**(REMOVED PERSONAL DETAILS) – Digital Solutions FIA1 Draft**

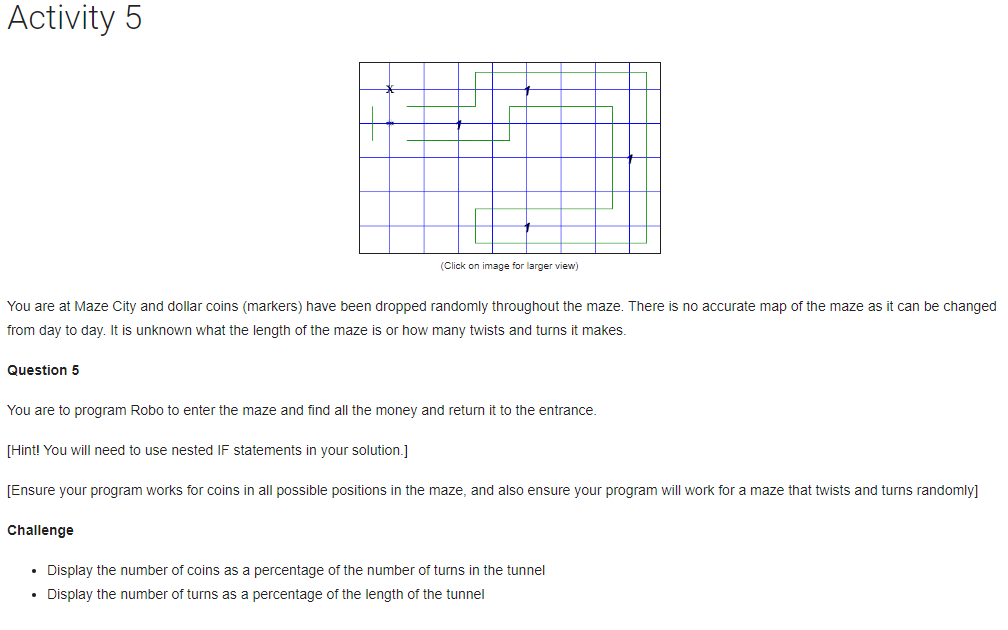
**EXPLORE- Analysis**

The Path is random every time, therefore the solution must be universally compatible to work with any combination of maze and markers. I will create a central function that will run If/ Else If statements to check whether RoboAnt can move in a direction, preventing him from running into the wall.

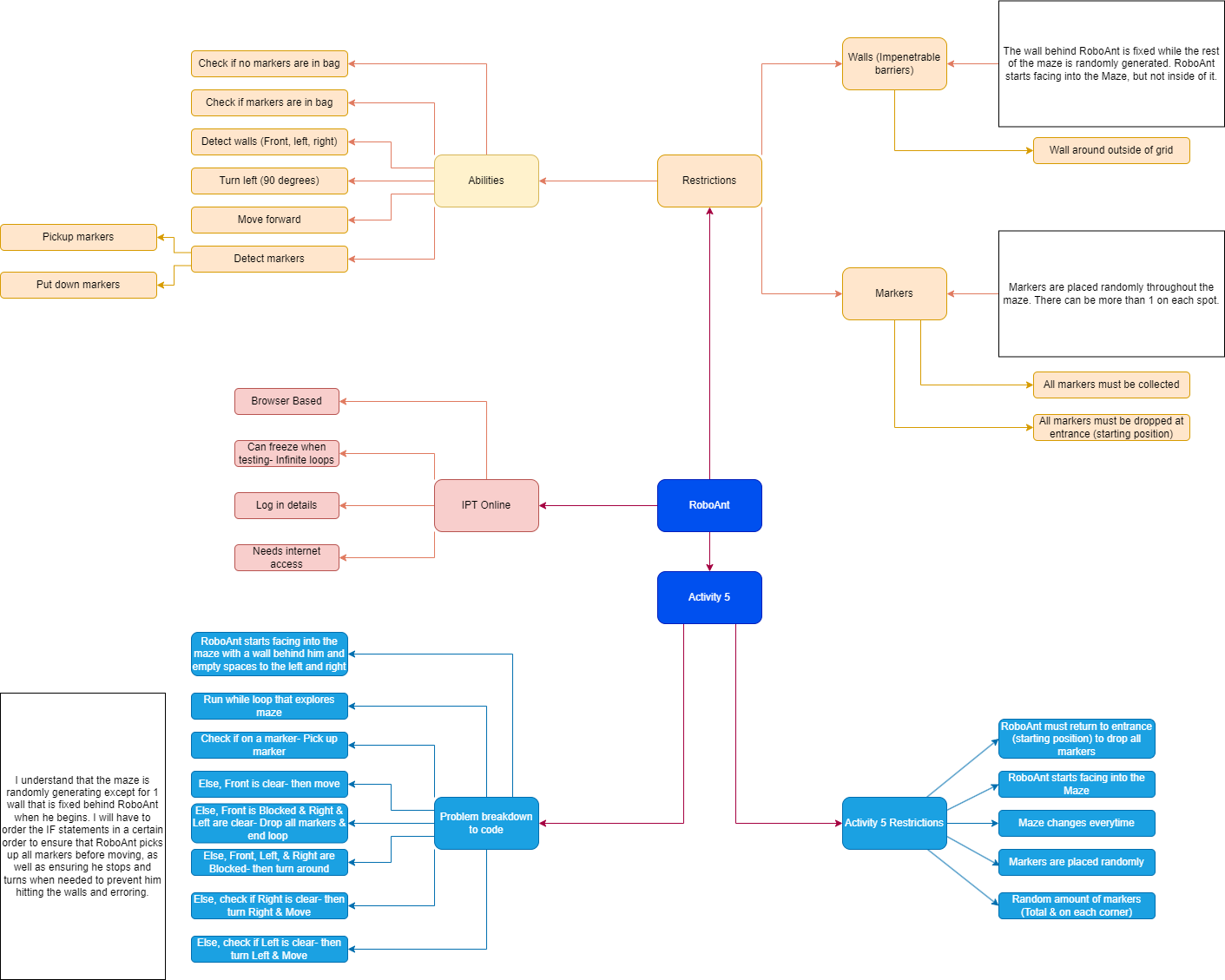
Clear instructions state that the solution will need nesting of IF statements.

