Chapter 1 - Algorithms and Abstractions\Robot Mazes\Sequencing\Sequencing HW (STUDENT)\Sequencing HW.html

```
<!DOCTYPE html>

<!--
 *
 * Maze Simulator (c) by Christopher Grattoni
 * Maze Simulator is licensed under a
 * Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.
 * You should have received a copy of the license along with this work.
 * If not, see <http://creativecommons.org/licenses/by-nc-sa/4.0/>.
 *
 * Last Edited: Aug 9, 2017
 *
 -->

<html>
<head>
    <title>
        Robot Maze Simulator
    </title>

    <style>
        canvas{
            background: #000000;
        }
    </style>
    <script type="text/javascript" src="maze.js"></script>
    <script type="text/javascript" src="speed.js"></script>
    <script type="text/javascript" src="security.js"></script>
    <script type="text/javascript" src="movementfunctions.js"></script>
    <script type="text/javascript" src="engine.js"></script>
    <script>
        /**
         *
         * 1) READ THIS ENTIRE COMMENT
         * 2) AFTER READING, YOU MAY DELETE THIS COMMENT
         * 3) INSERT YOUR OWN CODE HERE TO CONTROL YOUR ROBOT.
         * 4) YOUR GOAL IS TO GET YOUR ROBOT TO THE GRAY SQUARE.
         *
         * Functions you can use:
         *      moveForward(): The robot will move forward
         *          by one square relative to the direction
         *          it is currently facing. If you move into
         *          a white square, the game continues.
         *          If you move into a gray square, you win.
         *          If you try to move into a black square,
         *          you lose the game.
         *
         *      rotateRight(): The robot will rotate to the
         *          right relative to its current orientation.
         *
         *      rotateLeft(): The robot will rotate to the
```

```
52            *          left relative to its current orientation.
53            *
54            *      goalReached(): The function returns true if
55            *          you have reached the end of the maze. It
56            *          returns false if you are still in a white
57            *          square. This function can only be called
58            *          100 times per maze to try to prevent the
59            *          game from crashing.
60            *
61            *      canMove(direction): This function returns true
62            *          if the robot can move in the specified direction
63            *          relative to its current orientation. Otherwise,
64            *          it returns false. You must replace the parameter
65            *          'direction' with one of the following arguments:
66            *              'forward'
67            *              'backward'
68            *              'left'
69            *              'right'
70            *          Note: you need to include the quotes since this function
71            *          only accepts a string as its argument.
72            *
73            *  Other programming techniques you can use:
74            *      -You can use iteration, such as 'for loops' and 'while loops'.
75            *      -You can define your own functions.
76            *      -You can define your own variables.
77            *
78            */
79           function robotInstructions()
80           {
81               // Your code here
82               moveForward();
83               moveForward();
84               moveForward();
85               rotateLeft();
86               moveForward();
87               moveForward();
88               moveForward();
89               rotateRight();
90               moveForward();
91               moveForward();
92               moveForward();
93               moveForward();
94               moveForward();
95               moveForward();
96               rotateRight();
97               moveForward();
98               moveForward();
99               moveForward();
100              moveForward();
101              moveForward();
102              rotateLeft();
103              moveForward();
104              rotateRight();
105              moveForward();
106              rotateLeft();
107              moveForward();
```

```
            rotateRight();
            moveForward();
        }
    </script>

</head>


<body onload="gameFrameworkInit()">

    <canvas id="myCanvas" width="400" height="500"></canvas>

</body>


</html>

```