

## Objectives

- Software Quality Metrics
  - Metrics plugin

Oct 31, 2011

Sprenkle - CSCI209

1

## Review

- What property are we designing to?
- What is the typical fix for designing more flexible/maintainable code?
- What are a few examples of code smells?
  - How do we address those code smells?

Oct 31, 2011

Sprenkle - CSCI209

2

## Notes on Assignment 9

- Goals
  - Learn to read, understand someone else's code
    - Refactoring can help
  - Refactor for readability
  - Justify decisions
- No "right" answer
  - Many design decisions
  - Want you to defend your design decision in code critique

Oct 31, 2011

Sprenkle - CSCI209

3

## Evaluating a Test Suite

- How do we know a test suite is *good*? What metrics can we use?
- How do you think I will evaluate your test suites?

Oct 31, 2011

Sprenkle - CSCI209

4

## Testing Project

- If a test case passes, what does that mean?
- If a test case fails, what does that mean?

Oct 31, 2011

Sprenkle - CSCI209

5

## Testing Project

- If a test case passes, what does that mean?
  - Code is correct (positive)
  - Code has a bug that test case doesn't reveal (false negative)
- If a test case fails, what does that mean?
  - Code has a bug (negative)
    - Or doesn't follow your specification
  - Code is correct but test case is incorrect (false positive)
- Are there any other possibilities?

Oct 31, 2011

Sprenkle - CSCI209

6

## Grading Testing Project

		Test Case Result	
		Pass	Fail
Application	Correct	True Positive	False Negative
	Faulty	False Positive	True Negative

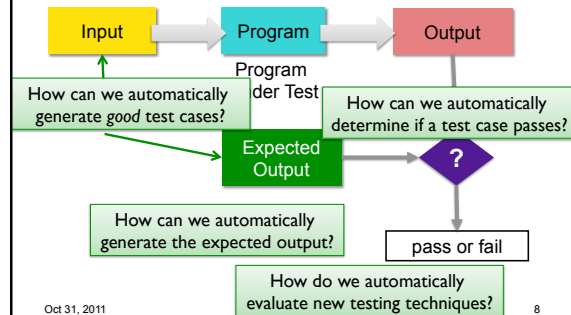
- Which is worse: false positives or false negatives?

Oct 31, 2011

Sprenkle - CSCI209

7

## Testing Research Challenges



Oct 31, 2011

8

## SOFTWARE QUALITY METRICS

Oct 31, 2011

Sprenkle - CSCI209

9

## Metrics to Measure Software Quality

- Create metrics to help us figure out if our code is good and what we can improve
  - Add a little more science
- Examples: number of methods, # loc / method, # attributes/class
- Tricky: Not clear what is "good" number
  - Requires good judgment, experience
  - Metrics often should not be considered in isolation

Oct 31, 2011

Sprenkle - CSCI209

10

## Example Metrics

Metric	Description
<b>Afferent Coupling (Ca)</b>	Number of classes outside package that depend upon classes within package
<b>Efferent Coupling (Ce)</b>	Number of classes inside package that depend on classes outside package
<b>Instability (I)</b>	$Ce / (Ca + Ce) \rightarrow \text{range } [0,1]$ Indicates resilience to change
<b>Abstractness (A)</b>	Number of abstract classes divided by total number of classes in a package. 0 → concrete, 1 → abstract

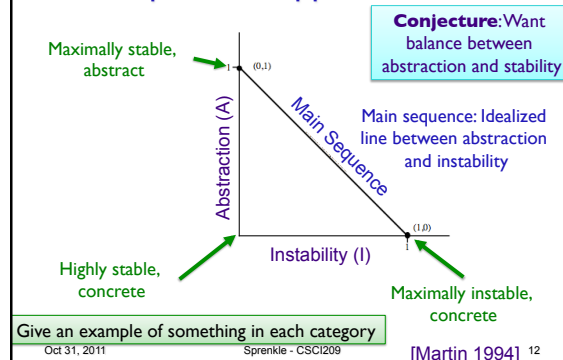
Instability: How does this metric measure instability?  
What does a 0 or 1 mean?

Oct 31, 2011

Sprenkle - CSCI209

11

## Main Sequence: Supports OCP



Oct 31, 2011

Sprenkle - CSCI209

[Martin 1994] 12

### Example: Lack of Cohesion of Methods (LCOM)

- A measure of a class's *cohesiveness*
- Calculated with the Henderson-Sellers method:
  - $m(A)$  : # of methods accessing an attribute A
  - Calculate the average of  $m(A)$  for all attributes, subtract the number of methods  $m$  and divide the result by  $(1-m)$

$$\frac{\overline{m(A)} - \# \text{ of methods}}{1 - \# \text{ of methods}}$$

Oct 31, 2011

Sprenkle - CSCI209

13

### Analysis and Discussion:

#### What does LCOM tell us?

$$\text{LCOM} = \frac{\overline{m(A)} - \# \text{ of methods}}{1 - \# \text{ of methods}} \quad \text{where } m(A) \text{ is \# methods accessing attribute A}$$

- What is the relationship between  $m(A)$  and # of methods?
- What are the extremes?
  - Every method accesses every attribute?
  - Every attribute is accessed by one method?

Oct 31, 2011

Sprenkle - CSCI209

14

### Example: Lack of Cohesion of Methods (LCOM)

- A measure of a class's *cohesiveness*
- Calculated with the Henderson-Sellers method:
  - $m(A)$ : # of methods accessing an attribute A
  - calculate the average of  $m(A)$  for all attributes, subtract the number of methods  $m$  and divide the result by  $(1-m)$
- Low value → a cohesive class
- Value close to 1 → a lack of cohesion
  - Suggests class might better be split into a number of (child) classes

Oct 31, 2011

Sprenkle - CSCI209

15

### Metrics Plugin

- Install plugin: Help menu -> Software Updates -> Find and Install
  - New Remote Site
    - Name: Metrics
    - URL: <http://metrics.sourceforge.net/update>

Oct 31, 2011

Sprenkle - CSCI209

16

### Metrics Plugin

- Provides information about your classes
  - # of classes
  - # of lines of code per method
  - # of attributes
  - Coupling (afferent, efferent)
  - Instability
  - ...
- <http://metrics.sourceforge.net>

Oct 31, 2011

Sprenkle - CSCI209

17

### PMD Reports

- Java source code analyzer
- Looks for possible bugs, poor coding practices
  - Duplicate code
  - Dead code
  - Empty if/while/catch blocks
  - Suboptimal code (e.g., Strings, StringBuffers)
- Eclipse Plugin:
  - Update site: <http://pmd.sourceforge.net/eclipse>

Oct 31, 2011

Sprenkle - CSCI209

See Also: FindBugs

18

## To Do

- Assignment 9: Due Wednesday
- Friday, November 4: Extra Credit Opportunity
  - [Turing Award in CS \(10 pts\)](#)
    - Commons – Women's Resource Room, 12:30 p.m.
    - Professor Stough to present on prize awarded to Leslie G. Valiant