CHALMERS
UNIVERSITY OF TECHNOLOGY

UNIVERSITY OF GOTHENBURG

Exercise 4: Structural Testing

Gregory Gay
DIT635 - February 21, 2020

Finish In-Class Activities First!

The Planning System Returns

- Code on Canvas:
  - https://canvas.gu.se/courses/25762/files/folder/Misc?preview=2280199
- Everybody likes meetings.
  - Not true - but we need to book them.
- We don’t want to double-book rooms or employees for meetings.
- System to manage schedules and meetings.

Structural Testing

- You already tested this system based on the functionality. Now we want to fill in the gaps.
- Goal: 100% Statement Coverage (Line Coverage) of all classes **except** Main.
  - First, measure coverage of your existing tests
  - Then, fill in any gaps with additional tests targeting the missed code.
  - If code cannot be covered, identify why.
  - **If you finish early, also do this for the CoffeeMaker**

Measuring Coverage

- The easiest way: use an IDE plug-in.
  - Eclipse: EclEmma - https://www.eclemma.org/
  - IntelliJ: IntelliJ IDEA code coverage runner: https://www.jetbrains.com/help/idea/code-coverage.html
- Command line:
  - Emma, Cobertura offer executable tools.
  - JaCoCo available as a Maven plug-in: https://medium.com/capital-one-tech/improve-java-code-with-unit-tests-and-jacoco-b342643736ed

UNIVERSITY OF
GOTHENBURG

CHALMERS
UNIVERSITY OF TECHNOLOGY