Homework 6

To: All students in CSCI 5802 (Spring 2018)
CC: Teaching Assistant
From: Instructor
Date: 4/19/2018
Due: 4/30/2018, Monday, 11:55 PM on Moodle.
Re: Homework Assignment 6 –Oracles, Automation, Process

There are 5 questions worth a total of 90 points. You may discuss these problems in your teams and turn in all files (Question 1 to 4) as a single zipped submission on Moodle.
Finally, turn in a peer evaluation individually (10 Points) – see details at the end of document.

PROBLEM 1. 24 Points
In the class lectures (and in Chapter 17) various forms of oracles were discussed – a model, a second implementation, properties, self-checks, a team of experts, etc.

a) Provide an analysis that qualitatively compares at least three different kinds of oracles of your choice, addressing their strengths and weaknesses with respect to at least three attributes relevant to the verification process (e.g., cost, effort, completeness). You may find it helpful to organize your comparison in a tabular format, rate the oracles using a Likert-type scale, and justify your choices for the ratings briefly.
b) For each of the oracle types that you considered, explain using examples, in what situations you would use that oracle in a verification project and why.

PROBLEM 2. 30 Points
Organizations that use agile development methodologies often take a"test first" approach when developing software – they write all unit tests for a class before actually creating the code for that class. As a developer working on the next generation of CoffeeMaker (from homework #4), you are tasked with adding a new "money manager" class. Now, when a drink is requested, the user must insert money (the hardware mechanism for this is outside of scope – you just need to perform software checks on the amount inserted). Your new "money manager" class must perform the following functions:

a) Check whether the amount inserted matches the amount of money required for the requested recipe.
b) If more money was inserted than was required, return the remainder to the user.
c) Track the amount of money spent on drinks since the CoffeeMaker was turned on (since the program began execution – you do not need to keep a permanent record).

1) (20 Points) Design this class using the "test first" development paradigm:
   a) Write a class outline – include a list of data members and methods needed to perform the above tasks, with descriptions for each.
   b) Write unit tests for the class and its methods to ensure that it performs all required functions correctly. Each test must provide test inputs, expected outputs and a brief description of what the test is intended to check.

   You do NOT need to actually program the class.

2) (10 Points) An ideal, but usually unachievable, testing goal is to automatically derive test oracles from the same specification statement that establishes the acceptable behavior of a program or module. To what extent does the "test first" approach of extreme programming achieve this goal? Briefly discuss advantages and limitations of using test cases as a specification of behavior.

PROBLEM 3. 12 Points

Exercise 22.2 from textbook: Answer the questions providing brief justifications.

PROBLEM 4. 12 Points

Exercise 22.5 from textbook. Briefly justify your choice of the testing phase and explain how your reformulation enables checking the property in the alternative phase.

PROBLEM 5. 12 Points

Suppose two kinds of fault are equally common and equally costly, but one is local (entirely within a module) and the other is inherently non-local (e.g., it could involve incompatibility between modules). If your project budget is enough to automate detection of either the local or the non-local property, but not both, which will you automate? Why?

PEER EVALUATION (10 Points, complete individually)

Provide a peer-evaluation for your team members. I will leave the format and content for you to decide, but at a minimum provide some input for each member. Include a self-evaluation. You may also discuss what worked well and what didn't for the team, what you would do again the same way and what you would do differently.