SUBJECT: OBJECT ORIENTED ANALYSIS AND DESIGN

Objectives and Relevance:

Uml is an international industry standard graphical notation for describing Software Analysis and Design Systems. It is a modeling language or a Diagrammatic notation for Object Oriented Programming.

Scope:

A way to express the “blue prints “of your System.

Prerequisites:

The Software Development Life Cycle is must and basic Analysis towards any System.

Syllabus (JNTUH)

Unit -1:

Objective:

It deals with

- An Introduction to UML
- Significance of Modeling
- What are the principles of Modeling?
- SDLC (Software Development Life Cycle)
- The Conceptual Model of UML

Syllabus:

Introduction to UML, Importance of Modeling, Principles of Modeling, Object Oriented Modeling, Conceptual Model of UML, Architecture, Software Development Life Cycle.

Unit -2:

Objective:

It deals with

- Classes, attributes, operations and responsibilities
- How do we model the vocabulary of the System?
- Modeling the Distribution of Responsibilities in a System
- Stereotypes , tagged values and constraints
- Modeling Comments
- Modeling new Semantics
- Modeling New Building Blocks, Diagrams, Views, Models, Complex Views
- Organizing diagrams and other artifacts
- Classifiers, Special Properties of attributes and operations and different kinds of classes
- Advanced Dependency,Generalization,association,realization and refinement relationships
- Modeling webs of Relationships
- Creating Webs of Relationships
- Interfaces, Types ,Roles and Realization
- Modeling the Seams in a System
- Modeling Static and Dynamic types
- Making Interfaces understandable and approachable
- Packages, visibility, importing, and Exporting
- Modeling Groups of elements
- Modeling Architectural Views
- Modeling Object Structures
- Forward and Reverse Engineering

Syllabus:

Basic Structural Modeling: Classes, Relationships, Common Mechanisms

Diagrams.

Advanced Structural Modeling: Advanced classes, Advanced Relationships

Interfaces, Types and Roles, Packages

Class and Object Diagrams: Terms, Concepts, Modeling Techniques for

Class & Object Diagrams

Unit -3:

Objective:

It deals with

- Roles, Links, Messages, actions and Sequences
- Modeling flows of control
- Sequencing and Collaboration
- Modeling flows of controls by time ordering
- Modeling flows of control by organization
- Use cases , actors, include, and extend
- Modeling the Behavior of element
- Realizing use cases with collaborations
- Modeling the Context of the System
- Modeling the requirements of a System
- Modeling a workflow
- Modeling an operation
- Forward and Reverse Engineering

Syllabus:

Basic Behavioral Modeling- 1: Interaction, Interaction Diagrams

Basic Behavioral Modeling-2: Use cases, Use case Diagrams, Activity Diagrams

Unit -4:

Objective:

It deals with

- Signal events, call events, time events and change events
- Modeling a family of Signals
- Modeling Exceptions
- Handling events in active and Passive objects
- States , Transitions and activities
- Modeling the Life-time of an object
- Active objects, processes and threads
- Modeling multiple flows of control
- Modeling inter process communication
- Building Thread safe Abstractions
- Time ,Duration and Location
- Modeling timing constraints
- Modeling the Distribution of Objects
- Modeling objects that migrate

- Dealing with Real time and Distributed Systems
- Modeling reactive objects
- Components, interfaces and realization
- Modeling Executables and Libraries
- Modeling tables, files, and documents
- Modeling an API
- Modeling Source code
- Modeling between Logical and physical models
- Nodes and Connections
- Modeling Processor and Devices
- Modeling the Distribution of Components
- Systems Engineering
- Modeling physical Databases
- Modeling Adaptable Systems
- Modeling an embedded System
- Modeling a client/server system
- Modeling a fully Distributed System

Syllabus:

Advanced Behavioral Modeling: Events and Signals, State Machines, Processes and Threads, Time and space, State chart Diagrams

Architectural Modeling: Component, Deployment, Component And Deployment Diagrams.

