Eduardo Ramirez Torres

CS 32

**Homework 2 Problems**

2. Our main program looks like this, where “S” represents our starting position and “E” stands for our end goal:

"XXXXXXXXXX",

"X...X..X**E**X",

"X.XXX....X",

"X.X.XXXX.X",

"XXX**S**.....X",

"X...X.XX.X",

"X.X.X..X.X",

"X.XXXX.X.X",

"X..X...X.X",

"XXXXXXXXXX"

In our stack program, we are performing a depth-first search, meaning that we will check one path fully before checking the others. Our push order goes east, south, west and then north, with the pop order going backwards. Therefore, our first 12 coordinates popped off will be: (4,3), (3,3), (5,3), (5,2), (5,1), (6,1), (7,1), (8,1), (8,2), (6,3), (4,4), and (4,5).

4. Again, our main program looks like this:

"XXXXXXXXXX",

"X...X..X**E**X",

"X.XXX....X",

"X.X.XXXX.X",

"XXX**S**.....X",

"X...X.XX.X",

"X.X.X..X.X",

"X.XXXX.X.X",

"X..X...X.X",

"XXXXXXXXXX"

Our queue program is a breadth-first search algorithm, so we will explore our options equally one tile at a time. The first 12 coordinates popped off are as follows:

(4,3), (4,4), (5,3), (3,3), (4,5), (6,3), (5,2), (4,6), (5,1), (4,7), (6,1) and (4,8).

The two algorithms are different in the way that they check coordinates first. Stacks are Last In First Out, meaning that new positions are checked immediately after the loop completes. Queues are First In First Out, so each new position discovered is added to the array to be checked once every other position already in the array is checked off. This leads to the differences in solving the maze.