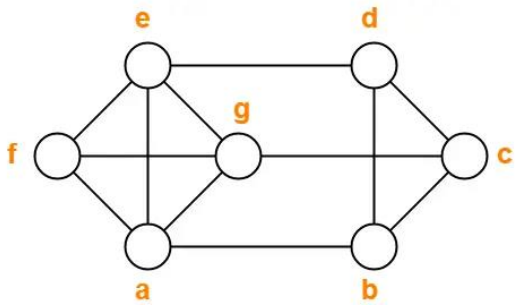
CSE 3512 – Algorithm Design and Analysis
Lab 4
Graph Theory: Graph Coloring

1. **Time Complexity Analysis of Graph Coloring**
   a) Find the chromatic number of a given graph using Graph coloring algorithm (backtracking).



   b) Analyze the time complexity of a Graph Coloring algorithm as the number of vertices (N) and edge density (D) vary.
      i) For N vertices, vary N from 50 to 200 in steps of 50. For each N, vary edge density D (percentage of possible edges that exist) from 10% to 90% in steps of 20%.
      ii) For each (N, D) combination, record the total execution time. Plot the execution time against N for each fixed D, and observe the trend.

2. **Graph Coloring for Class Scheduling**

Suppose you have N classes that need to be scheduled such that no two classes that share students are scheduled at the same time. Each class has a set of students, and if two classes share any students, they must be assigned different time slots. You need to use a graph coloring approach to assign the minimum number of time slots (colors) to each class such that no two conflicting classes are scheduled simultaneously.

Input:

- **N (number of classes)**. [Each vertex in the graph corresponds to a class.]
- **The adjacency matrix representing conflicts between classes.** [An edge between two vertices (classes) exists if those two classes share at least one student. So, if Class 1 and Class 2 share students, you'd place an edge between Vertex 1 and Vertex 2.]
- **The maximum number of time slots m** (colors) to try.

Output:

The minimum number of time slots (colors) required such that classes with overlapping students cannot be in the same time slot.