**SER 315 Software Engineering Review Questions and Answers**

**Week 1: Engineering Design Process and Class Diagrams**

**Q1: True or False: Design problems have one solution**

**A:** False

**Q2: What are the advantages of a formal design process?**

**A:**

1. It makes design decisions explicit.
2. Allows for greater documentation.
3. Reduces the likelihood that important issues are forgotten or overlooked.

**Q3: What are Class Diagrams?**

**A:** They are structural models that have varying levels of abstraction. Each level aims to refine the architecture of the system in greater detail.

**Q4: What are the different types of Design Patterns?**

**A:**

* Creational – the creation of objects.
* Structural – the composition of classes.
* Behavioral – how classes or objects interact and distribute responsibility.

**Week 2: Use Case and Sequence Diagrams**

**Q5: What kind of diagram is a Use Case Diagram?**

**A:** Use Case Diagrams are behavior diagrams. They model the functionality of the system from the perspective of the user.

**Q6: What kind of model is a Sequence Diagram?**

**A:** Sequence Diagrams are Interaction models. Sequence Diagrams represent the interaction that occurs via messaging between the system and its environment (outside systems, users, etc.).

**Q7: What are the main components of Use Case Diagrams?**

**A:** Main components consist of use cases, actors, scenarios, and associations.

**Week 3: Activity Diagrams, UI Design, and System Class Models**

**Q8: What are Activity Diagrams?**

**A:** Activity Diagrams are behavioral models that refine the activities that must be completed and their general flow.

**Q9: What are the 3 layers of the System Class Model?**

**A:**

* Data Layers or Entity layer/class – holds the system's data
* Application Layer or Control layer/class – encapsulates the system's business logic
* Presentation Layer or Boundary layer/class – represents the outward facing interface to the user.

**Q10: What elements are in an Activity Diagram?**

**A:**

* Actions are the steps of the workflow.
* Transitional arrows indicate the next step.
* Splits and merges are where decisions are made about the next step.
* Forks and joins indicate concurrency.
* Swim lanes show the division of responsibilities between actors.
* Objects can be shown as squares in UML.

**Week 4: Formalization and Z-Notation**

**Q11: What is Design by Contract?**

**A:** Design by contract states that if the preconditions are met then this component should fulfill the post conditions.

**Q12: Why are formal methods vital in safety critical systems?**

**A:** Formal methods provide precision and allow for verification of correctness in systems where failures could be catastrophic.

**Q13: What is Z-notation used for?**

**A:** Z-notation is a formalization language used in Operational patterns.

**Q14: What do the following Z Notation operators mean?**

**A:**

* ∣ is the cardinality operator - number of elements in a set
* ∈ is the "element of" operator.
* ∪ is the union operator.
* ∩ is the intersection operator

**Q15: What is the purpose of formal specification in Software Engineering?**

**A:** Formal specification removes all ambiguity that may exist in natural language descriptions. It allows for the developer to verify correctness, consistency, and completeness.

**Week 5: Operational Patterns**

**Q16: Which parts of documentation are used for Operational Pattern components?**

**A:**

* Description – requirements
* Input – Sequence Diagrams and System Class Model
* Reads/Changes – System Class Model
* Sends – Sequence Diagrams
* Pre-conditions – assumptions gathered from the requirements
* Post-conditions – assumptions about the state after the operation, from System Class Model and Sequence Diagrams.

**Q17: What does the implicit keyword mean?**

**A:** Implicit implies that the existence or uniqueness of an object from the reads or changes is expected.

**Q18: What does the post-condition tell us?**

**A:** How the system state changes after the operation.

**Week 6: Communication Diagrams**

**Q19: What kind of model is a Communication diagram?**

**A:** A communication diagram is an interaction model. It shows how the system communicates through messages to perform a specific behavior or operation.

**Q20: What keyword does a Communication diagram use to create an instance of an object?**

**A:** O = create(parameters)

**Q21: What are Communication diagrams used for?**

**A:** Communication diagrams are interaction models that describe the communication (primarily function calls) of objects within the context of a feature/scenario.

**Week 7: Design Patterns and Architectural Styles**

**Q22: What are the constraints to a layered system?**

**A:**

* Layers cannot be skipped.
* Each layer must have its own distinct responsibilities.
* Layers should only be dependent on the ones below it.
* Processing should occur from top to bottom.

**Q23: What Design pattern would you use if you had a resource and wanted to ensure only one instance of it existed?**

**A:** The Singleton pattern

**Q24: What is an anti-pattern?**

**A:** An anti-pattern is a programming mistake that gets repeated.

**Q25: What are some design patterns to focus on?**

**A:** Composite, builder, strategy, factory, adapter, decorator, observer, singleton, and facade.

**Q26: What are architectural styles to understand?**

**A:** Layered systems, blackboard, pipe and filter, and microkernel.