Final Exam Review

CS 3398
Spring 2014
Jill Seaman

Final Exam

Section 251(MW): Wed, May 7, 2:00PM to 4:30PM
Section 252(TR): Tues, May 6, 11:00AM to 1:30PM
Closed book, closed notes, clean desk
Lectures:
- System Models, Detailed Design, Implementation,
- Testing, Evolution
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 state diagram, or control flow diagram)
  - 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
  - Requires memorization of topics and issues
- 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 (goals & activities):
  1. Requirements elicitation
  2. Object-oriented analysis
  3. System Design
  4. Object Design
  5. Implementation

Design characteristics and metrics

- Legacy Characteristics:
  - Halstead Complexity
  - McCabe’s Cyclomatic Complexity
- Measuring simplicity:
  - Loose coupling
  - Strong cohesion
- OO Design guideline:
  - Law of Demeter

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 over from scratch?)
  - Techniques

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

Final advice

- See Assignment 5 and Assignment 6 solutions on TRACS
- See the Sample Final exam on the website
- Note: read multiple choice questions carefully.
- See “How to Study for CS3398” on the website.

Office Hours

Finals Week

<table>
<thead>
<tr>
<th>Day</th>
<th>Date</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Th</td>
<td>5/1</td>
<td>2:30-4:30pm</td>
</tr>
<tr>
<td>M</td>
<td>5/5</td>
<td>12:00-1:00pm</td>
</tr>
<tr>
<td>T</td>
<td>5/6</td>
<td>2:30-3:30pm</td>
</tr>
<tr>
<td>W</td>
<td>5/7</td>
<td>11:00-12:00noon</td>
</tr>
</tbody>
</table>

and by appt.