PROGRAMS THAT TEST THEMSELVES!

Formal Methods in Software Development
Shohreh Takyar
Fall 2009
Testing
Testing and Software Development

I wish programs could test their functionalities themselves...

Is it possible? Let’s see...
Outline

• Basic definitions
  • Automated test, Design by contracts, EiffelStudio

• Automated testing in industry (current state)
  • Test execution, Regression testing, Resilience

• Towards fully automated testing
  • Test preparation, Test oracles, Test minimization

• Summary and future directions
Auto Test

• Definition
  • A set of software tools/programs designed and implemented to test their functionalities themselves

• Components
  • Test generation
  • Test extraction
  • Integration of manual tests
Auto Test (continued)

• Test generation
  • Create and run test cases automatically

• Test extraction
  • Contributes to the set of test cases for future regression testing

• Integration of manual tests
  • The final goal of testing is to reveal all possible scenarios that can make a software fail
Design by Contracts

- Contract components
  - Pre conditions
  - Post conditions
  - Class invariants
    - A set of properties that will be preserved during the routine call
Auto test…
Programming by contracts…
Let’s release an auto test Framework…
We can even name it “EiffelStudio” …

I’m sorry! It has already been done!
Testing Practices in EiffelStudio

• EiffelStudio presents an integrated version of three components of an auto test
  • Test synthesis (Generation)
  • Test extraction takes place anytime the underlying process is paused in EiffelStudio debugger
  • Manual tests are considered

(http://eiffelstudio.origo.ethz.ch)
Automated Testing in Industry (current state)

• Test execution
  • Junit and its positive impact on automated testing

• Regression testing
  • To determine if the new changes have any negative impact on any of the working features of the code

• Resilience (stability)
  • To ensure that the system after each failure is able to recover automatically

Interesting to be automated…
Fully Automated Software Testing

- Test preparation
  - Test generation
  - Test extraction
  - Integration of manual test

- Test oracles
  - A mechanism that determines whether a test has been passed or failed

- Test minimization
  - A way to remove the redundant scenarios without missing any test case
Test Preparation

• Preparing test data in OOP
  • Generate instances of classes (objects pool)
  • Select some of theses objects
    • Picking up from the objects pool
    • Generating new instance (preferred strategy)
  • Select arguments for the features to be called
    • Random testing
    • Adaptive random testing (not preferred yet)
Test Oracles

• Contracts are in the form of boolean expressions. take advantage!

• Quality of the test oracles relies on the quality of the contracts
Test Minimization

• **Be careful!** We should not lose/ignore any of failures as a result of minimization

• **Minimization strategies**
  
  • Static program slicing (relies on dependencies)
  
  • **Delta debugging** (verifies causality)

  - The subset of the code that possibly has an impact on the oracle outcome
  - Following back data dependencies
  - Failing test case
  - Repeats the test with subsets of these factors
  - A set of factors that might influence a test outcome
Example
Summary

Current automated testing
- Test execution
- Regression testing
- Resilience

Ongoing process
- Test preparation
- Test oracles
- Test minimization