COMSATS University Islamabad
Department of Computer Science
Course Description Form (CDF)

Course Information	
Course Code: **CSC241**	Course Title: **Object Oriented Programming**
Credit Hours: **4(3,1)**	Lecture Hours/Week: **3**
Lab Hours/Week: **3**	Pre-Requisites: **CSC103-Programming Fundamentals**

Course Objective

- To introduce the object-oriented programming paradigm;
- To teach in depth the philosophy of object-oriented design and concepts of encapsulation, abstraction, inheritance and polymorphism;
- To develop understanding of sub typing and generic types;
- To explain the usage of library components;
- To develop code that responds to exception conditions raised during execution;
- To develop understanding of event handlers for use in reactive systems, such as GUIs;
- To demonstrate implementation of the concepts.

Course Content

This course emphasizes the concepts of object-oriented techniques used in developing computer-based system. The topics include: Overview of Object-Oriented Programming; Classes & its Concepts; Problem Solving in Object Oriented Paradigm; Inheritance; Polymorphism; Library Components; Object Oriented Concepts of File Handling; Swing Classes; Events & Event Handlers; and Canonical Uses.

Unit wise Major Topics:

Unit #	Topic	No. of Teaching Hours
1.	Overview of Object-Oriented Programming: Concepts, Principles, Evolution, and Advantages.	1.5
2.	Classes: Data & Member Functions; Member Access; Constructors & Finalizer; Static Data Members, Memory Allocation; and Encapsulation: Privacy & Visibility of Class Members.	13.5
3.	Object-Oriented Paradigm: Problem Solving, Design: Class Diagram, Forward Engineering of Class Diagram to Code, Reverse Engineering of Code to Class Diagram, Decomposition into Objects; Class-Hierarchy, Design for Modeling, Inheritance, Subclasses, Encapsulation, and Method Overriding.	9
4.	Sub-Typing: Polymorphism, Dynamic Dispatch: Method-Call, Implicit Up-Casts, Explicit Down-Casts, Notion of Behavioral Replacement (Subtypes acting like Super-Types); Relationship between Sub-Typing and Inheritance, Abstract Base Classes & Interface, Generic Types, and Static & Dynamic Typing.	9
5.	Library Components: Collection Classes & Iterators; and Object-Oriented Concepts of File Handling.	4.5
6.	Swing Classes; Events & Event Handlers; Canonical Uses: GUIs, Reactive Framework, and Externally & Program-Generated Events.	7.5

Total Contact Hours					**45**	

Mapping of CLOs and GAs

Sr.#	Unit #	Course Learning Outcomes	Blooms Taxonomy Learning Level	GA
		CLO's for Theory		
CLO-1	1-2	Demonstrate fundamental principles and concepts of object-oriented programming.	*Understanding*	2
CLO-2	3-4	Apply the concepts of object-oriented programming principles along with interfaces and exception handling to solve a real-world problem.	*Applying*	2-4
CLO-3	3-4	Apply event handling model to develop event-driven programs that respond to user events.	*Applying*	2-4
		CLO's for Lab		
CLO-4	3-4	Implement a small module utilizing Object-Oriented design.	*Applying*	2-4
CLO-5	1-6	Develop a GUI based project for a real-world problem in a team environment.	*Creating*	1-4,6,10

CLO Assessment Mechanism

Assessment Tools	CLO-1	CLO-2	CLO-3	CLO-4	CLO-5
Quizzes	Quiz 1	Quiz 2	Quiz 3&4	-	-
Assignments	Assignment 1	Assignment 2	Assignment 3&4	Lab Assignments	Lab Assignments
Mid Term Exam	Mid Term Exam	Mid Term Exam	Mid Term Exam	Lab Mid Term Exam	-
Final Term Exam	Final Term Exam			-	Lab Project/ Lab Final Term Exam

Text and Reference Books

Textbook:
1. Introduction to Java Programming and Data Structures, Comprehensive Version, Y. Liang, Y. Daniel Liang, Pearson, 2019.

Reference Books:
1. Concise Guide to Object-Oriented Programming, Kingsley Sage, Springer, 2019.
2. Absolute Java, Savitch, W. & Mock, K., Pearson, 2016.