CS 1301 Homework 6– Stoplight

Due: Friday, October 24th, before 11:55pm PM EST.
Out of 130 points
Files to submit: hw6.py

For Help:
- TA Helpdesk – Schedule posted on class website.
- Email TAs

Notes:

PAIR PROGRAMMING Assignment: Work with your PAIR 2 partner!

For pair programming assignments, you and your partner should turn in identical assignments. Your Submission must not be substantially similar to another teams' submission. Collaboration at a reasonable level will not result in substantially similar code. Students may only collaborate with fellow students currently taking CS 1301, the TA's and the lecturer. Collaboration means talking through problems, assisting with debugging, explaining a concept, etc. You should not exchange code or write code for others.

- **Don't forget to include the required comments and collaboration statement (as outlined on the course syllabus).**
- **Do not wait until the last minute to do this assignment in case you run into problems.**
- **If you find a significant error in the homework assignment, please let a TA know immediately.**

Part I

So far you've taught your robot tricks, how to avoid walls, and even how to navigate an obstacle course. Now, you are going to teach your robot the rules of the road! In this assignment your robot will learn what to do when it approaches a stoplight.

To simulate a stoplight, construction paper of various colors (green, yellow, red, and white) will be held in front of your robot and it must determine its actions based on what the color is. The actions are as follows:

Green: go forward at full speed (meaning the speed parameter should be 1) for two seconds

Yellow: go forward at half speed (0.5) for 2 seconds

Red: stop and beep!

White: randomly turn either left or right

Breaking down the assignment

As with many problems you encounter in computer science, to most effectively solve this problem you should break it down into smaller steps that will then later work together to perform a larger task.

Finding the color in front of you

Write a function called **findColor()** that takes in a picture object as its parameter and returns a string of the color that the picture is. [You may assume you will only be looking at pictures that are primarily a piece of paper of a single color.]

Parameter: a picture

Return Value: a STRING of the picture's color (i.e. "red", "green", etc.)

Randomly turning either left or right

Write a function called **turn()** that accepts no parameters and somehow uses randomness to turn either left or right, with both being equally probable. The Myro module already has a built in heads() function that essentially "flips a coin" and returns True or False based on the result. Try to turn as close to 90 degrees as you can. Now, all safe vehicles use a turn signal before turning so you will too! Check out the Myro setLED() function, when you turn left, turn the left LED light on and off before doing so to alert the traffic around you, and do the same with the right LED when turning right.

Parameters: none

Return Value: none

Determine what action to take based on the color

Putting it all together, write a function called **stopLight()** that accepts no parameters that will determine the robot's behavior. Within this function, you should have your robot take a picture, find the picture's color, and do the proper behavior based on that color as described above. This function should keep running until it detects a "red light" (a red picture in front of it) whereupon the robot should stop, beep to indicate it is done, and then function should exit.

Parameters: none

Return Value: none

Tips and Hints:

-If your robot is taking a long time to process the large images from the newer fluke 2's, insert this into the beginning of your script: setPicSize("small")

-If you are having trouble identifying the color of the construction paper in front of you, look at how the red, green, and blue values compare to each other by taking a picture of a color, using the show() function to display it in a window, and then clicking on individual pixels. In the output you will see the pixel's location and RGB values

If you need help with the image processing functions, go here:
http://calicoproject.org/Calico_Myro#Image_processing

Part II *--- Turning it in, and Demo.*
Be sure to put the lines "from Myro import *" and "initialize()" or "init()" at the beginning of the file (after the required comments). Be sure **not** to specify the port parameter in your initialize command, such as initialize("com4"). This makes it very time consuming to grade if we have to go into your code and change the com port to the one that works on our specific system.

Reminder on collaboration statement and submission:
This is a pair assignment, but each person is responsible for their own submission. **Each group member** needs to turn in hw6.py to T-square before the deadline. Please include your name, and all your group members' names in the collaboration statement.

Demo:
Each group (**All members**) needs to come to the TA's help desk or recitation to demo the program to one of the TAs, preferably your grading TA. You will be asked questions regarding your code as well.
If one of the group members is not present for the demo, his/her grade will be based ONLY on the code portion (a possible 55 points) UNTIL they come see a TA to answer questions relating to the code.
Print out and bring a grading sheet (next page) to your demo! The TAs will likely not have copies.

Grading Criteria:

Demo (TA’s Discretion, see next page)	**75 points**
Code Grading Criteria	**55 points**

turn():

• Uses a form of randomness to choose turn direction	4 pts
• Uses setLED() properly as a turn signal	1 pt

findColor():

• Correctly named and accepts one picture parameter	2 pts
• Correctly iterates through all pixels	3 pts
• Processes the pixel and categorizes its color	5 pts
• Correctly determines the color of the construction paper	15pts

stopLight():

• Correctly named	2 pts
• Takes a picture and finds its color using a helper function	5 pts
• Moves forward at full speed if green	2 pts
• Moves forward at half of the previous speed if yellow	4 pts
• Uses helper function to turn	2 pts
• Continues running until “red” is found	10pts

Robot Navigation Assignment
TA Demonstration Grading Sheet

Group Members: ____________________

____

Demo TA: ____________________________

Grading TA (if different): _____________

20 pts________ Correctly identifies what color is in front of it

5 pts________ Uses some random function correctly to turn

20 pts________ Robot does the correct action based on the color in front of it

30 pts ________ All group members understood and could explain the code.

Total: ________ / 75