

Assignment 2

White box testing

Model-Based Testing
DIT848/GU and TDA260/Chalmers

April, 2013

1 Introduction

Code coverage is used as a metric to measure quality of tests. The purpose of this assignment is to become familiar with code coverage analysis, which is one of the most common white box testing methods. Using this method you can find parts of the code that are not covered by the test cases, guiding you to come with new test cases to increase coverage and thus the quality of your tests. Before starting this assignment make sure EcEmma (Java code coverage tool) is installed in your machine.

2 Submitting your work

If you want to have feedback on your assignment, check with Hamid Ebadi (hamide@chalmers.se). If you want to submit, please attach a .zip or .tar.gz archive, containing your source code and .txt, .pdf or .doc file describing your answers. For subject of your email and also to name your archive file, please use the assignment number and your (last) name as in the following example: `assignment02_names.zip`.

The deadline for this assignment is **Wednesday, 17 April 2013**.

3 Structure

The file `calculator2.zip` contains the following files:

- `src/Calculator2.java`: Interface and implementation for a modification of module implementing the functionality of a simple calculator. This version fixes problems with the first version and also can evaluate expressions that contain parentheses, e.g. `55-(2*(3+7))`.
- `src/Calculator2Test.java`: Example showing three sample JUnit test cases for the calculator class.

4 What to do

The file `calculator2.zip` contains a modification of `calculator.zip` distributed last week. Run the existing test suite using EcEmma. Create more test cases to increase code coverage as much as possible. Does your test suite have full statement and branch coverage? If not, add test cases to get full coverage, or motivate that it is not possible.