Team Member: Wenhao Li #14328112

Problem: Maze Generator and Solver

Proposed Abstract:

We have learned several algorithms throughout this semester to solve different problems. Previously, I wanted to do the N-Queen problem with algorithm comparison. Later, I figured out that was somewhat a bit simple. So I decided to make a maze solver to further utilize the algorithms we have learned..

From another perspective, the size of this problem is easy to control, only determined by the rows and columns of the maze, and the complexity grows exponentially. Therefore, it is easy to be scaled from a small to a bigger problem, which can potentially produce an intuitive result of the correct algorithm to choose from. As for the comparison, I decided to add breadth first search, depth first search, A star search and uniform cost search. All these searching methods are based on the graph search approach, since a tree search will potentially causing a circulation in solving the maze.

I am planning to compare all aspects of the algorithms used in solving the maze problem, such as CPU execution time, memory usage and time consumption. In addition, I will probably implement multi-threading in certain algorithms too. Like in breadth-first search, it is possible to implement the branching process into a multi-threading problem. I will make a GUI interface to display the maze and the solution to produce more intuition.