Information Systems Analysis & Design (M8748)

Tutorial 4 Answer

1. Define object, class and instance.

   The UML Glossary gives these definitions:

   Object: a single thing or concept, either in a model of an application domain or in a software system, that can be represented as an encapsulation of state, behavior and identity; a member of a class that defines a set of similar objects.

   Class: a descriptor for a collection of objects those are logically similar in terms of their behavior and the structure of their data.

   Instance: a single object, usually called an instance in the context of its membership of a particular class or type (also object instance).

2. What do you think is meant by 'semantics'?

   Semantics is the study of meaning. In object-oriented development it is used to denote the meaning that an element has for the user. The semantics of an object include its purpose, description, relationships and behavior seen from a user perspective.

3. How does the object-oriented concept of message passing help to encapsulate the implementation of an object, including its data?

   Other parts of a system only see an object's interface (services it can perform and operation signatures). Internal details including data are hidden and can only be accessed by a message that contains a valid signature.

4. What is polymorphism?

   Polymorphism means that when one message is sent to objects of different types, each has an appropriate, but different, implementation for its response. The object that sends the message need not know which type of object is addressed. One way of implementing polymorphism is through inheritance and overriding.

5. What is the difference between generalization and specialization?

   Generalization and specialization are opposite sides of the same coin. As we look further up an inheritance hierarchy, the classes become more generalized in their application. As we look down the same hierarchy, they become more specialized. For example Employee is more generalized than HourlyPaidEmployee, while the latter is more specialized.

6. What rules describe the relationship between a subclass and its superclass?
   A subclass inherits all characteristics of its ancestors including its immediate superclass (some may be overridden, but are still technically inherited). Each subclass is different from its ancestors in at least one way.

7. What does it mean to say that an object-oriented system is highly modular?
   Objects are likely to be cohesive (that is, reasonably simple and having a single purpose), well-encapsulated and relatively weakly coupled to other objects in the system.

8. Why is it particularly hard for a designer to anticipate a user's sequence of tasks when using a GUI application?
   The interface for a typical GUI application (for example, a word processor that runs in a windowing environment) consists of many buttons, tools and menu choices. The user can usually select any of these whenever she wishes, and the software designer must cater for this flexibility of use.

9. What does 'object state' mean?
   A condition in the life of an object during which it behaves in a particular way, e.g. waiting for some event or performing some activity. The behavior of an object in response to events depends on its current state.

10. What is an operation signature?
    An operation signature gives the format of a valid message that will cause the operation to execute.