Assignment 2

CSE 4313 3.0 Software Engineering Testing, Winter 2014

Due: Monday, February 24, 2:30pm.
Format: Individual.

Testing with JUnit

The purpose of this assignment is to give you experience in creating automated test code with JUnit. Your task will be to create a test suite in JUnit for the BORG Calendar, produce bug reports (if any), and submit a written report describing your testing.

To get started

- Create a new directory, say `winter2014`. We will refer to it as the *working directory*
- Download and unzip the BORG Calendar source code (see course webpage for link) in your working directory
- Install the `lombok` package for Eclipse if you are using your personal computer (link on course website), or use `eclipse-lombok` on York University's computers.
- Run Eclipse, select `File -> Import...`. Under General, select Existing Projects into Workspace, click Next. Select the `borg_src` directory to import. This will create the BORGCalendar project in Eclipse.
- Select Properties from the Project menu. Select `Run/Debug settings`. Select Borg, click on `Edit...`. Under the Classpath tab, select BORGCalendar, click on `Advanced...` and `Add Folders`. Select the `res` folder and click OK.
- Launch BORG Calendar from Eclipse by running the Borg class in the `net.sf.borg.control` package.

What to do

You should select an appropriate number of classes and methods for your testing so you can demonstrate at least one example of using boundary testing, equivalence class testing, and decision table testing. (Note: You should not select classes with already developed test cases.) You should define a detailed specification for every selected class based on your understanding of the purpose of the class and then develop a test suite for the selected classes. You may use any testing strategy you believe is appropriate including the ones discussed in class. You are free to create as many test classes as you like.

For each implemented test case, develop a clear description of the test case. You should also provide a justification for the developed test case. Both test case description and test case justification will be a part of your report. If you believe that the implementation fails to meet the specification in some way, create bug reports.

Following is an example of specification for a Java class from a different system:

Specification

The `GraphGenerator` class contains the following two methods:

1. `public GraphGenerator (int numberOfNode, int maxDegree, boolean isConnected, String outputFileName)`

   The first argument is the number of nodes that the generated graph should have. The second argument is the maximum degree for any node.

   If the boolean argument is true, then the produced graph must be connected in the non-directed sense. If the boolean argument is false, then the graph may or may not be connected.

   Finally, the String argument is the name of the file that will contain the produced output in the XML format.

2. `public void createGraph() throws Exception`

   This method generates the random graph and outputs it in the designated file.

   If the number of nodes or the maximum degree is less than 1, then a `NegativeInputException` is thrown.

What to Submit

Before the deadline, submit electronically the test code package you created. To submit, navigate to the directory that contains the working directory (i.e. `winter2014`), and give a command like the following:

```
submit 4313 a2 winter2014
```

You also need to submit a written report. This report must include:

- Specification of the selected Java classes and justification of your selection.
- A description of your application of the three testing strategies, as well as an evaluation of the test cases they derived. If you had to complement the derived test cases with special value testing, describe that as well. The marker will not read your code in order to see what you tested. You have to describe it.
- Show one detailed example of each testing technique that was discussed in class
- The bug reports you created (if any).

To submit your written report, give:

```
submit 4313 a2 a2.pdf
```

where `a2.pdf` is your report.

Presenting your thought processes in English is an important skill for a software professional. If you have trouble writing English, have somebody proof-read and correct your prose. You might find the services offered by The English as a Second Language Open Learning Centre useful: `http://www.yorku.ca/eslolc`