COURSE CONTENT

JAVA PROGRAMMING LAB

IV Semester: ME / CE

Course Code	Category	Hours / Week			Credits	Maximum Marks		
		L	T	P	C	CIA	SEE	Total
2540577	Core	1	0	2	2	40	60	100
Contact Classes: 15	Tutorial Classes: Nil	Practical Classes: 30				Total Classes: 30		

Prerequisites: **Programming for Problem Solving Lab**

Course Overview:

This course provides a foundation in object-oriented programming concepts and hands-on experience in using them. It introduces the concepts of abstraction and reusable code design via the object-oriented paradigm. Through a series of examples and exercises students gain coding skills and develop an understanding of professional programming practices. Mastering Java facilitate the learning of other technologies.

Course Objectives:

1. To write programs using abstract classes.
2. To write programs for solving real world problems using the java collection framework.
3. To write multithreaded programs.
4. To write GUI programs using swing controls in Java.
5. To introduce java compiler and eclipse platform.

Course Outcomes: After Completion of the Course, Students should be able to

1. Demonstrate proficiency with Java development environments, debugging tools, code refactoring features, and basic program execution workflows.
2. Develop Java applications using GUI components, layout managers, event handling mechanisms, and exception handling techniques.
3. Apply object-oriented programming concepts such as classes, inheritance, abstraction, polymorphism, and data structures within Java programs.
4. Implement multithreading concepts including thread creation, synchronization, inter-thread communication, and concurrent problem solving.
5. Design Java programs for file handling, directory traversal, data storage using collections, and user interaction through event-driven programming.

Note:

1. Use LINUX and MySQL for the Lab Experiments. Though not mandatory, encourage the use of the Eclipse platform.

2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.

List of Experiments:

1. Use Eclipse or Net bean platform and acquaint yourself with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto

fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.

2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.

3. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.

4. Write a Java program that implements a multi-thread application that has three threads. First thread generates a random integer every 1 second and if the value is even, the second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of the cube of the number.

5. Write a Java program for the following:

   Create a doubly linked list of elements.
   Delete a given element from the above List.
   Display the contents of the list after deletion.

6. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in the selected color. Initially, there is no message shown.

7. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

8. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.

9. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).

10. Write a Java program that loads names and phone numbers from a text file where the data is

organized as one line per record and each field in a record are separated by a tab (\t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint:use hash tables).

11. Write a Java program that correctly implements the producer – consumer problem using the concept of inter thread communication.

12. Write a Java program to list all the files in a directory including the files present in all its subdirectories.
13. Write programs to implement following using Collection Framework:
    a) to add, retrieve & remove element from Array List
    b) to Sort & reverse the Linked List elements
    c) to sort Array List using Comparable and Comparator
14. Write programs to implement following using Collection Framework:
    a) to copy elements from Hash Set to Array
    b) to remove duplicate key from hash table
    c) to iterate Tree Map

TEXT BOOKS:

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.
2. Thinking in Java, Bruce Eckel, Pearson Education.

REFERENCE BOOKS

1. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
2. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

ELECTRONIC RESOURCES:

1. https://docs.oracle.com/en/java/
2. https://www.geeksforgeeks.org/java
3. https://www.tutorialspoint.com/java/index.htm
4. https://www.coursera.org/courses?query=java
5. https://java-iitd.vlabs.ac.in/

MATERIALS ONLINE:

1. Lab manual
2. Open-ended experiments