Department of Information Technology

Course Handout

Session (Semester):	July to December 2024 (III)
Branch:	Computer Science and Engineering (Cyber Security)
Section:	A,B
Class:	LAB Course
Course Name (Code):	Object Oriented Programming Lab (IT_2166)
Contact Hours/Week:	L: 0, T: 0, P: 3, C: 1
Course Coordinator:	Mr. Gaurav Prasad
Course Instructor:	Mr. Gaurav Prasad,Dr.Susheel kumar

A. Introduction: Object oriented techniques have revolutionized the software development process and are used tremendously in IT industry to develop software products of various kinds. The course is designed to give students an in-depth understanding of the basic concepts of object-oriented programming such as encapsulation, inheritance, and polymorphism and JavaFX using Java programming language as an aid in tool. The Lab course curriculum and structure has been divided into eight basic modules which covers the programming aspects related with object-oriented domain such as class, objects, constructors, strings, inheritance, interfaces,exception handling, multithreading and JavaFX. The lab course will be taught with the help of several teaching aides such as power point presentation and via live debugging and execution demonstrations of several programming problems using Eclipse tool.

B. The main objective of the course are as follows:

- To understand the programming skills using object orientation concepts through Java

  to write, compile and execute application programs in Java.
- To develop skills to implement the concepts of object orientation like abstraction, polymorphism, inheritance, encapsulation.
- To develop skills of Exception Handling, Multithreading, concurrent programming, Stings in Java
- To develop efficient Graphical User Interfaces (GUI) using JavaFx components to understand event handling mechanism of Java.

C. Assessment Rubrics:

Criteria	Description	Duration	Maximum Marks
Internal Assessment (Summative)	Viva (bimonthly) Execution (bimonthly)	During normal Lab hours (2 rounds)	Viva-12 marks Execution-06 marks
	Record/File (bimonthly)		3*6=18
	Internal Lab Examination	Will be informed by the Instructor	(Viva-8,Write up-6,Execution-10)-24
End Term Exam (Summative)	End Semester Exam/ Make-up Examination	120 minutes	40 (20-execution,10-viva,10-writeup)
	Total		100
Attendance (Formative)	A minimum of 75% Attendance is required to be maintained by a student to be qualified for taking up the End Semester examination. The allowance of 25% includes all types of leaves including medical leaves.		

D. Syllabus

Simple Java programs using control structures and Arrays, Programs using Classes, objects, methods, Programs on Constructors and static members, Programs using Inheritance, Packages,
Interfaces and Generics, Programs using Exceptions and Multithreading, GUI based programs
using Javafx

E. Text Books:

T1. Herbert Schildt and Dale Skrien, Java Fundamentals – A Comprehensive Introduction, (1e), McGrawHill, 2015

T2. Herbert Schildt, The Complete Reference JAVA 2, (10e), Tata McGrawHill, 2017

T3. Dietel and Dietel, Java How to Program, (9e), Prentice Hall India, 2012

F. Reference Books:

R1.Bruce Eckel, *Thinking in Java*, (5e), Prentice Hall, 2013

R2. Herbert Schield , *Java A beginner's Guide*, (6e), 2014

R3. Dietel and Dietel, *Java How to Program*, (9e), Prentice Hall India, 2012

R4. Steven Holzner, *Java 2 Programming Black Book*, DreamTech, India, 2005

G. Course Outcomes: At the end of the course, students will be able to

[CO.1] Develop a software using object-oriented paradigm.

[CO.2] Use the constructs of an object-oriented language Java in achieving object-oriented principles

[CO.3] Understand the packages of Java to develop concurrent programs

[CO.4] Achieve high level reusability using generics

[CO.5] Design and implement small Java applications using JavaFx

H. Program Outcomes and Program Specific Outcomes

[PO.1]. Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

[PO.2]. Problem analysis: Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

[PO.3]. Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

[PO.4]. Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

[PO.5]. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

[PO.6]. The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal, and cultural issues, and the consequent responsibilities relevant to the professional engineering practice.

[PO.7]. Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

[PO.8]. Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practices.

[PO.9]. Individual and teamwork: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

[PO.10]. Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

[PO.11]. Project management and finance: Demonstrate knowledge and understanding of the engineering 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.

[PO.12]. Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

[PSO1]: Incorporating insights of cybersecurity principles, concepts, ethics, and best practices.

[PSO2]: Ability to apply methods and tools to identify, analyse and handle cyber security threats.

[PSO3]: Demonstrate the knowledge towards the domain specific initiatives of cyber security.

[PSO4]:Effective collaboration and communication skills necessary for teamwork in cybersecurity.

