Control of Mobile Robotics
CDA4621
Spring 2017
Lab 4
Path Planning
Total: 100 points
Due Date: 4-24-17 by 8am

A. **Lab Requirements**

The lab requires use of the course robotic hardware ("Robobulls-2017") provided to students at no charge for the duration of the course. Required software can be downloaded free of charge from the web. All labs are to be done by teams of two students. Note that no diagrams or descriptions by hand will be accepted. Each student is required to submit his or her joint report through CANVAS. Penalties will apply to any individual student submitting a late assignment even if the partner has already submitted it. Accepted documentation file types are PDF, Word (doc, docx), and Powerpoint (ppt, pptx). You need to upload all your code together with the document as part of a "zip" or "rar" file. Any corrupt files uploaded to Canvas will not be considered and will not be graded. This includes incorrect or private links to videos. Other than external links to videos, no external links to other documents or images will be considered, such as google drives, etc.

- Hardware Requirements

The "Robobull-2017" (Figure 1) is the main robot hardware used for the course.

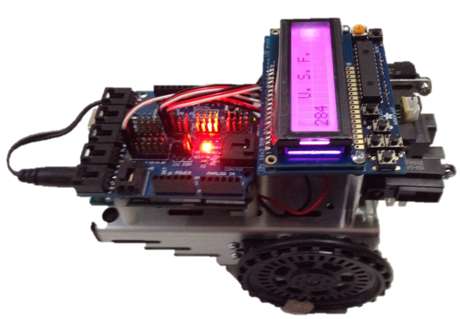

Figure 1: Robobulls-2017

- Software Requirements

Arduino Software (Version 1.8.1 or later)
https://www.arduino.cc/en/Main/Software

B. **Task Evaluation**

Each individual task is worth a specific number of points where these points are always split 50% between Task Execution and Task Report:

- Task Execution

The robot should execute the task correctly with a video clearly and completely showing the task execution (points will be taken for errors or missing aspects of task execution).

- Task Report

Each task report requires an accompanying document to be uploaded to Canvas together with ALL the files required to run the program in the robot. The task report needs to include ALL the following sections (points will be taken off if anything is missing):

1. Task description.
2. Flowchart describing the detailed solution used to solve the task. (Example of flowchart: https://en.wikipedia.org/wiki/Flowchart.
3. Code sections with detailed explanations showing how they relate to each block in the flowchart used to solve the task (this is in addition to "ino" code submitted as separate files).
4. Images taken from the actual robot task execution (at least one image per task).
5. Conclusions where you analyze any issues you encountered when running the task and how these could be improved.
6. Video link to different task executions (you should split each task execution as a different video link or as separate sections in a single video most preferably in YouTube). Provide at the beginning of each video your name and description of the task being performed.

