Lab 6 Report

[Task 2 – Path Planning with Wavefront Planner]

Task two is where I started the lab so it's where I'll start the report. In this task we are using the Wavefront Planner algorithm for 1) planning a path from start to goal and 2) telling the robot to execute it. It wasn't too difficult to implement once I decided to use 2d lists for the grid and drew out where the neighbors would be for looping purposes. Ive done several routing and path finding algorithm projects in Operating Systems and the Algorithms courses but ive never had to send a robot out to follow my plan, which made this task kind of fun despite that also being the hard part for me. I had trouble figuring out how to use the generated plan to work the robot because for the first time this semester I used my favorite IDE (with a python extension) to code up the controller and had to port that code over to the Webots application. That is when I remembered we had to deal with the goofy while loop-timestep structure. I got it working eventually and I'm very glad I got to enjoy one last lab this semester.

[Task 1 – Wall Mapping]

The most trouble I had by far was task 1. Since I didn't have time to attempt the Monte Carlo particle filter localization algorithm in Lab 5, I decided to give it a go this time around. I based the entire thing off the methods I just recently learned for Exam 3, and it all made sense to me up until the motion update. Why is it that the first calculation and resampling get a close estimation of which cell you're in but the more you "motion update" and resample the less accurate it gets? I must have missed a step. Either way I used most of the algorithm to help narrow down the cell and used my Lab 5 functions to print the grid. Then I threw out the code on accident and figured it out using an if statement.

It's been a tough semester and the busiest I've had as a student but that's what I'd expect from a computer engineering degree's endgame. As much as I wanted to hate some of the labs it was beyond rewarding when things worked out in the end, and I really did learn a lot which feels great in hindsight. I imagine the workforce is going to be filled with plenty of these highs and lows. Wish me luck.

Calculations [YouTube : https://youtube.com/playlist?list=PLmQVFU1FBDddYV_4IRW1zfXH6CAKuZjIM]

Finding 4-Point Connectivity Neighbors in 2D list {Planner = [i [j] ← 9x9 matrix of integers}

```
up = planner[ i-1 ] [ j ]
down = planner[ i+1 ] [ j ]
left = planner[ i ] [ j-1 ]
right = planner[ i ] [ j+1 ]
```

Finding 4-point Connectivity Neighbors in 1D list {Cells[i] ← 81 cell objects stored in list}

```
if index > 8:
        up=  cells[i-9].value
if index not in east column:
        right = cells[i+1].value
if index < 72:
        down = cells[i+9].value
if index not in west column:
        left = cells[i-1].value
```

After Generating Wavefront Plan, Find Shortest path

Beginning from the start cell locate whichever neighbor has the lowest value and moving to it until you reach the goal cell value of 2