I. Lecture Plan:

Lectures	Major topics	Topics	Mode of delivery	Corresponding CO
1	JAVA FEATURES & SIMPLE PROGRAMS USING CONTROL STRUCTURES	Introduction and Course Hand-out briefing	Lecture	NA
		Java Programming Fundamentals: Java language, Java development kit, Simple Programs, The Java Keywords, Identifiers in Java, The Java Class Libraries ,Data types & Operators: Primitive data types, Literals, Variables, and their scope, Operators and their precedence, Expressions Control structures: if, switch, for, while, do-while, break, continue, Literals, Variables, Type Conversion and casting, wrapper classes, Boxing and Unboxing	Lecture Demo	CO1
			Lecture Demo	CO1
2			Lecture Demo	CO1
3	JAVA FEATURES & DATA TYPES, OPERATORS AND ARRAYS	JAVA language basics, simple example programs, examples on operators and expressions, expression evaluation, 1D Arrays, 2D Array, multi-dimension Array, Variable Length Array, Operators	Lecture Demo	CO2
4	INTRODUCTION TO CLASSES	Class Fundamentals, Creating Objects, Reference Variables and Assignment, simple Methods Returning from a method, Constructors, Parameterized Constructors, The new operator, Garbage Collection and Finalizers, This keyword, Controlling Access to Class Members, Tutorial: Pass Objects to Methods, Arguments Passing, Returning Objects, Arrays and Strings, Array declaration and initialization, Method Overloading, Overloading Constructors, Understanding static, static members, static methods and static block.	Lecture Demo	CO1 CO2
5				
6	CONSTRUCTORS AND STATIC MEMBERS, STRINGS	Utilize various types of constructors, Overloading constructors, Understanding static , Know different ways of creating String objects and constants, Learn and use string handling methods, Know the difference between String and String Buffer classes	Lecture Demo	CO1 CO2

7	INHERITANCE AND PACKAGES	Inheritance Basics, Using Super, Creating a multilevel hierarchy,Method overriding, Dynamic method dispatch,Using Abstract class, using final with	Lecture Demo	CO1 CO2 CO3

8		Inheritance, Example Programs on Inheritance, PACKAGES: Package Fundamentals, Packages and Member Access, Importing Packages, Static Import Tutorial: Creating and implementing interfaces and packages.		
9	INTERFACES AND EXCEPTION HANDLING	Interface Fundamentals, Creating and Implementing an Interface, Using Interface References, Implementing Multiple Interfaces,	Lecture Demo	CO3 CO2 CO1
		Constants in Interfaces, Extending Interfaces, Nested Interfaces, Fundamentals, Exception types, Uncaught Exceptions, check to uncheck Exception	Lecture Demo	
10		Using try and catch, multiple catch clauses, nested try statements, Throw, throws, finally, built-in exceptions, creating own exception.	Lecture Demo	
11	MULTITHREADED PROGRAMMING	Thread Model: thread priorities, synchronization, messaging,main thread, creating single thread and multiple threads, using isAlive(), join(),Interthread communication, suspending, resuming, and stopping threads,Thread Synchronization,Creating Threads and Thread synchronization	Lecture Demo	CO1 CO2 CO3
12	JAVA GENERICS	Generic fundamentals, Generic class, Bounded types, using wildcard arguments, Generic constructors Generic restrictions, example programs,Java Generics Example	Lecture Demo	CO1 CO2 CO3 CO4
13	JAVAFX	Introducing Javafx Gui Programming using Javafx: Basic concepts,Application Skeleton, Event handling Event handling using Button, TextField and Label control,Tutorial: Example program to handle events Exploring Javafx Controls: Toggle button, Radio button,Check box, List view,Combo box, Working with Canvas,Tutorial: Example program which uses Checkbox, Canvas, List view.	Lecture Demo	CO1 CO2 CO5

J. Course Articulation Matrix: (Mapping of COs with POs)

CO	STATEMENT	CORRELATION WITH PROGRAM OUTCOMES														
		PO.1	PO.2	PO.3	PO.4	PO.5	PO.6	PO.7	PO.8	PO.9	PO.10	PO.11	PO.12	PSO1	PSO2	PSO3	PSO4
CO.1	Develop a software using object-oriented paradigm.	1	2	3		1			1		1		1	1	1		
CO.2	Use the constructs of an object-oriented language Java in achieving object-oriented principles		1	2	1	1			1		1		1	1	1		
CO.3	Understand the packages of Java to develop concurrent programs		1	2	1	1			1		1		1				
CO.4	Achieve high level reusability using generics		1	2	1	1			1		1		1		1		
CO.5	Design and implement small Java applications using JavaFx		1	2	1	1			1		1		1				

Note: CO to PO & PSO mapping level (1 – low, 2- moderate and 3 – substantial).