Unit -5:

Objective:

It deals with

- Patterns and frame works
- Artifact diagrams,
- A Case Study

Syllabus:

Patterns and frame works**,** Artifact diagrams Case-Study: The Unified Library Application.

SYLLABUS - GATE

Not applicable

SYLLABUS - IES

Not applicable

Suggested Text Books:

T1: The Unified Modeling Language User Guide (Pearson Education) 2nd Edition. by Grady Booch, James Rumbaugh, Ivar Jacobson

T2: UML 2 Toolkit, Wiley (Dreamtech India pvt.ltd By Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado.

Reference Text Books:

R1: Fundamentals of Object Oriented Design in UML, Pearson Education by MeilirPage-Jones

R2: Pascal Roques: Modeling Software Systems using UML2, WILEY Dreamtech India pvt.ltd

R3: Atal Kahate: Object Oriented Analysis & Design, The McGraw-Hill Companies

R4: Mark Priestley: Practical Object Oriented Design with UML, THM.

R5: Appling UML and Patterns: An introduction to object - oriented analysis and design and unified process, craig Larman, Pearson Education.

R6: Object oriented Analysis and Design with the uniofied process by John W. Satzinger, Robert B Jackson and Stephen D Burd, Cengage Learning.

R7: UML AND c++, r.C.Lee, and W.M.Tepfenhart, PHI.

R8: Object oriented Analysis, Design and implementation, B.Dathan, S.Ramnath, Universities Press.

R9: OODesign with UML and Java, K.Barclay, J.Savage, Elsevier.

R10: Learning UML 2.0 Russ Miles and Kim Hamilton, O'Reily, SPD.

Websites:

- www.wikipedia.org
- www.uml.edu
- www.Books.google.in

Journals:

- **The Positive Spin by *Candide Smith (Professor of Severa Stern)***
- **Programming Tools – UML tools by Charney**

Subject(Lesson Plan):

Sno	Topic to cover	Suggested Books	TEACHING METOD	No. of Lectures Required
	UNIT-1			
1	Introduction	T1		L1
2	Modeling concept	T1		L2
3	OOM	T1		L3
4	UML Architecture	T1		L4
5	SDLC	T1		L5
	UNIT-2			
6	Classes	T1		L6, L7
7	Relationships	T1 & T2		L8,L9
8	Common Mechanisms	T1		L10,L11,L12,L13
9	Adv..Classes	T1 & T2		L14,L15,L16,L17
10	Adv..Relationships	T1 & T2		L18,L19,L20,L21
11	Interfaces	T1		L22
12	Roles & Types	T1		L23
13	Packages	T1		L24,L25
14	Terms and concepts for class Diagram	T1		L26
15	Modeling Techniques for class Diagram	T1		L27,L28
16	Terms and concepts for object Diagram	T2		L29
17	Modeling Techniques for	T1		L30,L31

	Object Diagram			
	UNIT-3			
18	Interactions	T1		L32
19	Interaction Diagrams	T1		L33,L34
20	Modeling Techniques	T1		L35,L36
21	Use cases	T1		L37,L38
22	Use case Diagram	T1		L39,L40
23	Activity Diagram	T1		L41,L42
	Unit-4			
24	Events	T1		L43,L44,L45
25	Signals	T1		L46,L47,L48
26	State Machines	T1 & T2		L49,L50
27	Processes &Threads	T1		L51,L52,L53
28	Time and Space	T1 & T2		L54,L55
29	State chart Diagrams	T1 & T2		L56,L57
30	Component	T1 & T2		L58,L59
31	Component Diagrams	T1		L60,L61
32	Deployment	T1		L62,L63
33	Deployment Diagrams	T1		L64,L65
34	**UNIT -5**			
35	Patterns and Frame Works	T1 & T2		L66,L67
36	Artifact Diagrams	T1 & T2		L68,L69
37	Case Study	T2		L70,L71

