CS 483: Software Testing
Homework Assignment #3   **Assigned: Friday, Oct 27, 2017**
Total Points: 70   **Due: Sunday, Nov 06, 2017 11:55PM**

Using Assertions in Software Specification(JML)

The objective of this assignment is to get you to use the assertion language we learned in class namely, Java Modeling Language or JML on a couple of different programs

Clone the project *jml-exercise* from the course repository in github. There are two packages in the project; namely *stack* and *tictactoe*. You will see Stack.java and StatckTest.java in the first class and TicTacToeBoard. Java and TicTacToeGame.java.

As part of your assignment, you will write assertions for the *Stack* class and the *TicTacToeBoard* class. If you think it will be beneficial for your exercise to add assertions in other classes, you are free to do so as well. You will write *class-invariants*, and *pre-* & *post-conditions* for methods. For all loops, you will write *loop invariants*. You should write other assertions that you think would be appropriate.

The TicTacToeGame class has code for an interactive Tic-Tac-Toe GamePlay. Write another class (e.g. TicTacToeTest that creates board objects with preset positions and exercises its different methods).

Run your programs with the assertions using JML and see if you face any run-time assertion failure. Make a copy of the program outputs in a text file. Take note of any assertion failure. Then fix the bug(s), if there is any, causing the run-time assertion failure.

There are two prominent JML tools out there:
1. OpenJML - http://www.openjml.org/downloads/
2. Jml4c - http://www.cs.utep.edu/cheon/download/jml4c/download.php

OpenJML has both an eclipse plugin and a command line tool. Jml4c only a command line compiler (jml4c) and runtime environment. You are free to use either one of the tools. Your report should clearly mention which one did you use and your experience in using it.

Your completed assignment will consist of the following:
a) Your JML annotated source code.
b) Additionally create pdf files of all your .java source files and submit them through Moodle. This helps me mark up your submission.
c) Convert all your output interactions (the text file) into a pdf file as well and upload it through Moodle.
d) A brief (~ 1-2 pages) summary of your experiences.
  a. Describe what you did and did not like about using assertions and the JML assertion system.

     b. Highlight a few examples from your annotated code that you think are particularly effective uses of assertions.

     c. How do you think writing assertions for runtime check is different from unit testing?

     d. How useful do you think assertions are or can be in ensuring qualities of a software system?

The grading will be based on
  a) Thoroughness of the exercise and write-ups
  b) Important observations about the usefulness and obstacles associated with using runtime assertions
  c) Here is a tentative breakdown of points (out of 70):
     a. JML Annotations for TTT – 20
     b. JML Annotations for Stack – 20
     c. Proper use of Runtime Assertion Check Tool – 10
     d. Observations, reflections, and report – 20