CMSC 143: Object-Oriented Programming with Robots
Lab 4: Programmable Behavior
Due October 1, 2015

Chapter 6 of the textbook discusses ways to make your robot exhibit different types of behavior. In this lab, you are asked to implement four different behaviors. You will work in teams of two, but each student should submit a copy of the program. You will work together on each behavior using an approach to programming called **Pair Programming**[1]: "One, the driver, writes code while the other, the observer, pointer or navigator, reviews each line of code as it is typed in." Each student should drive twice and navigate twice.

Before writing any code, you should write a paragraph or rough outline, sketching out the design of the behavior. After you have finished the code, you should write another paragraph commenting on how the behavior was implemented, reflecting on the differences between the design and the actual implementation.

Each behavior should be programmed as a function that runs for a specified amount of time in seconds. For each function, there should be a single outer `for` or `while` loop that does its thing for the specified amount of time. You should use each timing method at least once. Myro's `currentTime()` method is very useful for the `while` loop based approach. (Extra credit if you can do it recursively without `for` or `while`!). The `setForwardness('scribbler-forward')` and `setForwardness('fluke-forward')` functions can be used to change which way is forward for the robot's motor commands (but not the sensors!).

Behaviors

1. `lightSeeker(time)`
   A robot behavior that runs toward the light (`getLight()`).

2. `avoid(time)`
   A robot behavior that runs away from obstacles triggered by the infrared detectors (`getIR()` or `getObstacle()`).

3. `securityGuard(time)`
   Do you have a feeling your roommate is snooping through your desk drawer? Write a program that detects if the drawer is opened and takes and saves a picture of the offender and then beeps loudly.

4. `digitalCamera(time)`
   Write a function that takes and displays a picture when the light sensor is pressed. The left light sensor should take a color picture (`takePicture()`) and the right sensor should take a gray-scale picture (`takePicture('gray')`). Feel free to say "Cheese" before snapping the picture.

Lab Report

Submit an electronic copy of your lab using moodle. Your program should have your name, email, assignment description, the date, and collaboration statement at the top of the file as a comment. Your submission should be a zip file that expands to a folder with two files:

```
cmsc143-lab4-LASTNAME-FIRSTNAME/
     lab4.py
     lab4.pdf
```

Learning Objectives

∘ Create robot behaviors.   ∘ Use if statements.   ∘ Use while loops.   ∘ Program in pairs.

---

[1] http://en.wikipedia.org/wiki/Pair_programming