

CS 576 Dependable Software Systems

JUnit Testing Assignment 2

Please submit a **PDF** file via email. The file should be in the form **id_JUnit2.pdf** (in my case it would be **jhk39_JUnit2.pdf**). You should send the email to jayk@drexel.edu with the subject **CS576 JUnit2 id** (again, in my case the subject would be **CS576 JUnit2 jhk39**). If you have any questions, let me know and I will get back to you as soon as possible.

The assignment will be due in 3 parts:

- Part 1: Monday, July 17th, 2006
 - **Part 2: Wednesday July 26th, 2006**
 - Part 3: Monday, July 31st, 2006
-

The objective of this assignment is to utilize a coverage analysis tool to determine the adequacy of the test suites you found for the previous assignment. The goal is to understand how to use and interpret the results of a coverage analyzer, and measures the effectiveness of a test suite. We will be using Clover, a coverage analyzer to evaluate the test suites.

Clover is a Java based Coverage Analysis tool. You can download your own version of Clover from the Clover website. The download version is a 30-day trial version.

Clover can be found at:

<http://www.cenqua.com/clover/>

Please read the users-manual found at:

<http://www.cs.drexel.edu/~jhk39/teaching/cs576/clover-manual.pdf>

If the application you downloaded (from assignment 1) has an ANT build file, you will want to add clover targets to the file. Check out the build file in the tutorial directory of Clover to see an example of how to execute Clover and generate a coverage report. Read the user's manual from more instructions.

Next, evaluate the test suites that you found. Use Clover to instrument your code, and then generate a coverage report.

1. What are the results? Turn in the coverage report generated by the tool.
2. Based on the results, are you confident the program is behaving as expected?
3. Based on the results, determine the parts of the application that are not adequately tested. Write-up a description of why you think those parts are not tested thoroughly.
4. Based on the results, write-up a recommendation describing a plan for testing the untested parts of the application. What would you test first? How would you test? What difficulties do you envision encountering?