SUBJECT COURSE OBJECTIVE & OUTCOME BASED PLAN (SCOOPS)

FACULTY: k.vijay babu **SUBJECT: Object Oriented Analysis & Design**

DEPT:CSE

Lecture No.	JNTUH Topic	Objective of each Topic	Course Outcome	Practical inferences	Method of Teaching
UNIT – I					
L1	**Introduction**	Introduction to UML	**Co1**	Understand the concept of UML	**M1**
L2	Modeling concept	Explanation about Modeling concepts		Understand the Modeling concept	**M1**
L3	Object Oriented Modeling	Explanation about the Object Oriented Modeling		Strong background knowledge principles of OOM	**M1**
L4	UML Architecture	Explanation about the UML Architecture		Understand about UML Architecture	**M1**
L5	SDLC	Explain about software Development Life Cycle		Understand the SDLC	**M1,M9**
UNIT – II					
Lecture No.	**JNTUH Topic**	**Objective of each Topic**	**Course Outcome**	**Practical inferences**	**Method of Teaching**
L6,L7	Classes	Explanation about the Classes and common Modeling Techniques	**Co2, co3**	Understand the classes and Modeling techniques	**M1**
L8,L9	Relationships	Explanation about the Four relations and common Modeling Techniques		Understand the relations b/w classes and objects	**M1**
L10, L11, L12, L13	Common Mechanisms	Explanation about New Building Blocks , Comments, and New semantics		Understand the New Building Blocks , Comments, and New semantics	**M1**
L14, L15, L16, L17	Advance Classes	Explanation about Common Modeling Techniques		Understand the Semantics of a Class	**M1 ,M4**
L18, L19, L20, L21	Advanced Relationships	Explaining the Webs of relationships		Understand the different types of relationships	**M1**

L22	Interfaces	Explanation about the Interfaces		Understand in detail about Interfaces	**M1**
L23	Roles and Types	Explanation about Roles and Types		Understand the Roles and Types of UML	**M1,M4**
L24, L25	Packages	Explanation about the Packages		Understanding grouping of elements	**M1,M4**
L26	Terms and concepts for class Diagram	Explanation about the terms and concepts		Understand the terms and concepts	**M1**
L27, L28	Modeling Techniques for class Diagram	Explanation about the Common Modeling Techniques for classes		Understand the Vocabulary of a system, non Software things	**M1**
L29	Terms and concepts for object Diagram	Explanation about the terms and concepts		Understand the terms and concepts	**M1**
L30, L31	Modeling Techniques for Object Diagram	Explanation about the Common Modeling Techniques for objects		Understanding the object stuctures	**M1**
UNIT – III					
Lecture No.	**JNTUH Topic**	**Objective of each Topic**	**Course Outcome**	**Practical inferences**	**Method of Teaching**
L32	Interactions	Explain about interactions	**Co4**	To Understand the Basics of sequence and collaboration	**M1**
L33, L34	Interaction Diagrams	Explain about interactions Diagrams		To Understand to draw sequence and collaboration diagrams	**M1,M5**
L35, L36	Modeling Techniques	Explain about Modeling Techniques for interactions		To Understand the Modeling flow of control	**M1**
L37, L38	Use cases	Explain the concepts of Use cases		Understand the Behavior of the use cases	**M1**

