Final Exam Review

CS 3398
Fall 2013
Jill Seaman

Final Exam

- Wednesday, December 11, 8:00AM to 10:30AM
- Closed book, closed notes, clean desk
- Lectures:
  - Models, Detailed Design, Implementation,
  - Chapter 8, Chapter 9
- 25% of your final grade
- I recommend using a pencil (and eraser)
- I will bring extra paper.

Exam Format

- Multiple choice: 17 questions
- Problems: 3
  - draw a UML class diagram (or one of the other 3)
  - given a class diagram, decompose it into subsystems (loose coupling, high cohesion)
  - derive some test cases
- Written answers: 3
  - 3 to 5 sentences, generally
  - Define, explain, compare, evaluate
  - Make claims and give support
- Each question will indicate how many points it is worth (out of 100)

System modeling

- Simple Context Model (SRS section 2.1)
- UML Models:
  - class diagram (SRS section 3.4)
  - state diagram
- Control Flow Diagrams (aka Flowcharts)

- Be able to
  - Recognize the models
  - Draw simple versions of the models
Detailed Design

- Design Processes
  - Functional Decomposition (top down design)
  - Relational Database Design
    - tables, foreign keys (ER Diagrams)
  - Object-oriented design and UML
    - using class diagrams, state diagrams, etc.
- 5 steps:
  1. Requirements elicitation
  2. Object oriented analysis
  3. System Design
  4. Object Design
  5. Implementation

Design characteristics and metrics

- Desired Characteristics:
  - Consistency
  - Completeness
  - Simplicity
- Legacy Characteristics:
  - Halstead Complexity
  - McCabe’s Cyclomatic Complexity
- Measuring simplicity:
  - Loose coupling
  - Strong cohesion

Implementation

- Desired characteristics, and how to achieve them:
  - Readability and maintainability
    - Programming style and coding guidelines
    - Using comments well
    - Refactoring
  - Correctness
    - Testing and debugging
  - Performance
    - Optimization
- Other issues:
  - Configuration management: why version control?
  - Open source development: pros/cons, licensing issues

Ch 8: Software Testing

- Concepts:
  - Verification and Validation
  - static vs dynamic verification
  - Failure, Fault, Test cases, Testing
  - white box vs black box testing
  - Test stubs and drivers
- Testing process
  - Development Testing
    - Unit
    - Component
    - System
  - Release Testing
  - User Testing
    - Alpha
    - Beta
    - Acceptance
Ch 8: Software Testing

Deriving test cases:

- **Unit Testing**
  - Partition testing (Equivalence Class Partitioning)
  - Boundary value analysis
  - Path testing (Path Analysis)
  - State-based testing
  - Guideline-based testing

- **System + Release Testing**
  - Use case-based testing
  - Scenario testing
  - Requirements-based testing

Ch 9: Software evolution

- **Evolution Process**
  - Spiral model: maintenance = iterative development
  - Change requests, Impact analysis, Release Planning, Change Implementation
  - Program understanding
  - Handling urgent change requests

- **3 Types of software maintenance**
  - Defect fixing, adapting to new environment, new features

- **Reengineering**
  - What, when, why (why not start from scratch?)
  - Techniques

- **Refactoring**
  - What, when, why
  - Bad smells

Example Problems

- Assignment 5 and Assignment 6
- See the Sample Final exam on the website
- Note: read multiple choice questions carefully.

Office Hours

Finals Week

<table>
<thead>
<tr>
<th>Day</th>
<th>Date</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>M</td>
<td>12/9</td>
<td>12:30-1:30pm</td>
</tr>
<tr>
<td>T</td>
<td>12/10</td>
<td>9:30-10:30am</td>
</tr>
<tr>
<td>W</td>
<td>12/11</td>
<td>11:00am-12:00noon</td>
</tr>
<tr>
<td>Th</td>
<td>12/12</td>
<td>1:30-3:30pm</td>
</tr>
</tbody>
</table>

and by appt.