UKA TARSADIA UNIVERSITY

M.C.A. (1st Semester) Syllabus, 2025-2026

Course Code:CS7043 Course Title: Object Oriented Programming

Course Credits: [Lecture: 03, Tutorial: 00, Practical: 02]

Prerequisites: Basics of Programming

Objectives: To design and develop an application usingobject-oriented programming concepts namely encapsulation, inheritance, interface, packages, and exception handling.

Course Outcomes: Upon completion of the course, students shall be able to

CO1: Analyze and apply object-oriented programming concepts by comparing programming paradigms, identifying lexical elements, classifying data types, evaluating operators, and implementing control flow statements.

CO2: Demonstrate the preventions to the states of structured data objects using class and methods fromunauthorized access.

CO3: Apply the concept of inheritance to reducethe length of code and polymorphism using abstract classes.

CO4: Identify the need of Interface & Package and implement them.

CO5: Evaluate and implement exception handlingmechanisms.

Course Objective and Course Outcomes Mapping:

- To design and develop an application using object-oriented programming: CO1 to CO5

Programme Outcomes:The student will have

PO1-Computational Knowledge: Apply knowledge of computing fundamentals, computing specialisation, mathematics, and domain knowledge appropriate for the computing specialisation to the abstraction and conceptualisation of computing models from defined problems and requirements.

PO2-Problem Analysis: Identify, formulate, research literature, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences, and relevant domain disciplines.

PO3-Design /Development of Solutions: Design and evaluate solutions for complex computing problems, and design and evaluate systems, components, or processes that meet specified needs with appropriate consideration for public health and safety, cultural, societal, and environmental considerations.

PO4-Conduct Investigations of Complex Computing Problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of t h e information to provide valid conclusions.

PO5-Modern Tool Usage: Create, select, adapt and apply appropriate techniques, resources, and modern computing tools to complex computing activities, with an understanding of the limitations.

PO6-Professional Ethics: Understand and commit to professional ethics and cyber regulations, responsibilities, and norms of professional computing practices.

PO7-Life-long Learning: Recognise the need, and have the ability, to engage in independent learning for continual development as a computing professional.

PO8-Project management and finance: Demonstrate knowledge and understanding of the computing and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO9-Communication Efficacy: Communicate effectively with the computing community, and with society at large, about complex computing activities by being able to comprehend and write effective reports, design documentation, make effective presentations, and give and understand clear instructions.

PO10-Societal and Environmental Concern: Understand and assess societal, environmental, health, safety, legal, and cultural issues

within local and global contexts, and the consequential responsibilities relevant to professional computing practices.

PO11-Individual and Team Work: Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary environments.

PO12-Innovation and Entrepreneurship: Identify a timely opportunity and using innovation to pursue that opportunity to create value and wealth for the betterment of the individual and society at large.

Programme Outcomes and Course Outcomes mapping:

Course Outcomes	Programme Outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	P10	P11	P12
CO1	3	2	1	0	0	2	2	0	0	2	0	0
CO2	2	3	2	2	2	2	1	0	0	2	1	0
CO3	3	2	2	0	2	2	0	0	1	1	0	0
CO4	2	3	2	2	3	2	2	0	0	1	1	0
CO5	2	2	2	2	2	3	2	0	2	3	1	0

Curriculum Unit Titles

Units	Unit Description	Weightage
1	Overview of Object Oriented Programming	[20%]
	1.1. Data Types: Overview, Types, Type conversion: Implicit and Explicit	
	1.2. Operators: Types, Precedence and associativity, Expression evaluation	
	1.3. Control flow statements	
	1.4. Object Oriented Programming: Overview, Use case, Features and Advantages	
2	Encapsulation	[15%]
	2.1. Classes: Defining, Declaration, Creation and Access Modifiers	
	2.2. Objects: Creation, Declaration, Instantiation	
	2.3. Constructors: Use case and Types	
	2.4. Methods: Declaration, Invocation, Types: Getter & Setter	
	2.5. Static and Final Keywords	
3	Inheritance	[15%]
	3.1. Need, Use case and Types	
	3.2. Base Class: Accessibility of Properties & methods, Constructor calling approach	
	3.3. Derived Class: Accessibility of Properties & methods, Constructor calling approach	
4	Polymorphism	[15%]
	4.1. Need, Use case and Types	
	4.2. Polymorphism: Use case, Types: Overloading, Overriding	
	4.3. Abstract class and methods	
5	Interfaces and Package	[20%]
	5.1. Interface: Essentials, Characteristics, Declaration and Implementation	
	5.2. Extending Interfaces, Static and Default Methods in Interface	
	5.3. Polymorphism in interface	
	5.4. Package: Need, Creation and Usage of User Define Packages	
6	Exception Handling	[15%]
	6.1. Exceptions: Define, Need, Types: Checked, Unchecked	

6.2. Handling Exceptions: Throw and Try statements

6.3. Usages of built-in Exceptions: Arithmetic, Collections, Class, Object and Input – Output

6.4. User define Exception: Creation, Usage and Implementation

Course Units and Course Outcomes Mapping:

Unit No.	Unit	Course Outcomes				
		CO1	CO2	CO3	CO4	CO5
1	Overview of Object Oriented Programming	✓				
2	Encapsulation	✓	✓			
3	Inheritance	✓	✓	✓		
4	Polymorphism	✓	✓	✓	✓	
5	Interfaces and Package	✓	✓	✓	✓	✓
6	Exception Handling	✓	✓	✓	✓	✓

Computing Environment:

A student must have the following computing environment in the laboratory and/or on his/her laptop.

- CE#1: Any IDE supporting Java run time environment

Text Books

1. Buyya, R. –"Object-Oriented programming with Java: Essentials and Applications" -McGraw Hill

References:

1. Uttam K. Roy –"Advanced Java Programming"– Oxford Higher Education.
2. Herbert Schildt –" Java The Complete Reference", 11th Edition – McGraw Hill Education
3. P Radha Krishna–" Object Oriented Programming through JAVA"– University Press
4. C Xavier – "Java Programming A Practical Approach" – Tata McGraw Hill Education