| Savitribai Phule Pune University |
|---|
| **Second Year Artificial Intelligence & Machine Learning (2020 Course)** |
| **218544: Object Oriented Programming** |

| Teaching Scheme: | Credit Scheme: | Examination Scheme: |
|---|---|---|
| Theory (TH): 03hrs/Week | 03 | Mid_Semester: 30 Marks<br>End_Semester: 70 Marks |

**Prerequisites:** Principles of Programming Languages

**Course Objectives:**
1. Apply concepts of object-oriented paradigm.
2. Design and implement models for real life problems by using object-oriented programming.
3. Develop object-oriented programming skills.

**Course Outcomes:**

On completion of the course, students will be able to–

**CO1:** Differentiate various programming paradigms.

**CO2:** Identify classes, objects, methods, and handle object creation, initialization, and Destruction to model real-world problems.

**CO3:** Identify relationship among objects using inheritance and polymorphism principles.

**CO4:** Handle different types of exceptions and perform generic programming.

**CO5:** Use of files for persistent data storage for real world application.

**CO6:** Apply appropriate design patterns to provide object-oriented solutions.

| COURSE CONTENTS | | |
|---|---|---|

| Unit I | Foundations of Object Oriented Programming | 06 hrs |
|---|---|---|

**Introduction OOP** : Software Evolution, Introduction to Procedural, Modular, Object-Oriented and Generic Programming Techniques, Limitations of Procedural Programming, Need of Object-Oriented Programming, Fundamentals of Object-Oriented Programming: Objects, Classes, Data Members, Methods, Messages, Data Encapsulation, Data Abstraction and Information Hiding, Inheritance, Polymorphism, Static and Dynamic Binding, Message Passing.

| Case Study | Model a real world scenario (vehicle class, fruit class, student management in university etc.) using Object Oriented Paradigm |
|---|---|
| **Mapping Course Outcomes for Unit 1** | CO1 |

| Unit II | Classes, Objects and Methods | 06 hrs |
|---|---|---|

**Class:** Creating a Class, Visibility/Access Modifiers, Encapsulation, Methods: Adding a Method to Class, Returning a Value, Adding a Method That Takes Parameters, The 'this' Keyword, Method Overloading, Object Creation, Using Object as a Parameters, Returning Object, Array of Objects, Memory Allocation: 'new', Memory Recovery: 'delete', Static Data Members, Static Methods, Forward Declaration, Class as Abstract Data Types (ADTs), Classes as Objects.

| Case Study | Represent a vector using class and include appropriate methods to perform various tasks. | |
|---|---|---|
| Mapping of Course Outcomes for Unit II | CO2 | |
| **Unit III** | **Constructors and Destructors** | **06 hrs** |

**Constructors**: Introduction, Use of Constructor, Characteristics of Constructors, Types of Constructor, Constructor Overloading, Dynamic Initialization of an Object, Constructor with Default Arguments, Symbolic Constants, Garbage Collection: Destructors and Finalizes.

| Case Study | A book shop inventory | |
|---|---|---|
| Mapping of Course Outcomes for Unit III | CO2 | |
| **Unit IV** | **Inheritance and Polymorphism** | **06 hrs** |

**Inheritance:** Introduction, Need of Inheritance, Types of Inheritance, Benefits of Inheritance, Cost of Inheritance, Constructors in derived Classes, Method Overriding, Abstract Classes and Interfaces.
Polymorphism and Software Reuse: Introduction, Types of Polymorphism (Compile Time and Run Time Polymorphism), Mechanisms for Software Reuse, Efficiency and Polymorphism

| Case Study | A bank account system | |
|---|---|---|
| Mapping of Course Outcomes for Unit IV | CO3 | |
| **Unit V** | **Exception Handling and Generic Programming** | **06 hrs** |

**Exception:** Errors, Types of Errors, Exception and its Types, Exception-Handling Fundamentals, Uncaught Exception, Using try and Catch, Multiple Catch Clauses, Nested Try Statements, User Define Exception using Throw.
**Generics:** What are Generics? Introduction to Language Specific Collection Interface: List Interface and Set Interface, Collection Classes: ArrayList Class and LinkedList Class.

| Case Study | Exception handling and generic programming using array list (ArrayList class) | |
|---|---|---|
| Mapping of Course Outcomes for Unit V | CO4 | |
| **Unit VI** | **File Handling and Design Patterns** | **06 hrs** |

**File Handling**: Introduction, Concepts of Stream, Stream Classes, Byte Stream Classes, Character Stream, Classes, Using Stream, and Other Useful I/O Classes, Using the File Class, Input/output Exceptions, Creation of Files, Reading/Writing Character, Reading/Writing Bytes, Handling Primitive Data Types, Concatenating and Buffering Files, Random Access Files.
**Design Patterns:** Introduction, Types of Design Patterns, Adapter, Singleton, Iterator

| Case Study | Student Management System | |
|---|---|---|
| Mapping of Course Outcomes for Unit VI | CO5 and CO6 | |

| Text Book: |
|---|
| 1. An Introduction to Object Oriented Programming (3rd Ed), by Timothy A. Budd, published by Addison-Wesley,2002 |
| 2. E. Balaguruswamy, "Object Oriented Programming Using C++ and Java", Tata McGraw Hill |
| **Reference Books:** |
| 1. Object-Oriented Programming and Java by Danny Poo (Author), Derek Kiong (Author), Swarnalatha Ashok (Author)Springer; 2nd ed. 2008 edition (12 October 2007), ISBN-10: 1846289629, ISBN-13: 978-1846289620,2007 |
| 2. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd. |
| 3. Object-Oriented Design Using Java, Dale Skrien, McGraw-Hill Publishing, 2008, ISBN - 0077423097, 9780077423094. 4. UML for Java Programmers by Robert C. Martin, Prentice Hall, ISBN 0131428489, 2003. |