AI LAB EXP – 2

DEVELOPING AGENT PROGRAMS FOR REAL WORLD PROBLEMS

Graph Coloring Problem

Date: 13-01-2022

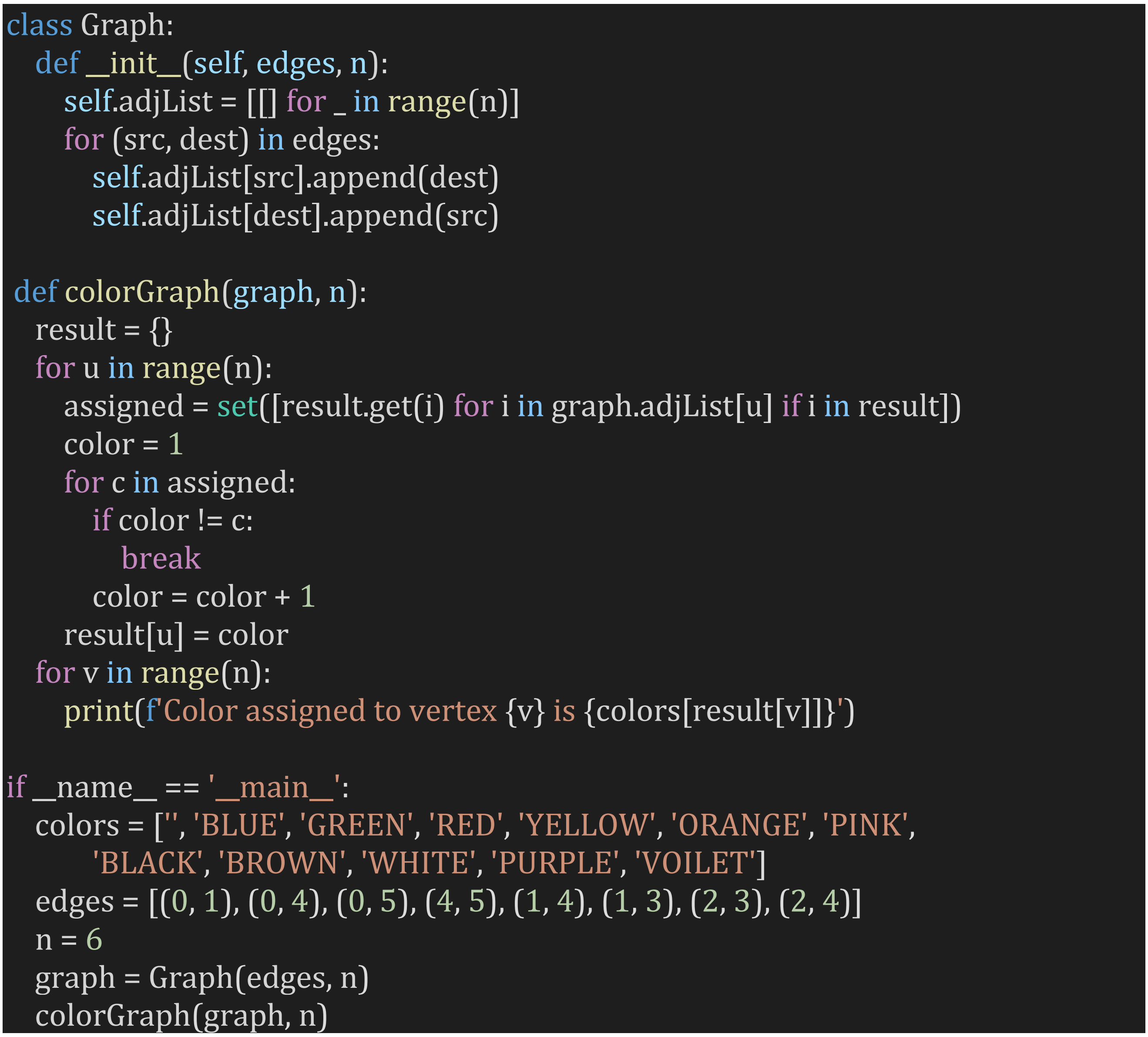
Name: VINOTH S

Reg No: RA1911030010103

CODE: (Vertex Colouring)