Before moving throughout the maze, RoboAnt will have to check for a marker and pick all of them up if he is on one.

To complete the challenges, I will have to create multiple variables. Whenever RoboAnt moves/turns, etc the code will have to add to the variable. Then using the **Say** command I will have to divide the variables by the other then times it by a 100 to get an accurate percentage. I will also have to define what is considered a ‘turn’.

****

**EXPLORE- Mind Map**



**EXPLORE- Criteria**

**Prescribed**

In activity 5, by 06/03/23 RoboAnt will complete:

1. Be able to enter the Maze and navigate through it. (All possible combinations)
2. Pick up any markers along the path
3. Turn around at end
4. Return to entrance (starting position)
5. Drop all markers
6. Display challenge percentages
7. Use iteration, selection, and nesting to complete the task.

**Self-Determined**

In activity 5, by 03/03/23 I will complete:

1. Create functions where needed e.g. Turn right
2. Create “Maze explorer” function using If’s, Else If’s,
3. Ensure the function works for **ALL** possible maze combinations.
4. Complete task in under 50 lines of code
5. Use indentation, comments, white space, and proper variable names effectively.

**DEVELOP- Interface**

|  |  |
| --- | --- |
| **Starting interface**  There are two possible maze combinations for RoboAnt activity 5. Despite this, the solution is built to work for any possible maze combination or length. Markers are placed randomly inside the maze and in random quantities. The solution also takes this into account. | **Step 1**  Before RoboAnt can move he must first check if he is on a marker, if so, he will pick it up. |
| **Step 2**  Next RoboAnt will check if Front is clear, If so, he will move forward. | **Step 3**  Next RoboAnt will check if Front is blocked and Right & Left are clear, if so, RoboAnt will run a while loop to drop all markers he collected from the maze. After dropping all the markers, he will calculate the percentages stated inside of the challenge activities before using the SAY command to display it on the screen. RoboAnt will also increase the variable ‘stop’ which ends the main maze exploring loop. |
| **Step 4**  RoboAnt will check if the Front and Both sides are blocked, if so, he will turn around. This signifies the end of the maze. | **Step 5**  RoboAnt will check if Front & Left are blocked, if so, he will turn Right. I decided against creating a function for the Turn Right command as it is only called once during the script, adding unneeded lines of code. |
| **Step 6**  RoboAnt will check if Front & Right are blocked, if so, he will turn Left. | **Overview**  The entire procedure for RoboAnt Activity 5 can be placed inside of a while loop (Except for the variables and Change delay command). Each of these steps is run in a loop until **Step 3** if the IF statement becomes true. When **step 3** becomes true it changes the main function variable (stop) making the loop false and ending the procedure.  \*Everytime RoboAnt picks up a marker he adds 1 to the variable ‘markers’.  \*\*Every time he moves he adds 1 to the variable ‘moves’.  \*\*\*Every time he turns, he adds 1 to the variable ‘turns’. Left & Right turns are considered 1 while the turn around function in **Step 4** is counted as 2. |