L39, L40	Use case Diagram	Explanation about Use case Diagram		Understand and draw the use case diagram	**M1,M5**
L41, L42	Activity Diagram	Explanation about Activity Diagram		Understand and draw the Activity Diagram	**M1,M5**
UNIT – IV DATA RETRIEVAL IN SENSOR NETWORKS					
Lecture No.	**JNTUH Topic**	**Objective of each Topic**	**Course Outcome**	**Practical inferences**	**Method of Teaching**
L43, L44, L45	Events	Explanation about the Events	**Co5**	Understand Events and triggers	**M1**
L46, L47, L48	Signals	Explanation about Signals		Understand signals	**M1**
L49, L50	State Machines	Explanation about State Machine		Understand and draw the State Machine	**M1**
L51, L52, L53	Processes &Threads	Explanation about Processes &Threads.		Learn to implement Processes &Threads	**M1**
L54, L55	Time and Space	Explanation about Time and Space		Understand the Time and Space	**M1**
L56, L57	State chart Diagrams	Overview of the State chart Diagrams		To understand and draw State chart Diagrams	**M1,M5**
L58, L59	Component	Description of the Components		To understand the Component	**M1**
L60, L61	Component Diagrams	Explanation of Component Diagrams		To understand the Executables and libraries	**M1,M5**
L62, L63	Deployment	Explanation about Deployment		To understand the Deployment	**M1**

L64, L65	Deployment Diagrams	Explanation of Deployment Diagrams		To understand The Processes and devices	**M1,M5**
UNIT – V					
Lecture No.	**JNTUH Topic**	**Objective of each Topic**	**Course Outcome**	**Practical inferences**	**Method of Teaching**
L66, L67	Patterns and Frame Works	Explanation about Patterns and Frame Works.	**Co6**	To understand basics of Patterns and Frame Works	**M1**
L68, L69	Artifact Diagrams	Explanation about Artifact Diagrams		Differ from other diagrams	**M1**
L70, L71	Case Study	Explanation about Library management system		Implementation of library management system	**M5,M9**

METHODS OF TEACHING:

M1 : Lecture Method	**M4 : Presentation /PPT**	**M7 : Assignment**
M2 : Demo Method	**M5 : Lab/Practical**	**M8 : Industry Visit**
M3 : Guest Lecture	**M6 : Tutorial**	**M9 : Project Based**

Course Out Comes

At the end of the course student will be able to:

CO56028.1	**Define importance of modeling and building blocks of modeling?**
CO56028.2	**Illustrate classes and relation ships**
CO56028.3	**Describe modeling techniques for classes and object diagrams**
CO56028.4	**Distinguish basic behavioral modeling of interaction diagrams like activity and use case diagrams**
CO56028.5	**Explain state machines and state chart diagrams, component and deployment diagrams to illustrate ATM machine.**
CO56028.6	**illustrate unified library application**

Question Bank:

1. (a) Explain briefly runtime polymorphism illustrating a program in Java or C++.

(b) What are the principles of modeling?

(c)Explain the anti symmetric and transitive properties of aggregation.

2. (a) Explain any three features used in creating abstractions.

(b) Enumerate the steps to model the vocabulary of a system.

(c) Write a simple JAVA applet for printing “Hello, World!” in a web browser.

3. (a) Enumerate the steps to forward engineer a class diagram.

(b) Enumerate the steps to reverse engineer a class diagram.

(c) What are forward engineering and reverse engineering?

4. (a) What is sequence diagram? What is collaboration diagram? What are the features in each one?

(b) What are the properties and common uses of sequence diagrams and collabo-ration diagrams?

5. (a) Sketch the use case diagram for modeling a hospital information system aimed at collecting and storing complete information pertaining to the patients treatment history and disease behavior where actors could be doctor, lab technician, patient, duty nurse, receptionist, visitors etc.

(b) What are the contents and common uses of activity diagrams?

(c) Contrast: action state Vs activity state. How are forking and joining used in activity diagram? Illustrate.

6. (a) Give the sketch of a state machine for the controller in a home security system ,which is responsible for monitoring various sensors around the perimeter of the house. Briefly explain.

(b) Explain the following parts of a transition

i. Event trigger.

ii. Guard condition.

7. Enumerate the steps to model the following. Illustrate UML diagrams and explain briefly.

(a) Modeling processes and devices.

(b) Modeling distribution of components

8. (a) Write a Java program for the Loan class

(b) Draw a deployment diagram for the library system

(c) Draw a class diagram showing architectural overview of the library system

9. Explain briefly about the various diagrams in UML.

10. (a) Explain any three features used in creating abstractions.