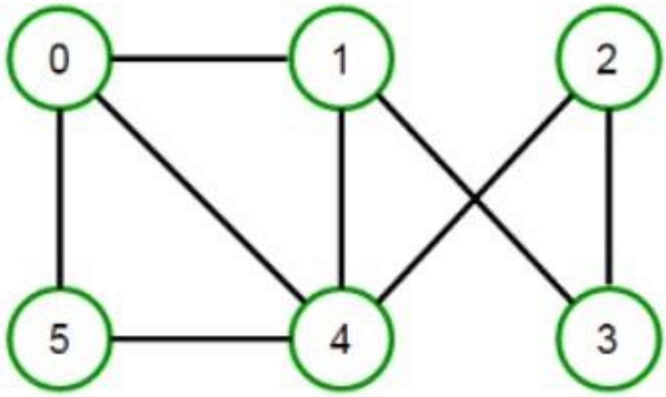
```
class Graph:
   def __init__(self, edges, n):
      self.adjList = [[] for _ in range(n)]
      for (src, dest) in edges:
         self.adjList[src].append(dest)
         self.adjList[dest].append(src)

 def colorGraph(graph, n):
   result = {}
   for u in range(n):
      assigned = set([result.get(i) for i in graph.adjList[u] if i in result])
      color = 1
      for c in assigned:
         if color != c:
            break
         color = color + 1
      result[u] = color
   for v in range(n):
      print(f'Color assigned to vertex {v} is {colors[result[v]]}')
```

```
if __name__ == '__main__':
   colors = ['', 'BLUE', 'GREEN', 'RED', 'YELLOW', 'ORANGE', 'PINK',
         'BLACK', 'BROWN', 'WHITE', 'PURPLE', 'VOILET']
   edges = [(0, 1), (0, 4), (0, 5), (4, 5), (1, 4), (1, 3), (2, 3), (2, 4)]
   n = 6
   graph = Graph(edges, n)
   colorGraph(graph, n)
```

