ETSA02-ADM-LAB4

Lab 3

Code coverage testing and static code anaysis

Version 0.2 approved

Prepared by Markus Borg
Dept. of Computer Science, Lund University

April 21, 2018

Revision History

Name	Date	Reason For Changes	Version
Markus Borg	2018-04-18	Initial structure.	0.1
Markus Borg	2018-04-21	Added eclEmma and SpotBugs.	0.2

1 Introduction

During Lab 4 you will continue working with automated testing of Basic Melee Bot. More specifically, Lab 4 covers working with two open source tools:

- Code coverage testing with eclEmma
- Static code analysis with SpotBugs

Furthermore, you will generate javadoc.

2 Before the lab

The source code required for Lab 4 is available on GitHub:
https://github.com/lunduniversity/introsofteng

If you have already cloned the repository, pull the latest source code to make sure you work with the latest version. If you prefer downloading the code, instead click the button presented in Figure **??** and choose "Download ZIP". Once downloaded, locate the files you need for Lab 4. The files are in the folder introsofteng-master/labs/lab3/src, and its subfolder: 'test'. Rewatch the video "Lab2_download.avi" on Google Drive (ETSA02 Everyone/Labs) if you need support.

In Lab 4, you will learn to download tools from the big Eclipse ecosystem through Eclipse Marketplace. You reach Eclipse Marketplace directly into from your Eclipse installation.

2.1 Install and run test cases with JUnit and a code coverage tool

2.2 The fundamentals of code coverage tool

https://en.wikipedia.org/wiki/Code_coverage
http://www.eclemma.org/jacoco/trunk/doc/counters.html

2.3 The fundamentals of static code analysis tool

3 At the lab

eclEmma introduces a new run option in Eclipse, beyond “Run as...” and “Debug as...”. The new run option is called “Coverage as...”. The eclEmma “Coverage view” automatically appears when a new coverage session is added or can manually opened from the Window → Show View menu in the Java category. It shows code coverage summaries for the active session.

The Coverage view shows all analyzed Java elements. Individual columns contain the following numbers for the active session, always summarizing the child elements of the respective Java element:

- Coverage ratio
- Items covered
- Items not covered
- Total items

The elements may be sorted in ascending or descending order by clicking the respective column header. Double-clicking an element opens its declaration in an editor with highlighted source code. You can select between different metrics, see last section for details.

4 After the lab