(b) Enumerate the steps to model the vocabulary of a system.

(c) Write a simple JAVA applet for printing “Hello, World!” in a web browser.

11. (a) Enumerate the steps to model logical database schema. Give all example class diagrams.

(b) Explain the common uses of class diagrams briefly.

12. (a) Draw a collaboration diagram that specifies the flow of control involved in registering a new student at a school. Explain.

(b) Explain forward engineering and reverse engineering of interaction diagrams.

13. (a) Consider an automated soda machine that gives cool drinks. Draw a use case model of the soda machine.

(b) Draw an extended use case diagram for the soda machine example depicting the ‘extend’, ‘include’ and generalization relationships.

14. (a) Illustrate modeling family of signals and modeling exceptions with UML dia-grams.

(b) Define event and signal. What are the four kinds of events modeled by UML?

15. (a) Enumerate the steps to model an executable release.

(b) What are the contents, common properties and common uses of component diagrams? Explain briefly.

16. (a) Draw activity diagram to inform a person when a loan is due and explain

(b) Draw activity diagram to remove reservations after a specified amount of time and explain diagram.

(C) What are the contents in class diagram?

17. (a) What are interaction diagrams? What are their contents and common proper-ties? Define semantic equivalence between two kinds of interaction diagrams.

(b) Enumerate the steps to model flows of control by time ordering.

18. (a) What are the properties of well-structured use cases?

(b) Enumerate the steps to model the requirements of a system.

(c) Consider a retail system that interacts with customers who place and track orders. In turn, the system will ship orders and bill the customers. Model the behavior of the system will ship orders and bill the customers. Model the behavior of the system by declaring the behaviors as use cases.

19. (a) Enumerate the steps to model the distribution of objects. Explain briefly considering a UML diagram.

(b) Enumerate the steps to model interprocess communication.

20. (a) Define component. What are the differences between components and classes?How are component and interface related?

(b) What are the properties of components?

Tutorial Questions

1. (a) What is genericity?

(b) Enumerate the principle of modeling.

(c) Enumerate any six artifacts.

(d) Briefly explain the extensibility mechanisms in UML. [3+4+3+6]

2. (a) Briefly explain any four standard constraints that apply to generalization relationships.

(b) Briefly explain the four adornments that apply to all association.

(c) What is the stereotype applied to generalization relationships? Give a brief.

3. (a) Enumerate the steps to forward engineer a class diagram.

(b) Enumerate the steps to reverse engineer a class diagram.

(c) What are forward engineering and reverse engineering?

4. (a) What is sequence diagram? What is collaboration diagram? What are the features in each one?

(b) What are the properties and common uses of sequence diagrams and collabo-ration diagrams?

5. (a) Sketch the use case diagram for modeling a hospital information system aimed at collecting and storing complete information pertaining to the patients treatment history and disease behavior where actors could be doctor, lab technician ,patient, duty nurse, receptionist, visitors etc.

(b) What are the contents and common uses of activity diagrams?

(c) Contrast: action state Vs. activity state. How are forking and joining used in activity diagram. Illustrate.

6 (a) What area various parts of a transition. Explain briefly.

(b) Define event and signal. What are the four kinds of events modeled by UML?

7. (a) Enumerate the steps to model adaptable systems. Illustrate with a UML diagram.

(b) Enumerate the steps to model an executable release. Illustrate with a UML diagram.

(c) What are the common uses of component diagrams?

8. (a) Draw the use case diagram for the library system and explain the relationships.

(b) Draw a sequence diagram for the use cse Lend Item and explain

(c) Draw a collabortion diagram for the add Title use case and explain

9. (a) What are the various views considered in modeling a system's architecture? Explain.

(b) What is the UML approach to software development life cycle? Explain the various phases

10. (a) Enumerate the steps to model webs of relationships.

(b) Contrast simple aggregation with composite aggregation. What is association class?

