**Yaman Yucel CS32 HW2:**

**Question 2: The first 12 Coords for Stack implementation**

Coord 1: (5, 3) : starting point or named as point 1

Coord 2: (6, 3) : neighbor of starting point, named as point 2

Coord 3: (4, 3) : neighbor of starting point, named as point 3

Coord 4: (4, 2) : neighbor of point 3, named as point 4

Coord 5: (4, 1) : neighbor of point 4, named as point 5

Coord 6: (3, 1) : neighbor of point 5, named as point 6

Coord 7: (2, 1) : neighbor of point 6, named as point 7

Coord 8: (1, 1) : neighbor of point 7, named as point 8

Coord 9: (1, 2) : neighbor of point 8, named as point 9 : DEAD END

Coord 10: (3, 3) : neighbor of point 3, named as point 10

Coord 11: (5, 4) : neighbor of starting point, named as point 11

Coord 12: (5, 5) : neighbor of point 11, named as point 12

**Question 4: First 12 Coords for Queue implementation**

Coord 1: (5, 3) : starting point or named as point 1

Coord 2: (5, 4) : neighbor of starting point, named as point 2

Coord 3: (4, 3) : neighbor of starting point, named as point 3

Coord 4: (6, 3) : neighbor of starting point, named as point 4

Coord 5: (5, 5) : neighbor of point 2, named as point 5

Coord 6: (3, 3) : neighbor of point 3, named as point 6

Coord 7: (4, 2) : neighbor of point 3, named as point 7

Coord 8: (5, 6) : neighbor of point 5, named as point 8

Coord 9: (4, 5) : neighbor of point 5, named as point 9

Coord 10: (4, 1) : neighbor of point 7, named as point 10

Coord 11: (5, 7) : neighbor of point 8, named as point 11

Coord 12: (3, 5) : neighbor of point 9, named as point 8

**Answer to question about difference at the next page.**

**Question 4 continued:**

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

* Stack implementation uses the depth first search methodology, rather than using old points in the stack, it prefers to check newly added points to the stack. It is intuitive since newly points are added to top of the stack and used before old stacked points. If we were searching an item in a graph, stack implementation proceeds to check items through all nodes as far as possible.
* Queue implementation uses the breadth first search methodology, rather than using new points in the stack, it prefers to check old added points to the queue. It is intuitive since old points are available to access at the queue before new points. If we were searching an item in a graph, queue implementation first checks all nodes on the same level before moving on to the next level.
* Therefore, the main difference between algorithms is that they use points in a different preference. Stack implementation firstly checks the upper left corner, and when it reached to an end starts checking other directions. Queue implementation does not check any specific corner, search is like water flowing out of the starting point. Water reaches corners at the same time.