CS414 Final Exam Study Guide

The final exam will have a format similar to Test 1 and Test 2:

**Part 1:** Multiple-choice questions
**Part 2:** Short answer questions (e.g., “What is run-to-completion semantics for state machine models?”)
**Part 3:** Modeling Questions: In this section you may be required to draw UML diagrams.

Students should be able to answer the following questions by the date of Final Exam:

### SE and Modeling Fundamentals
1. What are the important software principles supported by MDD?
2. What is the difference between a model and a diagram?

### Activity Modeling
1. Know the purpose of constructs used to build activity models.
2. How can activity models be used at the system engineering level?
3. How can activity models be used at the software requirements level?

### UML Requirements Class Models
1. Why are operations not shown in requirements class models?
2. Why is navigation not shown in requirements class models?
3. Given a Class Model you should be able to answer questions about the relationships that are described.

### OO Design (Design class models and sequence diagrams)
1. What aspect of behavior is described by a sequence diagram?
2. What is the difference between a synchronous and asynchronous message?
3. What do the navigation symbols in a class diagram represent?
4. What is the difference between a design class diagram and a requirements class diagram?
5. How do you describe alternative and loop structures in a sequence diagram?

### Design Patterns
1. Know the details of the following behavioral patterns: Command, State, Strategy, Template, Visitor, Observer. Know how to describe these patterns using class diagrams and sequence diagrams.
2. What specific problems are targeted by a particular design pattern?
3. What role does a particular class play in a pattern (e.g., What role does the Invoker play in the Command pattern)?

### OCL
1. How do you express invariants?
2. How do you express new queries?
3. How do you express query operations shown in a class diagram?
4. How do you specify operations in OCL?
5. Have knowledge of at least the following OCL operations: collect(), select(), forAll(), exists(), isEmpty(), notEmpty()

**Architectural and Detailed Design Modeling**

1. What is the difference between a deployment architecture and a logical architecture?
2. What is a provided interface? What is a required interface?
3. What is the difference between import and access in a basic architecture?
4. What is a component-based architecture?
5. How are subsystems represented in a static architecture model?
6. What is a layered architecture?
7. What is the difference between an architectural model and a detailed design model?
8. How does one ensure that components in a component architectural model can be swapped (i.e., how can they be designed to be “swappable”)?

**State Machine Models**

1. What is the run-to-completion semantics?
2. What is a state/transition/event/activity/guarded condition?
3. What are the different event types?
4. What is the difference between a passive object and an active object?
5. What are the different types of activities that can be specified in a statemachine model?
6. What is the difference between static and dynamic conditional branching?
7. What is a hierarchical state machine and why is it useful?
8. What is a completion transition?
9. What is a deferred event?
10. What do orthogonal regions represent?
11. How do you model interactions across orthogonal regions?
12. How do you model forks and joins in a statemachine model?
13. What is a group transition and what purpose does it serve?
14. Given a statemachine model you should be able to describe what happens when an event occurs when the system is in a particular state (i.e., you should be able to determine the state transitions that occur as a result of the event occurrence).