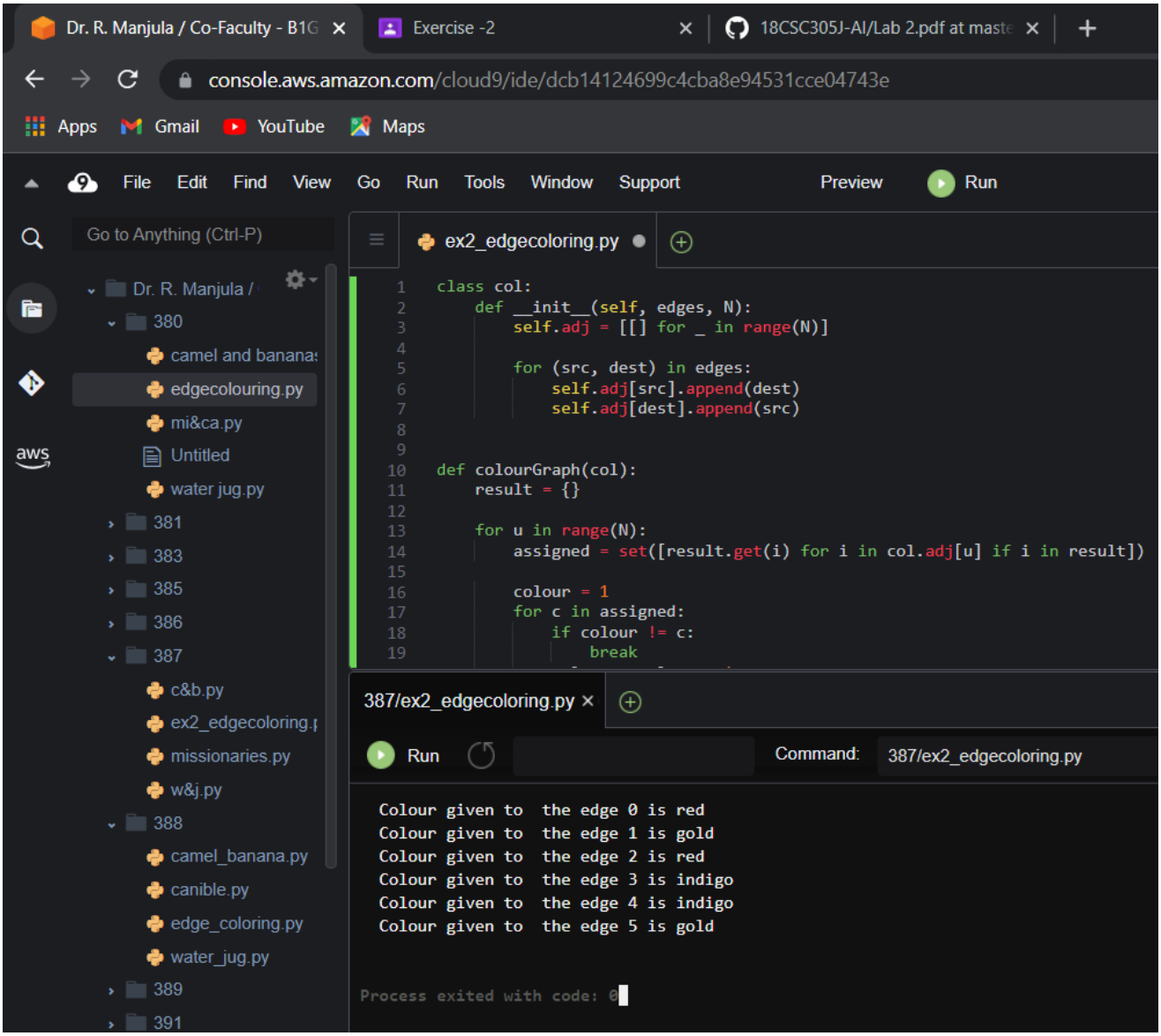
Chromatic Number: The smallest number of colors needed to color a graph G is called its chromatic number. For example, the following can be colored at least 2 colors.

CODE:

```
class col:
    def __init__(self, edges, N):
        self.adj = [[] for _ in range(N)]

        for (src, dest) in edges:
            self.adj[src].append(dest)
            self.adj[dest].append(src)



def colourGraph(col):
    result = {}

    for u in range(N):
        assigned = set([result.get(i) for i in col.adj[u] if i in result])
```

```
        colour = 1
        for c in assigned:
            if colour != c:
                break
            colour = colour + 1

        result[u] = colour

    for v in range(N):
        print("  Colour given to  the edge", v, "is", colours[result[v]])



if __name__ == '__main__':
    colours = ["", "red", "gold", "indigo", "purple", "light green",
"yellow",
               "blue", "violet", "grey", "white", "orange"]
    edges = [(0, 1), (0, 4), (0, 5), (4, 5), (1, 4), (1, 3), (2, 3), (2, 4)]
    N = 6
    col = col(edges, N)
    colourGraph(col)
```

OUTPUT:



RESULT: **The edge colouring problem was successfully executed**

RA1911003010387

CHITRALEKHA.CH