

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

```

1  <!DOCTYPE html>
2
3  <!--
4  *
5  * Maze Simulator (c) by Christopher Grattoni
6  * Maze Simulator is licensed under a
7  * Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.
8  * You should have received a copy of the license along with this work.
9  * If not, see <http://creativecommons.org/licenses/by-nc-sa/4.0/>.
10 *
11 * Last Edited: Aug 9, 2017
12 *
13 -->
14
15 <html>
16 <head>
17     <title>
18         Robot Maze Simulator
19     </title>
20
21     <style>
22         canvas{
23             background: #000000;
24         }
25     </style>
26     <script type="text/javascript" src="maze.js"></script>
27     <script type="text/javascript" src="speed.js"></script>
28     <script type="text/javascript" src="security.js"></script>
29     <script type="text/javascript" src="movementfunctions.js"></script>
30     <script type="text/javascript" src="engine.js"></script>
31     <script>
32
33         /*
34         * Use four separate FOR loops to complete the maze
35         * HINT: Look back at your previous code, how can you reuse it?
36         */
37
38         /**
39         *
40         * 1) READ THIS ENTIRE COMMENT
41         * 2) AFTER READING, YOU MAY DELETE THIS COMMENT
42         * 3) INSERT YOUR OWN CODE HERE TO CONTROL YOUR ROBOT.
43         * 4) YOUR GOAL IS TO GET YOUR ROBOT TO THE GRAY SQUARE.
44         *
45         * Functions you can use:
46         *     moveForward(): The robot will move forward
47         *         by one square relative to the direction
48         *         it is currently facing. If you move into
49         *         a white square, the game continues.
50         *         If you move into a gray square, you win.
51         *         If you try to move into a black square,

```

```
52      *           you lose the game.
53      *
54      *           rotateRight(): The robot will rotate to the
55      *           right relative to its current orientation.
56      *
57      *           rotateLeft(): The robot will rotate to the
58      *           left relative to its current orientation.
59      *
60      *           goalReached(): The function returns true if
61      *           you have reached the end of the maze. It
62      *           returns false if you are still in a white
63      *           square. This function can only be called
64      *           100 times per maze to try to prevent the
65      *           game from crashing.
66      *
67      *           canMove(direction): This function returns true
68      *           if the robot can move in the specified direction
69      *           relative to its current orientation. Otherwise,
70      *           it returns false. You must replace the parameter
71      *           'direction' with one of the following arguments:
72      *           'forward'
73      *           'backward'
74      *           'left'
75      *           'right'
76      *           Note: you need to include the quotes since this function
77      *           only accepts a string as its argument.
78      *
79      * Other programming techniques you can use:
80      * -You can use iteration, such as 'for loops' and 'while loops'.
81      * -You can define your own functions.
82      * -You can define your own variables.
83      *
84      */
85      function robotInstructions()
86      {
87          for(let i=0; i<9; i++)
88          {
89              moveForward();
90          }
91          rotateLeft();
92          for(let i=0; i<13; i++)
93          {
94              moveForward();
95          }
96          rotateRight();
97          for(let i=0; i<5; i++)
98          {
99              moveForward();
100          }
101          rotateRight();
102          for(let i=0; i<13; i++)
103          {
104              moveForward();
105          }
106          rotateLeft();
107          for(let i=0; i<6; i++)
```

```
108         {
109             moveForward();
110         }
111         rotateLeft();
112         for(let i=0; i<20; i++)
113         {
114             moveForward();
115         }
116         rotateLeft();
117         for(let i=0; i<15; i++)
118         {
119             moveForward();
120         }
121         rotateLeft();
122         for(let i=0; i<12; i++)
123         {
124             moveForward();
125         }
126     }
127 </script>
128
129 </head>
130
131
132 <body onload="gameFrameworkInit()">
133
134     <canvas id="myCanvas" width="400" height="500"></canvas>
135
136 </body>
137
138
139 </html>
140
```