**Ahsan, Prianna**

**SID: 704-068-040**

**Problem 2:**

The first 11 (r, c) coordinates popped off by the pathExists algorithm using a stack-based implementation are:

1. (6, 4)

2. (6, 5)

3. (6, 6)

4. (7, 5)

5. (8, 5)

6. (8, 6)

7. (8, 7)

8. (8, 8)

9. (7, 8)

10. (6, 8)

11. (5, 4)

**Problem 4:**

The first 11 (r, c) coordinates popped off by the pathExists algorithm using a queue-based implementation are:

1. (6, 4)

2. (5, 4)

3. (6, 5)

4. (4, 4)

5. (7, 5)

6. (6, 6)

7. (3, 4)

8. (4, 5)

9. (8, 5)

10. (2, 4)

11. (4, 6)

The two algorithms themselves are not different, but the data structures used to implement them are. My function enumerates the directions North, South, East, and West from 0 to 3, respective. One version of the algorithm uses a first-in, last-out data structure. Thus, the second coordinate popped off the stack is the one that was last entered (movement originating in the eastern direction (enum value 2) w/ coord (6,5). Movement originating from coord (5, 4) in the northern direction (enum value 0) isn’t explored until all routes originating from (6,5) are eliminated. The other version of the algorithm uses a first-in, first-out data structure to hold its Coords data. Thus, the second coordinate popped off the queue represents movement in the northern direction because it was the first Coords entered after the starting coordinate. The third coordinate, (6, 5) represents the first easterly step, and the fourth, (4, 4), represents the second northerly step.