hw

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?

(3, 4) (3, 3) (3, 5) (2, 5) (1, 5) (1, 6) (1, 7) (1, 8) (2, 8) (3, 6) (4, 4) (5, 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?

(3, 4) (4, 4) (3, 5) (3, 3) (5, 4) (3, 6) (2, 5) (6, 4) (5, 5) (1, 5) (7, 4) (5, 6)

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

With a stack, the algorithm would likely implement a depth-first search (DFS) approach where cells are visited in a top-to-bottom and left-to-right order. This means that the algorithm would explore as far as possible along each branch before backtracking. In a stack implementation, new cells are always inserted and removed from the same end. This approach allows the algorithm to explore a path as far as possible before backtracking, which is ideal for a DFS approach.

On the other hand, with a queue, the algorithm would likely implement a breadth-first search (BFS) approach where cells are visited in a left-to-right and top-to-bottom order. This means that the algorithm would explore all neighbors of a cell before moving on to the next level of cells. In a queue implementation, new cells are always added at one end and removed from the other end. This approach allows the algorithm to explore all neighboring cells of the current cell before moving on to the next level of cells, which is ideal for a BFS approach.