COM S 417 (SE 417) Assignment 3

Assigned: April 1, 2015
Due: April 10, 2015, 11:59pm

Summary

In this assignment, you will get familiar with Mutation Testing. Mutation testing is often used to determine quality of test suites and/or improve them. You will write JUnit tests for a given class and then use a mutation testing tool called PIT to improve your test suite. You will also write a short report to include some details of the process. This is an individual assignment; complete it on your own.

Part 1: Writing Junit Tests

1. Download `Account.java` from Blackboard.
2. Set up JUnit 4 and write adequate test cases for `Account.java` class. Write at least 15 different test cases. Make sure that you have a green test suite (i.e., all test cases pass).
3. In your report:
   (a) Include instructions on how to run your test suite.
   (b) Describe shortly the motivation behind each test case written.

Note: If you run into any issues in the `Account` class you may fix it, but you should document any changes you make in your report.

Part 2: Mutation Testing

1. Download `PIT` tool. (http://pitest.org/). You may use the stand-alone version. There is also an eclipse plug-in provided by a third party (https://github.com/philglover/pitclipse). Read the instructions in the web-page on how to set up and run the tool.
2. Run `PIT` on your JUnit tests against the `Account` class and generate a mutation report. You should get at least 25 mutants. Save the generated report by PIT. **UPDATED 04/05/2015 : used to say 30 mutants**
3. In your report:
   (a) How many mutants do you get in total? How many are killed? Why are there still live mutants?
   (b) Make reasonably detailed explanations on at least five mutants as to why they are live. Do this by referring to the mutation operator used to produce the mutant as well as your test cases.
   (c) Go back to your test suite and try to improve it so that you get the maximum number of killed mutants possible. Save the generated PIT report that has a higher number of killed mutants. Are there any mutants that you cannot kill no matter what test cases you write?

(d) If it is not possible to write a test case to kill a mutant, we call it an equivalent mutant (i.e., behaviorally equivalent to the original code). Can you identify any equivalent mutants? If so, why do you think they are equivalent mutant?

Grade Calculation

This assignment is graded out of 100 points. Part 1 has 20 points and part 2 has 70 points. The remaining points are for your report on how correct, well-written, and detailed it is.

For full credit on parts 1 and 2, you are expected to:

- (20pt) Write at least 15 well-formed unit tests with clearly identified expected outputs and descriptive comments
- (30pt) Generate adequate PIT reports for before and after improving your test suite
- (20pt) Accurately describe at least five live mutants and the operators used to produce them.
- (20pt) Accurately describe and possibly identify equivalent mutants

Submission Instructions

Submit your JUnit tests (for before and after modification in part 2), PIT reports (for before and after improving your test suites), and your report (pdf format only) in a .zip folder using Blackboard.