SENG 301

Sydney Pratte
sapratte@ucalgary.ca
TR 12-12:50 & 1-1:50

Test Coverage

- Test coverage refers to how much of the software has been run in test code.
- 100% test coverage is very unlikely - need to test strategically
    - Know the problems you want to look for (purpose)

Categories

- Categories of test selection:
  - Black-box testing - black-box analysis used to ensure conformance to interface
  - White-box testing - white-box analysis used to ensure that the code is more fully exercised

Techniques

- Black-box techniques:
  - Equivalence-based selection
  - Boundary-based selection
- White-box techniques:
  - Path-based selection
  - Branch-based selection
- Combination techniques:
  - State-based selection
  - Polymorphism-based selection

Lab 8: Test Coverage

In this lab, you will be seeking to unit test the AbstractHardware class in the hardware simulator for the vending machine (again, you are using simulator-1.0.jar).

Note that Object, Class, Method, String, and Vector are all classes in the Java Standard Library, so it is reasonable to treat them as bug-free (i.e., your test code is more likely to have bugs than these classes). SimulationException is simple so it is not worth trying to stub out.