**2. Given the algorithm, main function, and maze shown at the end of problem 1, what are the first 12 (r,c) coordinates popped off the stack by the algorithm?**

(6, 4)

(6, 3)

(6, 5)

(7, 5)

(8, 5)

(8, 6)

(8, 7)

(8, 8)

(7, 8)

(6, 6)

(5, 4)

(4, 4)

**4. Given the same main function and maze as are shown at the end of problem 1, what are the first 12 (r,c) coordinates popped from the queue in your queue-based algorithm?**

**How do the two algorithms differ from each other? (Hint: how and why do they visit cells in the maze in a different order?)**

(6, 4)

(5, 4)

(6, 5)

(6, 3)

(4, 4)

(6, 6)

(7, 5)

(3, 4)

(4, 5)

(8, 5)

(2, 4)

(4, 6)

The stack based algorithm is a depth-first search of the maze. It explores a path from the starting coordinate as far as possible before backtracking and exploring another path. This is due to the first in, last out nature of the stack. A path from a coordinate pushed onto the stack is the first to be popped off, so we keep examining the path until it is exhausted.

The queue based algorithm is a breadth-first search of the maze. It explores the neighbors of the current coordinate first before moving on to another coordinate and exploring that coordinate’s neighbors. This is due to the first in, first out nature of the queue. A path from a coordinate is enqueued to back of the queue, and thus all neighbors of a coordinate are explored first from the front of the queue before a path is taken.