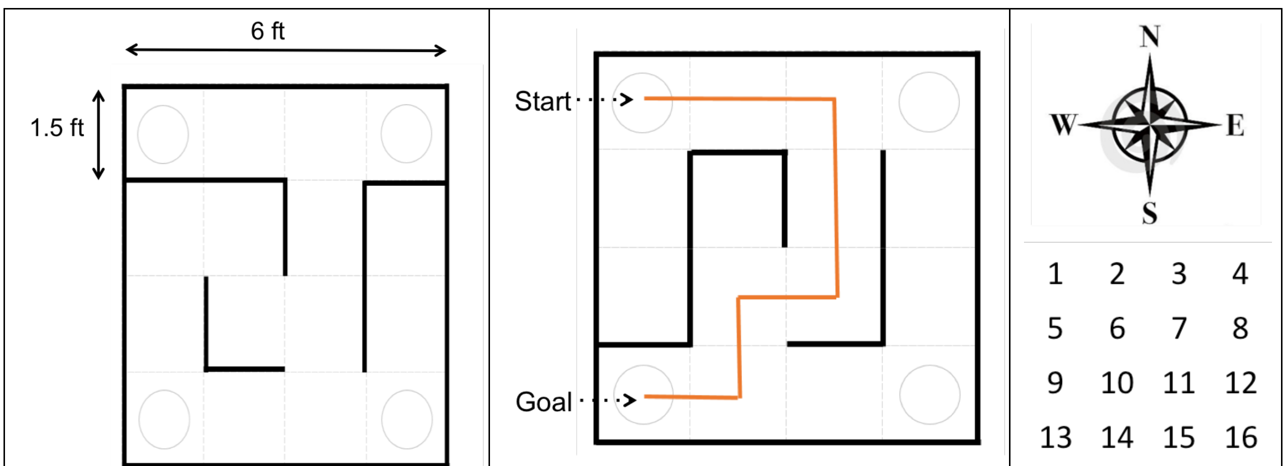

Figure 2. Left: Maze size is 6x6 sq. ft. consisting of a 4x4 non-marked grid cells of 1.5x1.5 sq. ft. Center: Sample robot route between starting cell "1" oriented "E" and goal cell "13". Right: Maze cell enumeration and global orientation.

C. **Task Description**

The goal of this assignment is to develop a path-planning algorithm to navigate an arbitrary maze from any starting robot location and orientation to any arbitrary goal location in the maze. The start and goal locations are given to the robot at the beginning of the task. The same path-planning algorithm has to work for different mazes. Figure 2 (left) shows a sample maze where grid cell size will be kept constant for all maze configurations, and only internal maze walls will change. Figure 2 (center) shows a sample path from starting location (cell "1" oriented "E") and ending location (cell "13"). Figure 2 (right) shows cell numbering and orientations. The robot needs to navigate the maze from start to goal following the shortest route to the goal by using any path-planning algorithm. The robot needs to stop at the goal at least for 5 seconds; otherwise navigation will not count as completed. You need to explain what path-planning algorithm was used to solve the problem.

Figure 3 shows functionality you will need to program for each of the LCD input buttons.

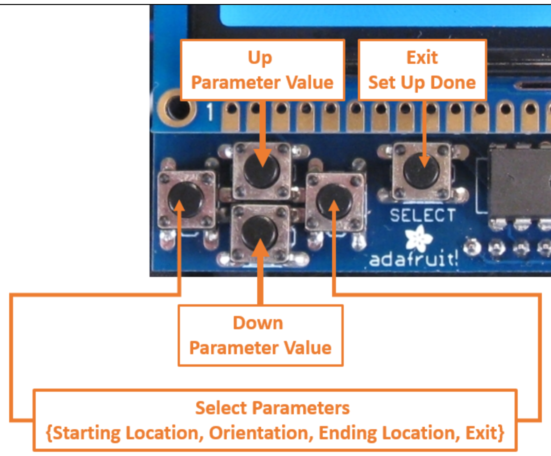

Figure 3. LCD Input buttons functionality.

Figure 4 shows examples of values assigned to each of input function buttons.

