Devin Gonzales

Jordan Conragan

CS 146 Section #2

**Project 3 Report**

Project 3 consists of writing a program that automatically generates and solves a maze. Our solution begins with the Vertex class, which contains useful fields we would need to model the maze as a functioning graph. The Cell class is a subclass of Vertex that represents each individual cell in a maze and keeps track of its cardinal neighbors and location in the maze (ie row and column). With respect to the theory discussed in class, each instance of Cell represents one of many vertices with the “doorways” between cells representing the edges connecting them.

The Maze class initializes mazes through a variety of constructors as well as containing various utility methods such as print and clone. Our solution to solving mazes includes the static Solver class, whose primary purpose is to solve mazes with DFS and BFS. Solved mazes are represented as an instance of the SolvedMaze class, which holds the solution path as well as various utility methods to aid in printing and displaying the solution. TestMaze and TestSolver are meant to unit test all non-trivial functions in our program using a combination or provided mazes and randomly generated ones.