(c) Illustrate with examples how realization is used to specify the relationships between the following.

i. Interface Vs. Class

ii. Interface component.

11. (a) Enumerate the steps to model simple collaborations. Give an example classdiagram.

(b) What are the contents in class diagram?

12. (a) What are interaction diagrams? What are their contents and common proper-ties? Define semantic equivalence between two kinds of interaction diagrams.

(b) Enumerate the steps to model flows of control by time ordering. [8+8]

13. (a) What are the properties of well-structured use cases?

(b) Enumerate the steps to model the requirements of a system.

(c) Consider a retail system that interacts with customers who place and track orders. In turn, the system will ship orders and bill the customers. Model the behavior of the system will ship orders and bill the customers. Model the behavior of the system by declaring the behaviors as use cases.

14. (a) Enumerate the steps to model the distribution of objects. Explain briefly considering a UML diagram.

(b) Enumerate the steps to model inter process communication

15. (a) Define component. What are the differences between components and classes? How are component and interface related?

(b) What are the properties of components?

(c) What are standard stereotypes UML defines that apply to components

16. (a) Draw the use case diagram for the library system and explain the relationships.

(b) What are the packages in the Library system? Explain

Objective Questions

1. Computer systems are designed by

a. simplifying requirements of system
b. breaking of the system into smaller self-contained co-operating subsystems
c. breaking up the systems into independent parts
d. modular design

2.Functions and procedures are

a. not useful in designing computer systems
b. old fashioned and they are not useful
c. useful in designing computer systems
d. have side effects which require special care if they are used as subsystems

3. A subsystem of a complex system must specify

a. what task it performs
b. how it performs a task
c. with which subsystems it co-operates
d. how it co-operates with other systems

4. A subsystem of a complex system must

a. ii, iii
b. ii, iv
c. iii, iv

5. A subsystem of a complex system

(i) should be reusable in other complex system
(ii) must not be able to inherit the properties of other subsystems

(iii)must have clearly specified responsibilities

(iv)must know the stimuli to which it should respond

a. i, ii, iii
b. ii, iii, iv
c. i, iii, iv
d. i, ii, iv

6. By polymorphism of a subsystem we mean

a. it should be reusable
b. it should have polymorphic data types
c. it should accept generic commands and interpret appropriately
d. it should morph polygons

7. The advantages of object-oriented modelling are

a. i, ii
b. i, iii
c. ii, iii
d. i, iv

Objects are

(i)tangible entities

(ii)intangible entities

(iii)transient entities

(iv)uniquely identifiable

a. i, ii

b. i, ii, iii

c. i, ii, iii, iv

d. i, ii, iv

8. A class is

a. a group of objects

b. template for objects of a particular type

c. a class of objects

d. a classification of objects

9. All objects have

(i) attributes

(ii) states

(iii)a set of operations

(iv) a unique identity

a. i, ii, iii

b. ii, iii, iv

c. i, iii, iv

d. i, ii, iii, iv

10. In UML diagram of a class

a. state of object cannot be represented

b. state is irrelevant

c. state is represented as an attribute

d. state is represented as a result of an operation

11. Attributes are assigned value

a. when operations are performed on an object

b. when instances of objects are defined

c. when methods are invoked

d. when classes are identified

12. The following are intangible entities which can be defined as objects

(i) a motor car

(ii) a bank account

(iii) an aircraft

(iv) a linked list

a. i, ii

b. ii, iv

c. iii, iv

d. ii, iii, iv

13. A query operation on a object

a. has side effect

b. has no side effects

c. changes the state of an object

d. is not allowed

14. An instance of an object is created by a

a. query operation

b. update operation

c. constructor operation

d. open operation

15. An update operation in an object instance

a. updates the class

b. has no side effects

c. deletes an instance

d. alters values of attribute(s) of an object instance

16. In object-oriented design

a. operations and methods are identical

b. methods specify algorithms whereas operations only state what is to be done

c. methods do not change values of attributes

d. methods and constructor are same

17. By abstraction in object-oriented modelling we mean picking

a. only attributes appropriate to model an object

b. only operations

c. both operation and attributes with operations appropriate to model an object

d. the appropriate abstract data type

18. By encapsulation in object-oriented modelling we mean

a. encapsulating data and programs
b. hiding attributes of an object from users
c. hiding operations on object from users
d. hiding implementation details of methods from users of objects