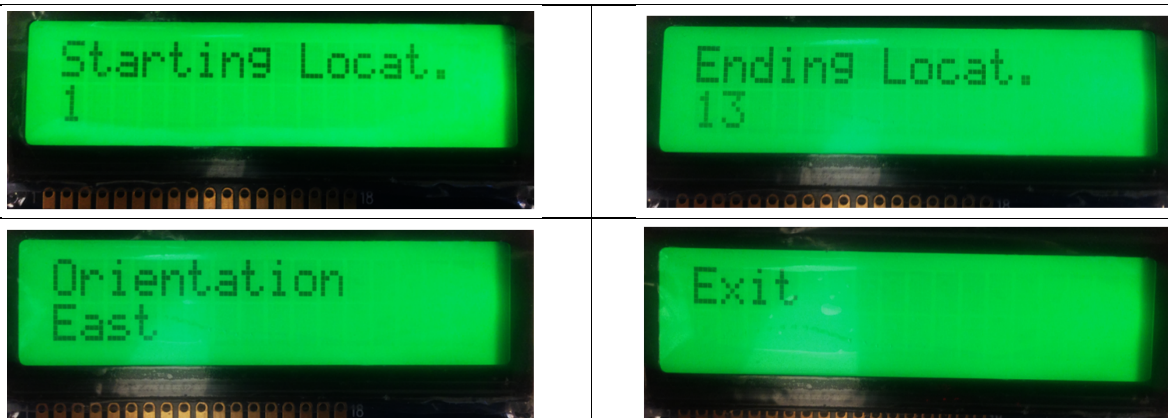

Figure 4. LCD Input buttons initial data.

Figure 5 shows sample LCD display data from Lab3 based on sensor readings that needs to be shown for both known and unknown mazes.

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	X	X	X	0	0	0	X	0	0	X	X	0	X	X	0	0
1	G	1		W	U		N	X		S	0		E	0		

Figure 5. Robot Path and Wall Information as shown in LCD

1. Path Planning with Known Maze (50 points)

The objective of Task 1 is to develop a program to allow the robot to find the shortest path for the 3 mazes shown in Figure 6, from any arbitrary start location and orientation to any arbitrary goal location. Each of the three maze configurations, shown in Figure 6, will be programmed in advance into the robot while the start and goal locations will be given only at the beginning of the task. Similar to Lab3, the robot LCD display needs to flash when moving between cells and provide a status on localization as previously shown in Figure 5. The robot needs to stop for at least 5 seconds at the goal.

2. Path Planning with Unknown Maze (50 points)

The objective of Task 2 is to develop a program to allow the robot to find the shortest path for a previously unknown maze configuration from any arbitrary start location and orientation to any arbitrary goal location. Similar to Lab3, the robot LCD display needs to flash when moving between cells and provide a status on localization as previously described in Figure 5. The robot needs to stop for at least 5 seconds at the goal.

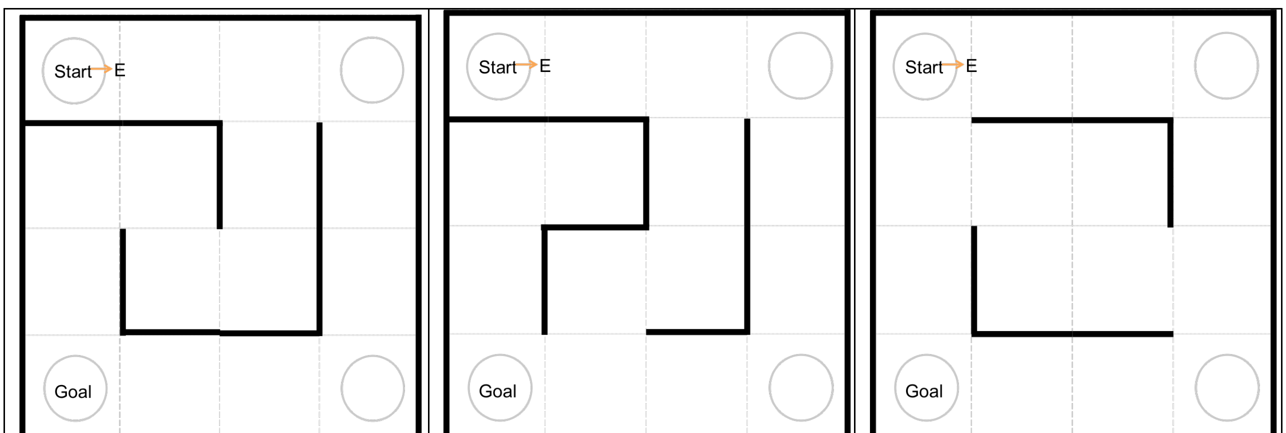

Figure 6. The 3 maze configurations with start and goal locations. Note that in all the shown cases the robot starts at Grid “1” oriented “E”.