2. The first twelve coordinates are:

**(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. The first twelve coordinates are:

**(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 only differences in my code were changing the include statement from

“#include <stack>” to “#include <queue>”

changing the stack to a queue

“stack<Coord> cstack;” to “queue<Coord> cstack;”

and changing one line from

“Coord a = cstack.top();” to “Coord a = cstack.front();”.

The differences in the algorithm are that stacks pop off the latest element to enter them, while queues pop off the earliest element to enter them. Therefore they evaluate the maze in different orders. Stacks will fully explore any branches, before entering any other branches, and queues explore outwards from the closest tiles to the farthest tiles. In addition because we push in the coordinates in the order north east south west, we will be checking the maze west south east north in a stack because it pops the last item out. For a queue, it will be the same order, that is north east south west.