Connor Norton

CSCI-6511

**Project 2: Graph Problem**

For this project, we are given a graph in the form of a text file, that we are supposed to color. The proper vertex coloring is such that each vertex is assigned a color and no two adjacent vertices are assigned the same color.

To solve this problem, I developed a Constraint Satisfaction Problem (CSP) algorithm in python with the following components:

* A backtracking search algorithm to solve the CSP
  + *backtracking\_search(graph, coloring)*
* Heuristics
  + Minimum remaining values (MRV)
    - *select\_vertex\_MRV(graph, coloring)*
  + Least constraining value (LCV)
    - *order\_LCV(vertex, graph, coloring)*
* Constraint propagation using AC3 for maintaining arc consistency
  + *ac\_3(graph, unassigned\_vertex, coloring\_value, coloring)*

The code is run by calling the *graphColoring(graph)* method with a valid graph object. If there is no solution, the function returns *False*. If a solution is found, a list of vertices and their colorings is returned.

The Input Text File has the following format:

colors = #

#,#

#,#

#,#

.

.

.

Unit Testing

* Unit tests were developed using pytest in Visual Studio Code. The test cases covered several aspects of the program. There are tests to: check that graphs are created properly, check if the input text file is read correctly, check that no solution is returned when there is no solution, and check that a correct solution is returned when there is a known solution.