**DEVELOP- Algorithm**

|  |  |
| --- | --- |
| **Maps** | **PSEUDOCODE** |
|  | Increase Movement Speed to 100 Milliseconds  Create Variable markers  Create Variable stop  Create Variable moves  Create Variable turns  WHILE stop is equal to 0  IF On a Marker & Variable stop is equal to 0  Pick Up Marker  Increase Variable markers by 1  ENDIF  ELSE IF Front is Clear  Move forward  Increase Variable moves by 1  ENDELSEIF  ELSE IF Front is Blocked & Right is Clear & Left is Clear  Increase Variable stop by 1  WHILE Any Markers in Bag  Put Down Markers  ENDWHILE  SAY Variable markers divided by Variable turns times by 100  SAY Variable turns divided by Variable moves times by 100  ENDELSEIF    ELSE IF Front is Blocked & Right is Blocked & Left is Blocked  Turn Left  Turn Left  Move forward  Increase Variable moves by 1  Increase Variable turns by 2  ENDELSEIF  ELSE IF Front is Blocked & Left is Blocked  FOR 3 times  Turn Left  ENDFOR  Move forward  Increase Variable moves by 1  Increase Variable turns by 1  ENDELSEIF  ELSE IF Front is Blocked & Right is Blocked  Turn Left  Move forward  Increase Variable moves by 1  Increase Variable turns by 1  ENDELSEIF  ENDWHILE |

**GENERATE- Code**

//Wk6 FIA1 Draft V1.8

//Increase Movement Speed

Change\_Delay\_To(100);

//Variables

var markers = 0; //Counts Markers

var stop = 0; //Stops RoboAnt from picking up markers he drops at the end

var moves = 0; //Counts every Move

var turns = 0; //Counts every turn (1 For Right & left, & 2 for turning around)

//Explore maze loop process

while (stop == 0) {

if (On\_A\_Marker() && stop==0) { // Picks Up Marker

Pick\_Up\_Marker();

markers++;

}

else if (Front\_Is\_Clear()){ //Moves forwards in Maze

Move();

moves++;

}

else if (Front\_Is\_Blocked() && Right\_Is\_Clear() && Left\_Is\_Clear()) { //Stops at End

stop++;

while (Any\_Markers\_In\_Marker\_Bag()) { //Drops all markers

Put\_Down\_Marker();

}

Say('Coins / Turns: ' + markers/turns\*100 + '%');

Say('Turns / Length: ' + turns/moves\*100 + '%');

}

else if (Front\_Is\_Blocked() && Right\_Is\_Blocked() && Left\_Is\_Blocked()) { //Turn around at end

Turn\_Left();

Turn\_Left();

turns=turns+2;

I chose to not use a function for the turn right command as it is only called once.

}

else if (Front\_Is\_Blocked() && Left\_Is\_Blocked()) { //Turns right

for (var x=1; x<=3; x++) {

Turn\_Left();

}

turns++;

}

else if (Front\_Is\_Blocked() && Right\_Is\_Blocked()) { //Turn Left

Turn\_Left();

turns++;

}

}

//END

I used iteration (while loop) here because the length of the maze is unknown and because it is the most efficient.

I had to place the ELSE IF statement for ending the loop 3rd, otherwise RoboAnt would run the turn commands instead. That would ultimately break the code.

**EVALUATION- Error Checking**

|  |  |  |
| --- | --- | --- |
| **ERROR** | **TYPE OF ERROR** | **PROBLEM/SOLUTION** |
| Infinite Loop | Logical | Made a function call another function which calls the first function.  Rewrote the entire script to avoid this problem. |
| RoboAnt/ IPTO Online Crash | Logical | Don’t use While loops with a fixed variable.  Insert a condition inside the loop that changes the variable, so it becomes **false** for the primary loop otherwise it has no possible way to end. |
| Uncaught SyntaxError: Unexpected identifier 'stop'On line: 23 | Syntax | Forgot to place an **(** at beginning of the while loop condition. |
| Uncaught SyntaxError: Unexpected token ';' On line: 48 | Syntax | Forgot to add the second **+** when increasing the increment of a variable. |
| Incorrect ordering of variables to get a percentage | Logical | Placed the wrong variable first inside a command to generate a percentage. Swapping the variables around resolved this and I got an accurate percentage. |
| Uncaught ReferenceError: Turn\_left is not defined On line: 44 | Syntax | Capitalised the L in Turn\_Left command. |

**EVALUATION- Evaluate Against Criteria**

I was successfully able to complete and achieve everything that was listed in the prescribed and self-determined criteria. I ensured that I gave myself reasonable time to plan out how I would complete the task. I ensured that I used nesting as it is a task requirement and assisted in reducing the size of the code.

I found that instead of running the main function as a function, I was able to make the function itself a while loop which reduced the overall size and complexity of the script. This allowed me to use nested IF statements as stated in the requirements.

Using a large range of IF & ELSE IF statements, I ensured that RoboAnt navigated through the maze regardless of the combination or length. I also ensured that I used appropriate indentation, comments, white space, and proper variable names and kept the code length under 70 lines.

**EVALUATION- Suggested Improvements**

I feel that my solution has completed the required task effectively. I believe that it may be possible to reduce the overall size of the code even more by using further nesting and moving variables into the maze explorer function.