EEGR 409: C Programming Applications

Project 2.2

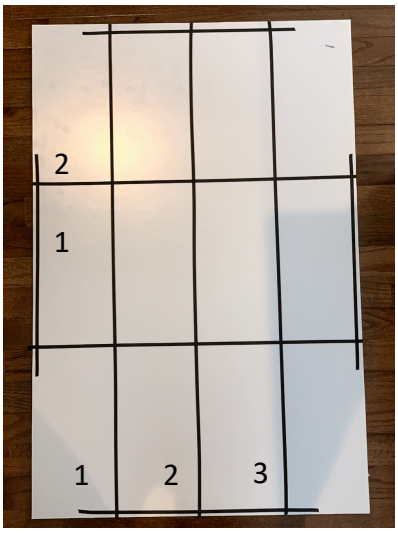
Goal:
The general goal of the project is to construct a prototype robot that can navigate in a manner similar to warehouse robots like Kiva, which is used in Amazon warehouses. The low-cost robotic platform will navigate using black guidance tape that will be laid in a matrix pattern. The robot will navigate by following the navigation line and making appropriate decisions at junctions and at the end of each navigation line.

Objective:
The objective of project 2.2 is to program the robotic platform developed in project 2.1 to navigate an environment using a grid of lines for navigation.

Caution:
- You can inadvertently short your Raspberry Pi by placing it on a conducting surface or allowing wiring leads to touch.
- You can destroy your Raspberry Pi through a buildup of static electricity on your body which you discharge into the board (discharge static buildup before touching the Pi).
- Do not place any voltages above 3.3v on any of the GPIO pins
- Do not place any load on the GPIO pins that draws more than 16mA

Remember:
- Backup all code on your SD card (copy to flash drive or upload to the cloud)

Navigation Grid:
Create a grid of 3 lines by 2 lines on a light-colored platform (see figure to the right). You can do this on a large poster board, or directly on a suitable ground surface. Make sure that the lines are spaced far enough apart to allow the vehicle to turn and have enough space to follow the line until the next junction.



Navigate Across Single Intersection:
Modify your robot class to include the following method:
- *Turn*: This will turn the robot 90 degrees in the direction provided as an argument

Place the robot at the start of the middle line (along the dimension with the most lines). In your *main* function, direct the robot to follow the line,

turn to the left at the intersection, and move until the next intersection by calling the following methods of the robot class:

- FollowLine(*timeout*)
- Turn(*direction)*
- FollowLine(*timeout*)

Navigate Across Grid

Once you've demonstrated that your robot can navigate a single intersection, add the following method to your robot class:

- *Navigate*: Will move the vehicle from a given started coordinate to an ending coordinate.

Number each line in your grid, starting from 1 (on the left side) and increasing by 1 (see figure). This way, you should be able to specify a starting position of (1,1) and an ending position of (3,2). The *Navigate* function will have parameters to define the starting and ending positions and execute the needed *FollowLine* and *Turn* methods move the vehicle to the proper intersection.

***What to submit**: A video and associated .java code demonstrating the navigate the grid.*