Object-Oriented Methodology HW 01

2024 Fall Semester

21 CST H3Art

Short answer questions

What is the difference between OOA and OOD?

Object-Oriented Analysis (OOA) and Object-Oriented Design (OOD) are two distinct stages in the software development process, each serving a different purpose. OOA is primarily concerned with understanding and modeling the problem domain by identifying the system's functional requirements. During this stage, the focus is on analyzing the problem and representing it through objects that reflect real-world entities or concepts. The goal is to create a clear and accurate model of what the system needs to do, without considering how it will be implemented.

In contrast, OOD takes the analysis model developed in OOA and transitions it into a blueprint for building the system. This stage is focused on how the system will be constructed, including the architecture, class design, and the interactions between objects. OOD addresses the technical aspects, such as how objects will interact through methods, the use of inheritance, and the implementation of encapsulation and polymorphism. Essentially, while OOA is about understanding "what" the system should do, OOD is about determining "how" it will do it.

What is the difference between the incremental and iterative approach to software development?

The incremental and iterative approaches are two methodologies used in software development, each with a unique focus. The incremental approach involves dividing the system into smaller, manageable parts, or increments, which are developed and delivered in sequence. With each increment, new features are added to the system, allowing for early and continuous delivery of working software. This method is particularly useful when there is a need to deliver a functional product quickly, as it enables developers to release portions of the system that are fully functional while continuing to develop additional features.

On the other hand, the iterative approach focuses on refining and evolving the entire system through repeated cycles, or iterations. Each iteration involves revisiting and improving various aspects of the system, such as analysis, design, coding, and testing. This approach allows for ongoing feedback and adjustments, making it possible to address issues and refine requirements as the project

progresses. The key difference between the two approaches lies in their focus: the incremental approach emphasizes delivering parts of the system in stages, while the iterative approach emphasizes refining the system as a whole through continuous cycles of improvement.

What is the Relationship Between the Disciplines and Phases?

In software development, the concepts of disciplines and phases are closely intertwined, guiding the organization and execution of a project. Disciplines refer to the various activities involved in the development process, such as requirements gathering, analysis, design, implementation, testing, and deployment. These disciplines are not confined to a single stage but recur throughout the development lifecycle.

Phases, on the other hand, are the time periods during which specific aspects of the project are the focus. For example, during the inception phase, the primary activities might involve requirements gathering and initial analysis. As the project moves into the elaboration phase, the focus may shift towards more detailed design and architecture. In the construction phase, implementation and testing take center stage, and finally, the transition phase may focus on deployment and user training.

The relationship between disciplines and phases is cyclical and interdependent. As the project progresses through its phases, different disciplines are emphasized based on the specific needs of that phase. This structure ensures that all necessary activities are carried out in a systematic way, allowing the project to evolve from initial concept to final deployment in a well-organized manner. The disciplines provide the "what" that needs to be done, while the phases provide the "when" and "how" it should be done.