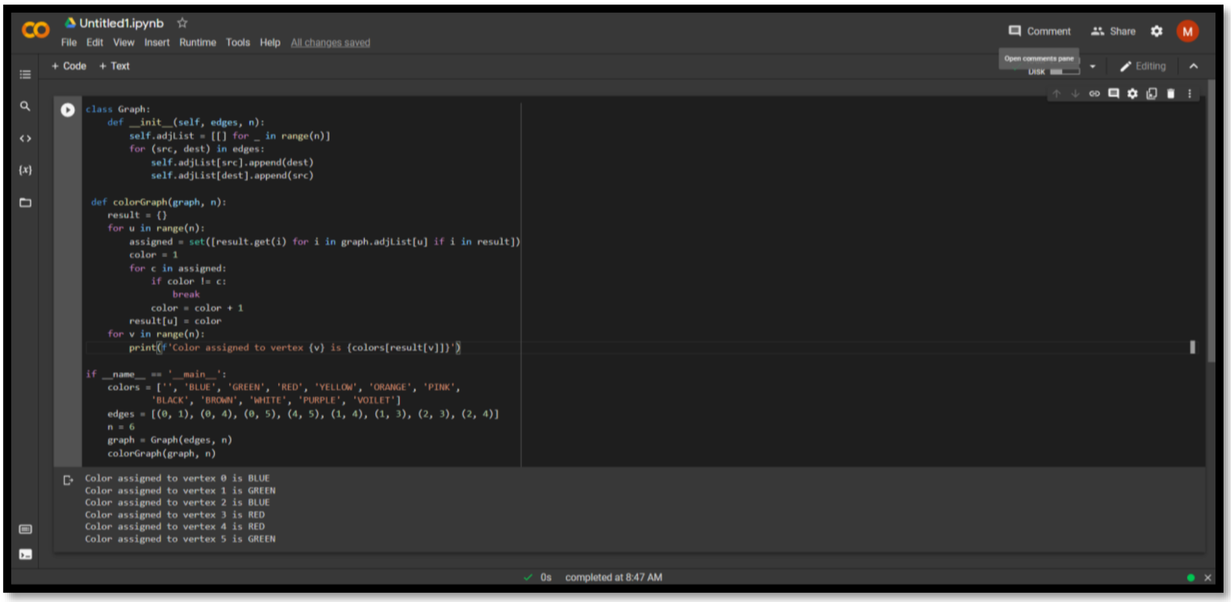


```
class Graph:
   def __init__(self, edges, n):
      self.adjList = [[] for _ in range(n)]
      for (src, dest) in edges:
         self.adjList[src].append(dest)
         self.adjList[dest].append(src)

 def colorGraph(graph, n):
   result = {}
   for u in range(n):
      assigned = set([result.get(i) for i in graph.adjList[u] if i in result])
      color = 1
      for c in assigned:
         if color != c:
            break
         color = color + 1
      result[u] = color
   for v in range(n):
      print(f'Color assigned to vertex {v} is {colors[result[v]]}')

if __name__ == '__main__':
   colors = ['', 'BLUE', 'GREEN', 'RED', 'YELLOW', 'ORANGE', 'PINK',
         'BLACK', 'BROWN', 'WHITE', 'PURPLE', 'VOILET']
   edges = [(0, 1), (0, 4), (0, 5), (4, 5), (1, 4), (1, 3), (2, 3), (2, 4)]
   n = 6
   graph = Graph(edges, n)
   colorGraph(graph, n)
```

Graph before Vertex Colouring:

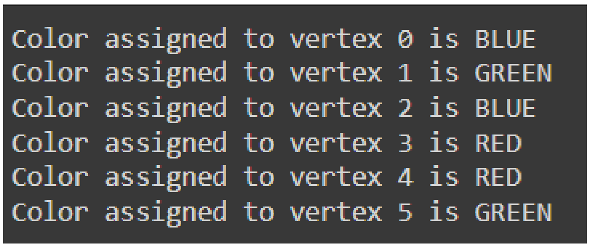


Output Screenshot:



```
class Graph:
    def __init__(self, edges, n):
        self.adjList = [[] for _ in range(n)]
        for (src, dest) in edges:
            self.adjList[src].append(dest)
            self.adjList[dest].append(src)

 def colorGraph(graph, n):
    result = {}
    for u in range(n):
        assigned = set([result.get(i) for i in graph.adjList[u] if i in result])
        color = 1
        for c in assigned:
            if color != c:
                break
            color = color + 1
        result[u] = color
    for v in range(n):
        print(f'Color assigned to vertex {v} is {colors[result[v]]}')

if __name__ == '__main__':
    colors = ['', 'BLUE', 'GREEN', 'RED', 'YELLOW', 'ORANGE', 'PINK',
              'BLACK', 'BROWN', 'WHITE', 'PURPLE', 'VOILET']
    edges = [(0, 1), (0, 4), (0, 5), (4, 5), (1, 4), (1, 3), (2, 3), (2, 4)]
    n = 6
    graph = Graph(edges, n)
    colorGraph(graph, n)
```

```
Color assigned to vertex 0 is BLUE
Color assigned to vertex 1 is GREEN
Color assigned to vertex 2 is BLUE
Color assigned to vertex 3 is RED
Color assigned to vertex 4 is RED
Color assigned to vertex 5 is GREEN
```



```
Color assigned to vertex 0 is BLUE
Color assigned to vertex 1 is GREEN
Color assigned to vertex 2 is BLUE
Color assigned to vertex 3 is RED
Color assigned to vertex 4 is RED
Color assigned to vertex 5 is GREEN
```

Graph after Vertex Colouring: