OBJECT-ORIENTED MODELING AND DESIGN

PART - A

UNIT - 1
INTRODUCTION, MODELING CONCEPTS, CLASS MODELING: What is Object Orientation? What is OO development? OO themes; Evidence for usefulness of OO development; OO modeling history.

Modeling as Design Technique: Modeling; abstraction; The three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips.

7 Hours

UNIT - 2
ADVANCED CLASS MODELING, STATE MODELING: Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips. State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips.

6 Hours

UNIT - 3
ADVANCED STATE MODELING, INTERACTION MODELING: Advanced State Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Practical tips. Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models.

6 Hours

UNIT - 4
PROCESS OVERVIEW, SYSTEM CONCEPTION, DOMAIN ANALYSIS: Process Overview: Development stages; Development life cycle. System Conception: Devising a system concept; Elaborating a concept; Preparing a problem statement. Domain Analysis: Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis.

7 Hours
PART - B

UNIT - 5
APPLICATION ANALYSIS, SYSTEM DESIGN: Application Analysis: Application interaction model; Application class model; Application state model; Adding operations.
Overview of system design; Estimating performance; Making a reuse plan; Breaking a system into sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy; Handling boundary conditions; Setting the trade-off priorities; Common architectural styles; Architecture of the ATM system as the example.

7 Hours

UNIT - 6
CLASS DESIGN, IMPLEMENTATION MODELING, LEGACY SYSTEMS: Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recursing downwards, Refactoring; Design optimization; Reification of behavior; Adjustment of inheritance; Organizing a class design; ATM example.
Implementation Modeling: Overview of implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing.
Legacy Systems: Reverse engineering; Building the class models; Building the interaction model; Building the state model; Reverse engineering tips; Wrapping; Maintenance.

7 Hours

UNIT - 7
DESIGN PATTERNS – 1: What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description.
Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber.

6 Hours

UNIT - 8
DESIGN PATTERNS – 2, IDIOMS: Management Patterns: Command processor; View handler.
Idioms: Introduction; What can idioms provide? Idioms and style; Where to find idioms; Counted Pointer example.

6 Hours
TEXT BOOKS:

REFERENCE BOOKS:
5. Design Patterns- Elements of Reusable Object-Oriented Software - E. Gamma, R. Helm, R. Johnson, J. Vlissides, Addison-Wesley, 1995.