EXPERIMENT 9

Objective: Implementation of vertex and edge coloring algorithms in graphs.

Brief Theory:

Graph coloring involves assigning colors to elements of a graph under specific constraints. The two primary types are:

1. Vertex Coloring: Assign colors to vertices such that no two adjacent vertices share the same color.
2. Edge Coloring: Assign colors to edges such that no two edges sharing a common vertex have the same color.

Key Concepts:

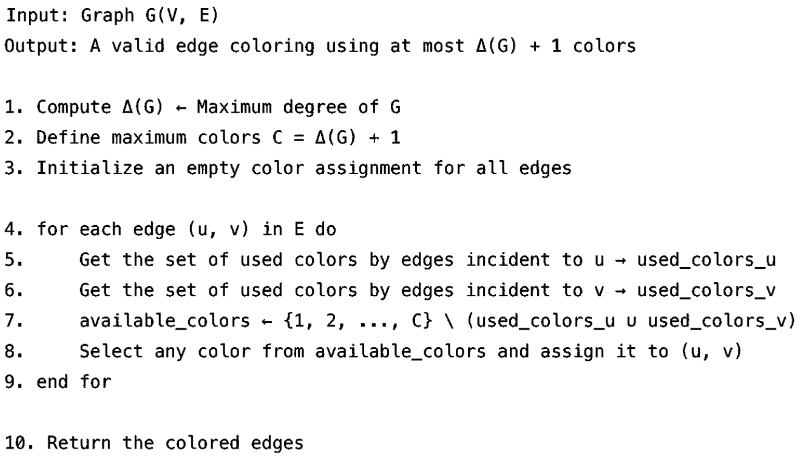
1. Chromatic Number: The minimum number of colors required to properly color a graph's vertices.
2. Chromatic Index: The minimum number of colors required to properly color a graph's edges.

Algorithms for Vertex Coloring:

1. Greedy Coloring:
   - Assign the smallest available color to each vertex, traversing the vertices in a given order.
   - Simple but may not always achieve the chromatic number.
2. Backtracking: Explore all possible color assignments using recursion, ensuring proper coloring.

Algorithms for Edge Coloring:

1. Greedy Algorithm: Assign the smallest available color to each edge while ensuring no two edges sharing a vertex have the same color.
2. Vizing's Theorem: Ensures that the chromatic index of a graph is either its maximum degree (Δ) or Δ+1. Exact coloring can be found through iterative algorithms.



```
Input: Graph G(V, E)
Output: A valid edge coloring using at most Δ(G) + 1 colors

1. Compute Δ(G) ← Maximum degree of G
2. Define maximum colors C = Δ(G) + 1
3. Initialize an empty color assignment for all edges

4. for each edge (u, v) in E do
5.     Get the set of used colors by edges incident to u → used_colors_u
6.     Get the set of used colors by edges incident to v → used_colors_v
7.     available_colors ← {1, 2, ..., C} \ (used_colors_u ∪ used_colors_v)
8.     Select any color from available_colors and assign it to (u, v)
9. end for

10. Return the colored edges
```

Task:

1) Implement a program using the greedy algorithm to color the vertices of a graph and calculate its chromatic number.
2) Write a program to find the optimal vertex coloring using a backtracking approach and verify the chromatic number.
3) Develop a program to color the edges of a graph using the greedy algorithm and calculate the chromatic index.
4) Create a program where users can input a graph (vertices and edges) and visualize the colored edges based on the vizing's algorithm.

Apparatus and components required: Computer with C or C++ Compiler and Linux/Windows platform.

Experimental/numerical procedure: Coding, compilation, editing, run and debugging.