Final Exam Individual Coding Project ENEE 148A Due: 15 December 2015

This final code development is an individual effort that counts as 50% of your final exam grade. The code must be uploaded before midnight on 15 December.

 The year is 2020 and the unthinkable has happened – global cyber warfare. The internet is gone. All major communication systems – wired or wireless – are not functional. Communication from one community to the next is now done with Morse code flashed from giant spotlights. Your job is to write a code that senses the “spotlight” with a photoresistor and translates the Morse code into legible words. The specifications for the Raspberry Pi code and hardware setup are given below:

There will be a transmitter-RPi that puts out in a random way sentences that have no more than 200 characters each. The transmitter-RPi will transmit between 2 and 5 sentences. A single red LED will be used to transmit the sentences in Morse code.

You will write your code and run it on your receiver-RPi. You need to provide the receiver RPi and related hardware, though you are welcome to borrow a sensor and resistors from the lab. You are permitted to place your sensor as close to the LED as you wish without actually touching the LED.

We will use *almost* standard Morse code conventions:

* A dot will turn the LED on for 0.2 seconds.
* A dash will turn the LED on for 0.6 seconds.
* The LED will be off for 0.2 seconds between elements (dots and dashes)
* The LED will be off for 0.6 seconds between characters
* The LED will be off for 1.4 seconds between words

In addition, we will use the following non-standard conventions

* The LED will be off for 2.8 seconds between sentences
* Transmission will end when any combination of elements does not correspond to any of the letters of the alphabet; numbers and special characters will not be included in the sentences.

Your code must conform to all of the standard coding practices that we have discussed in class. It must be as efficient as possible. Your code must:

* Output each sentence on a separate line both on the monitor and on a text file that includes up to the first 6 characters of your last name as the first part of the file name: e.g. Lawson.txt
* Print “end of transmission” both on the monitor and text file when the transmission end character is encountered.
* Terminate after printing the “end of transmission” notice.

You are free to implement your code as you see fit, but points will be lost for inefficient coding, inadequate formatting, comments, variable name selection, etc., even if the output is correct. YOU ARE REMINDED THAT THIS IS AN INDIVIDUAL EFFORT!!! Please contact the instructor or teaching fellows for any questions you might have.

Final Group Project ENEE 148A Fall 2015

Mission official Start date: Nov 13th – 18th

Complete Mission by: Dec 9th - 11th during lab

Turn in Final Report: December 15th 11:59 PM

The Tank Maze is shown in the figure below:



There are 10 moveable barriers that are approximately 1 ft high and 1 ft long. They are very thin, except for their feet. Their positions will be varied for each trial.

The tanks must be autonomous, and controlled by an Arduino, a Raspberry Pi, or similar device. They must be battery-powered.

The start and end points are indicated by a black circle on the floor (10 in diameter). The end point has a black/silver magnet at the center of the circle. Any combination of sensors may be used to navigate the course from start to end. Some part of the tank must touch the end circle to complete the first half of the mission. All sensors must be turned off for the return trip. The mission is deemed a complete success if the tank touches some part of the start circle, after first having touched the end circle, assuming that the tank HAS NOT touched any of the barriers in the classroom or the lab desk in the middle of the room. This is not a timed trial, but some weight will be given to the relative time needed to complete the mission.

The magnet strength is given by the table below:

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Distance (in)** | **12** | **10** | **8** | **6** | **4** | **2** | **1** |
| **B field (G)** | **2** | **3** | **5** | **9** | **21** | **65** | **161** |

Final Group Project Preparation ENEE 148A Due: 10 Nov 2015

Your lab time for the final project begins on Nov 13th or 18th, but you must do some preparatory work before then, to hand in (as a group assignment) BEFORE the beginning of lab on that date. The paper you turn in will have;

1. The names of everyone in your group
2. A group name
3. A list of the hardware you initially intend to use to complete the project
4. A pseudo code that you initially intend to use to write your code
5. A list of tasks and who will be principally responsible for each task
6. A schedule that you hope to meet in order to complete the project on time.

This document will be graded as a part of the final project grade and worth **20%** of that total grade. You will have four lab periods to complete the final project, plus any time you spend outside of class.

Grading rubric for Final Project

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Criterion | Level 0 | Level 1 | Level 2 | Level 3 |
| **Initial Report:**Hardware neededPsuedo CodeTask AssignmentProject Schedule | Severely lacking in 3-4 of the key areas listed to the left | Severely lacking in at least one area or moderately lacking in 3-4 areas | At most moderately lacking in 1-2 areas | All key areas sufficiently developed so that good progress can be made from start |
| **Individual Participation:**Only individual grade given | Student’s lack of participation significant / causing problems for group | Student is clearly participating, but not putting in his/her fair share | Student puts in her/his fair share | Puts in more than his/her fair share |
| **Progress towards goals** | Tank can’t generally traverse from start to finish | For minimal barrier arrangement, tank can travel from start to near finish at least once. Must be in under 15 minutes. | For a multiplicity of barrier arrangements, tank can usually traverse from start to finish Must be in under 10 minutes. | For a multiplicity of arrangements, tank can find end point in <5 min; for some barrier arrangement, can return at least 50% back to start |
| **Group Effort** | Group was frequently off topic (>30%) or unprepared and ineffective during class time | Group generally came prepared to lab but at least 10% of the time was off topic (not focused on class) | Group almost always used full lab time efficiently and productively | Group hit level 3 on previous criterion or clearly spent significant time outside of lab on project |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Criterion | Level 0 | Level 1 | Level 2 | Level 3 |
| **Code algorithm:**Assumes that the hardware functions perfectly; addresses only logical errors in algorithm  | Unlikely to ever result in successful navigation from start to finish | May be barrier arrangements that the algorithm can not handle (start to finish) | Likely to traverse from start to finish and may be able to return to start under required conditions | Certain to traverse from start to finish and has robust algorithm to return to start under required conditions |
| **Code comments:**Comment blocks, inline commenting and variable naming convention | Non-existent | Minimal comments, e.g. minimal title block and brief function descriptions; often non-descriptive choice of variable names | Title block has revision history, authorship;Functions have meaningful descriptions; some key sequences explained; good choice for variable names | Meets Level 2 plus has adequate commenting so that any successful 148A student could understand and modify code; ; excellent choice for variable names |
| **Code flow and functionality:**Effective programming skills – (1) Functions,(2) Variable / space allocation, (3) Program flow (tests and loops); and(4) Correct C usage (lack of logical coding / syntax errors) | Program realization severely lacking in 2-4 of the key areas listed to the left; or code does not even compile**THIS CRITERION DOUBLE-WEIGHTED** | Program realization severely lacking in one area or moderately lacking in 2-4 areas; code may die under some conditions**THIS CRITERION DOUBLE-WEIGHTED** | Program realization at most moderately lacking in 1 area; code error handling adequate – code unlikely to crash**THIS CRITERION DOUBLE-WEIGHTED** | All key areas sufficiently realized in code so that group demonstrates mastery of ENEE 148A and code could be used as a model of good programming techniques for future semesters**THIS CRITERION DOUBLE-WEIGHTED** |
| **Report documentation:**Separate report to turn in with code; describes strategy and has final hardware configuration, final pseudo code, and final effort report | Severely lacking in 3-4 of the key areas listed to the left | Severely lacking in at least one area or moderately lacking in 3-4 areas | At most moderately lacking in 1-2 areas | All key areas sufficiently complete so that students at comparable skill level could use report as a guide to successfully reproduce their results. |