EECS 293

Software Craftsmanship

2014 Fall Semester

Programming Assignment 8

Due at your recitation session on October 20- 24

Reading

Read Chapter 22 in the textbook.

Programming

In this assignment, you will design, write, and run test cases for the shortest path methods. You must use JUnit or a similar unit test harness.

The emphasis is on a methodic and coherent process to develop and implement test cases. You should avoid developing test suite in a generic and haphazard manner.

All methods must be tested individually. In particular, all protected and private methods must be tested individually. In other words, you cannot test a private or protected method solely by invoking the public methods that call it. Instead, you need to write test cases that invoke the private or protected method directly.

For each method, the **test cases have to exhaust the typology** described in Section 22.3:

- Structured basis
- Data-flow
- Boundary
- Compound boundaries
- Bad data
- Good data
- At least one stress test

The tests must be **labelled**: add a comment to each test case stating the category(ies) to which the test belongs (for example, a test case could be labeled: "structured basis, data flow, good data", another as "boundary", etc.). Some tests may be based on the examples that you formulated in assignment 6. All test cases must be committed on git.

Your tests must match your code. That is, you must give the true outputs of your program. If your program gives an incorrect result, submit the tests with the incorrect results and/or error messages.

Discussion Guidelines

The class discussion will focus on the **process** to design test cases, and on their completeness and implementation of the test cases.

Grading Guidelines

Starting with Programming Assignment 7, an automatic C (or less) is triggered by improperly named routines. An automatic C (or less) is also triggered by any routine with complexity greater than 4 or by any substantially repeated piece of code.