CS3050 Final Project Report

Matthew Carroll, Clayton Cornett, Brian Lasker, Jeffery Rohlman

**Design Process:**

The main design process started by trying to figure out each of the possible solutions to the maze problem. We knew there would be four situations the robots could find themselves in at the start:

1. Both robots could reach the exit successfully
2. Robot one could reach the exit successfully and robot two could not
3. Robot two could reach the exit successfully and robot one could not
4. Neither robot could successfully reach the exit

From here we decided the most effective way would be to run four Depth First Searches (DFS) on the maze, searching through each situation looking for a solution. If one robot could reach the end, then it is pathed through the maze to the exit and the second robot is left alone in the room to do as he pleases. If neither could path through the maze, it terminates without a solution. Sections 1-3 have multiple subcases, where a robot can move out of the way to provide access to the path for the other robot. If need be, a robot makes the minimum amount of moves to get out of the way trying to give a path to the robot.

**Efficiency of Algorithms:**

Because a DFS operates at an efficiency of O(|V| + |E|), our four DFS functions all operate at O(|V| + |E|). Meaning at max, the DFS operates at O(n) efficiency with “n” being all possible vertices in the maze. The load in and conversion functions both take O(n) efficiency so at max, the entire efficiency of the program would be O(n).

**Contribution by Team Members:**

Everyone contributed equally to the project. Jeff and Brian focused on the DFS algorithm design where Matt and Clay focused on the README.md, the project report, loading in the file, and error checking.