19. Encapsulation in object-oriented modelling is useful as

a. it allows improving methods of an object independent of other parts of system
b. it hides implementation details of methods
c. it allows easy designing
d. encapsulates attributes and operations of object

20. Objects may be viewed as

a. clients in a system
b. servers in a system
c. as both clients and servers in a system
d. neither as clients nor as servers in a system

21. Inheritance in object-oriented system is used to

a. create new classes from existing classes
b. add new operations to existing operations
c. add new attributes to existing attributes
d. add new states to existing states

22. Inheritance in object-oriented modelling can be used to

a. generalize classes
b. specialize classes
c. generalize and specialize classes
d. create new classes

23. When a subclass is created using inheritance the resulting class

a. may have only attributes of parent class
b. may have only operations of parent class
c. may have new operations only in addition to those in parent class
d. may have new attributes and new operations in addition to those of the parent class

24. By polymorphism in object-oriented modelling we mean

a. the ability to manipulate objects of different distinct classes
b. the ability to manipulate objects of different distinct classes knowing only their common properties
c. use of polymorphic operations
d. use of similar operations to do similar things

25. A polymorphic operation

a. has same name
b. has same name but uses different methods depending on class
c. uses different methods to perform on the same class
d. uses polymorphic method

26. Given a word statement of a problem potential objects are identified by selecting

a. verb phrases in the statement
b. noun phrases in the statement
c. adjectives in the statement
d. adverbs in the statement

27. Given a word statement of problem potential operations appropriate for objects are identified by selecting

a. verb phrases in the statement
b. noun phrases in the statement
c. adjectives in the statement
d. adverbs in the statement

28. Objects selected to model a system

(i) must be essential for functioning of the system
(ii) must have all attributes which are invariant during operations of a system
(iii) must have attributes relevant for performing services of object
(iv) must be able to perform assigned services

a. i, ii, iii
b. ii, iii, iv

c. i, iii, iv

d. i, ii, iii, iv

An object is selected for modelling a system provided

a. its attributes are invariant during operation of the system

b. its attributes change during operation of the system

c. it has numerous attributes

d. it has no attributes relevant to the system

29. An object is considered an external entity in object-oriented modelling

a. its attributes are invariant during operation of the system

b. its attributes change during operation of the system

c. it has numerous attributes

d. it has no attributes relevant to the system

30. Object-oriented system modelling using CRC method gives

a. Java programs for the system

b. C++ programs for the system

c. Classes of the system, their responsibilities and collaborating classes

d. Objective C programs for the system

31. The expansion of the acronym CRC is

a. Collecting Responsibilities Classes

b. Collaborating with Relevant Classes

c. Class Responsibilities and Collaborators

d. Creating Relevant Classes

32. In CRC based design a CRC team consists of

(i) one or two user's representatives

(ii) several programmers

(iii) project coordinators

(iv) one or two system analysts

a. i, ii

b. i, iii

c. i, iii, iv

d. i, ii, iii, iv

33. A class index card contains besides class name

(i) superclasses and subclasses

(ii) short description of class

(iii)collaborators

(iv)private responsibilities of class

(v)contract(s) with collaborators

a. i, ii, iii

b. i, iii, iv, v

c. i, ii, iii, iv

d. i, ii, iii, iv and v

List of Topics for Students Seminars:

- Principles of Modeling
- Why we need to Model?
- Importance of Use case Diagram
- Importance of Class Diagram
- Importance of Sequence Diagram
- Importance of Component Diagram
- Importance of Deployment Diagram