Exam Format

- 39 multiple choice
- 2 short design problems
- feedback

Bloom’s revised taxonomy

- Create
- Evaluate
- Analyze
- Apply
- Understand
- Remember

---

Study Hint: breadth-first
<table>
<thead>
<tr>
<th>Topic</th>
<th>Source</th>
<th>#?</th>
</tr>
</thead>
<tbody>
<tr>
<td>UML Class Diagrams</td>
<td>Previous courses, web, etc</td>
<td>5</td>
</tr>
<tr>
<td>Design Patterns</td>
<td>Gamma et al. slides</td>
<td>5+2</td>
</tr>
<tr>
<td>Architectural Styles</td>
<td>Garlan &amp; Shaw slides</td>
<td>6</td>
</tr>
<tr>
<td>Architecture Types</td>
<td>Hassan + Bowman slides</td>
<td>3</td>
</tr>
<tr>
<td>4+1 Views</td>
<td>Kruchten</td>
<td>4</td>
</tr>
<tr>
<td>Web/Enterprise</td>
<td>slides</td>
<td>11</td>
</tr>
<tr>
<td>Creativity</td>
<td>slides</td>
<td>5</td>
</tr>
</tbody>
</table>
Topics Not on Midterm (may be on final)

- Guest Presentation
  - Ian Davis (LSEdit)
- Design Processes
  - normal vs radical design
  - agile, RUP
- Case Studies
  - Linux, web servers, seL4 microkernel, KWIC, styles/compilers, styles/oscilloscopes
- Invariants, properties, verification, etc.
  - Alloy + Spin
Architectural Styles

- Styles
  - pipe & filter
  - data abstraction
  - implicit invocation
  - layered
  - repository
  - interpreter style

- For each
  - know the intent, advantages, disadvantages, variations, applications
Design Pattern

- Patterns
  - Singleton, Adapter, Bridge, Façade, Command, Iterator, Observer, Strategy, Composite, Visitor, Interpreter
- For each
  - intent, class diagram
  - applicability
    - given a scenario, which and why
  - discussion in the class
Types of Architectures

• Reference, Conceptual, Concrete
  • definition & differences
    – The role of each in the software design process
• drift / erosion
  – definition
Architectural Views

- 4+1 View
  - Kruchten
- For each view
  - definition
  - diagram
  - purpose
  - usage
    - what views are required and what views are not and when
Enterprise Architecture

• Types of Architectural Styles

• Evolution
  • know the differences between
    - first, second, & third (gwt) generation
    - improvement & weaknesses
  • draw the diagrams
    - identify components & connectors clearly
    - request-response cycle
Enterprise Architecture

- Synchronous & asynchronous
- Separation of concerns
  - responsibility of each tier
- Functional vs. non-functional requirements
  - know the difference
  - which ones are 
    - honoured (& why)
    - violated (& why)