CSC 440/540: Software Engineering

Test Driven Development and Refactoring
Topics

1. Bugs
2. Software Testing
3. Test Driven Development
4. Refactoring
5. Automating Acceptance Tests
Bugs

Photo # NH 96566-KN  First Computer "Bug", 1945

0800  Started
1000  stopped - started

13th May 0319  MP-MC (1930)
033  PRO

Relays 6-2 in 033 failed speed test
In relay

1100  Started Cosine Tape (Sine check)

1525  Started Multiplier Test

1545  Relay #70 Panel F

First actual case of bug being found.

1830  Moth found.

1700  closed down.
Ariane 5 Flight 501 Bug

Ariane 5 spacecraft self-destructed

- June 4, 1996
- Due to overflow in conversion from a floating point to a signed integer.
- Spacecraft cost $1 billion to build.
Software Testing

Software testing is the process of evaluating software to find defects and assess its quality.
Test Granularity

1. **Unit Tests**
   Test specific section of code, typically a single function.

2. **Component Tests**
   Test interface of component with other components.

3. **System Tests**
   - End-to-end test of working system.
   - Also known as Acceptance Tests.
Regression Testing

Regression testing focuses on finding defects after a major code change has occurred. Regressions are defects such as:

- Reappearance of a bug that was previously fixed.
- Features that no longer work correctly.
How to find test inputs

Random inputs
  - Also known as fuzz testing.

Boundary values
  - Test boundary conditions: smallest input, biggest, etc.
  - Errors are likely to occur around boundaries.

Equivalence classes
  - Divide input space into classes that should be handled in the same way by system.
How to determine if test is ok?

Diagram:

1. Test executed
2. Verifications
   - Expected to fail
     - No: Pass (normal)
     - Yes: Pass (defect fixed)
   - Expected to fail
     - No: Fail (new)
     - Yes: Fail (expected)
3. Check against expected outcome
4. Check against expected failed outcome
Test Driven Development

1. (Re)Write a test
2. Check if the test fails
3. Write production code
4. Run all tests
5. Repeat if tests fail
6. Clean up code

All tests succeed
Advantages of writing tests first

- Units tests are actually written.
- Programmer satisfaction.
- Clarification of detailed interface and behavior.
- Provable, repeatable, automated verification.
- Confidence to change code.
Refactoring

Refactoring is a structured technique for changing the design and implementation of code without changing its functionality.

Goals of refactoring

- Reduce complexity
- Improve code readability
- Improve maintainability
- Improve flexibility
Refactoring Techniques

Techniques for breaking code apart into logical pieces

- **Extract Class** moves part of the code from an existing class into a new class.
- **Extract Method** turns part of a larger method into a new method, improving method cohesion.

Techniques for improving names and location of code

- **Move Method** or **Move Attribute** – move to a more appropriate class or source file.
- **Rename Method** or **Rename Attribute** – change name to a new one that better reveals purpose.
- **Pull Up** – move method or attribute to a superclass
- **Push Down** – move method or attribute to a subclass
Refactoring Techniques

Techniques that allow for more abstraction

- **Encapsulate Field** – force code to access the field with getter and setter methods, reducing coupling.
- **Generalize Type** – create more general types to allow for more code sharing
- Replace conditional with polymorphism.

There are many other refactorings.

- Or Martin Fowler’s book *Refactoring*
Acceptance Test Frameworks

Cucumber (Ruby)
- Uses declarative language to describe tests.
- Similar tools: Behat (PHP), Lettuce (Python), JBehave

Fit (Framework for Integrated Test)
- Customers provide examples formatted as tables
- Fit checks document and fills in table red/green.

Selenium (Selenese DSL)
- Software testing framework for web applications.
- Can use record/playback tool without scripting.
- Ports: C#, Java, Perl, PHP, Python, Ruby
Web Automation

HtmlUnit
- Java based headless browser with JS.

Mechanize
- Automates HTTP form interactions, but not JS.
- Available for Ruby, Python, and Perl.

PhantomJS
- Headless, scriptable WebKit browser with JS, HTML5
- CasperJS same interface with Firefox browser.
- TrifleJS same interface with IE browser.

ScriptableBrowser
- Headless browser that’s part of PHP SimpleTest.

Spynner
- Headless WebKit browser with AJAX support for Python.
References