PART - A

1. What is an object?

An object is a combination of data and logic; the representation of some real-world entity.

2. What is the main advantage of object-oriented development?

• High level of abstraction

• Seamless transition among different phases of software development

• Encouragement of good programming techniques.

• Promotion of reusability.

3. What is Object Oriented System development methodology?

Object oriented system development methodology is a way to develop software by building self-contained modules or objects that can be easily replaced, modified and reused.

4. Distinguish between method and message in object.

Method Message

i) Methods are similar to functions, procedures or subroutines in more traditional programming languages. Message essentially are non-specific function calls.

ii) Method is the implementation. Message is the instruction.

iii) In an object oriented system, a method is invoked by sending an object a message. An object understands a message when it can match the message to a method that has the same name as the message.

5. What Is Analysis and Design?

Analysis emphasizes an investigation of the problem and requirements, rather than a solution. For example, if a new computerized library information system is desired, how will it be used.

Design emphasizes a conceptual solution that fulfills the requirements, rather than its implementation. For example, a description of a database schema and software objects. Ultimately, designs can be implemented.

6. What Is Object-Oriented Analysis and Design?

During object-oriented analysis, there is an emphasis on finding and describing the objects—or concepts—in the problem domain. For example, in the case of the library information system, some of the concepts include Book, Library, and Patron.

During object-oriented design, there is an emphasis on defining software objects and how they collaborate to fulfill the requirements. For example, in the library system, a Book software object may have a title attribute and a get Chap-ter method

7. What is UML?

Unified modeling language is a set of notations and conventions and diagrams to describe and model an application.

8. What are the primary goals in the design of UML?

• Provide users a ready – to use expressive visual modeling language so they can develop and exchange meaningful models.

• Provide extensibility and specialization mechanism to extend the core concepts.

• Be independent of particular programming language and development process.

• Provide a formal basis for understanding the modeling language.

• Encourage the growth of the OO tools market.

• Support higher – level development concepts.

• Integrate best practices and methodologies.

9. Define Class Diagram.

The main static structure analysis diagram for the system, it represents the class structure of a system including the relationships between class and the inheritance structure.

10. Define Activity Diagram.

A variation or special case of a state machine in which the states are activities representing the performance of operations and the transitions are triggered by the completion of the operations.

11. What is interaction diagram? Mention the types of interaction diagram.

Interaction diagrams are diagrams that describe how groups of objects collaborate to get the job done interaction diagrams capture the behavior of the single use case, showing the pattern of interaction among objects.

There are two kinds of interaction models

• Sequence Diagram

• Collaboration Diagram.

12. What is Sequence Diagram?

Sequence diagram is an easy and intuitive way of describing the behaviors of a system by viewing the interaction between the system and its environment.

13. What is Collaboration Diagram?

Collaboration diagram represents a collaboration, which is a set of objects related in a particular context, and interaction, which is a set of messages exchanged among the objects with in collaboration to achieve a desired outcome.

14. Define Start chart Diagram.

Start chart diagram shows a sequence of states that an object goes through during its life in response to events. A state is represented as a round box, which may contain one or more compartments. The compartments are all optional.

15. What is meant by implementation diagram?

Implementation Diagrams show the implementation phase of systems development such as the source code structure and the run- time implementation structure.

There are two types of implementation diagrams:

1. Component Diagrams

2. Development Diagrams.

16. Define Component Diagram?

A Component diagrams shows the organization and dependencies among a set of components. A component diagrams are used to model the static implementation view of a system. This involves modeling the physical things that reside on a mode, such as executable, libraries, tables, files and documents.

17. Define Deployment Diagram.

Deployment Diagram shows the configuration of run-time processing elements and the software components, processes, and objects that live in them.

Deployment diagrams are used to model the static deployment view of a system. A deployment diagram is a graph of modes connected by communication association.

18. What is the UP?

A software development process describes an approach to building, deploying, and possibly maintaining software. The Unified Process has emerged as a popular iterative software development process for building object-oriented systems.

19. What is Iterations?

A key practice in both the UP and most other modern methods is iterative development. In this lifecycle approach, development is organized into a series of short, fixed-length (for example, three-week) mini-projects called iterations

20. What is Iterative and Evolutionary Development?

The iterative lifecycle is based on the successive enlargement and refinement of a system through multiple iterations, with cyclic feedback and adaptation as core drivers to converge upon a suitable system. The system grows incrementally over time, iteration by iteration, and thus this approach is also known as iterative and incremental development. Because feedback and adaptation evolve the specifications and design, it is also known as iterative and evolutionary development

21. What are the Phases of Unified Process?

The Unified Process has 4 phases:

– Inception: Requirements capture and analysis

– Elaboration: System and class-level design

– Construction: Implementation and testing

– Transition: Deployment

22. What is Inception?

Inception is the initial short step to establish a common vision and basic scope for the project. It will include analysis of perhaps 10% of the use cases, analysis of the critical non-functional requirement, creation of a business case, and preparation of the development environment.

23. Define Use case modeling?

Use case modeling is a form of requirements engineering. How to create an SRS in what we might call the "traditional" way. Use case modeling is a different and complementary way of eliciting and documenting requirements.

24. Define Use case generalization?

Use case generalization is used when you have one or more use cases that are really specializations of a more general case

Part –B(16 Marks)

1. Explain Types of UML Diagrams with example?

2. Explain Unified Phase and their types with an example?

3. Explain CASE STUDY: THE NEXTGEN POS SYSTEM?

4. Explain Use Case Modeling With example?