Capstone Observation

Mark Bartolomeo

Curry College

Capstone Observation

In this project we had to use our knowledge we acquired over the semester to build a program which will cause Scribbler Robot to traverse a maze. To start off, I figured out first before I began coding that the speed for the Robot to move 1 inch is .45 seconds and that was what I used as the foundation in the design of the program. Utilizing loops, decisions and encapsulations where key into getting the program run efficiently. Each bit of the program was designed for a specific part of the maze. For the first exercise was just the robot traversing the maze and the second exercise was implementing sound and lights after each obstacle and turn. When I submitted the first code pass, the first prototype was a mess and it was inefficient. But it was able to get the job done. In this program first test program there was an overuse of loops and towards the end of the program it was all brute force because I just used movement blocks repetitively. Figuring out how to the robot run properly was a challenge, because I ran into the roadblock of running out of loops. I had used too many of loops embed into each other and it really made it overly complex than it needed to be. I had the goal with making the code much more tightly knit together. Which lead me to modify and tweak the timing and cut down on the loops. Instead of following .45 is equal to a 1 inch for every forward block I had to mix it up with times because it cut down on the amount of movement blocks needed and I was able to use less loops. This was able to solve the issue. Then I had to implement lights and sound into the final program. Lights are implemented when it turns left the left light on the face of the robot will light up and once that it turns right, the light on the right side will light up. When implementing the robot will play a sound once that the robot completes an obstacle. Though after I implemented it after I ran it and recorded it I discovered that it was somehow slightly angled in the second exercise but I did not come to a conclusion why it was behaving like that. I had also implemented pauses before each and see if it was the cause and it was, but the problem was it was hard to see when it would blink after each of the turns. I just kept it in any way for the sake of the recording.