20MCA132	OBJECT ORIENTED PROGRAMMING LAB	CATEGORY	L	T	P	CREDIT
		PRACTICAL	0	1	3	2

Preamble: This course enables the students to understand the concepts of object-oriented programming and to develop skills using these paradigms using Java.

Prerequisite: Knowledge of any programming language preferred.

Course Outcomes: After the completion of the course the student will be able to

CO 1	Understand object-oriented concepts and design classes and objects to solve problems
CO 2	Implement arrays and strings.
CO 3	Implement object-oriented concepts like inheritance, overloading and interfaces
CO 4	Implement packages, exception handling, multithreading and generic programming. Use java.util package and Collection framework
CO 5	Develop applications to handle events using applets
CO 6	Develop applications using files and networking concepts

Mapping of course outcomes 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
CO 1	2	2	2	2	3							
CO 2	**3**	2	2		3							
CO 3	**3**	2	2		3							
CO 4	**3**	2	2		3							
CO 5	**3**	3	3		3	2			3		3	
CO 6	**3**	3	3		3	2			3		3	

Assessment Pattern

Bloom's Category	Continuous Assessment Tests		End Semester Examination
	1	2	
Remember(K1)			
Understand(K2)			
Apply(K3)	10	10	10
Analyse(K4)	10	10	10
Evaluate(K5)	10	10	10
Create(K6)	20	20	20

Mark distribution

Total Marks	CIE	ESE	ESE Duration
100	50	50	3 hours

Continuous Internal Evaluation Pattern:

Maximum Marks: 50	
Attendance	15%
Maintenance of daily lab record and GitHub management	20%
Regular class viva	15%
Timely completion of day to day tasks	20%
Tests/Evaluation	30%

End Semester Examination Pattern:

Maximum Marks: 50			
Verification of Daily program record and Git Repository			5 marks
Viva			10 marks
Problem solving (Based on difficulty level, one or more questions may be given)	Flowchart / Algorithm / Structured description of problem to explain how the problem can be solved / Interface Design	15%	35 marks
	Program correctness	50%	
	Code efficiency	15%	
	Formatted output and Pushing to remote Git repository	20%	
Total Marks			50 marks

Course Level Assessment Questions

Course Outcome 1 (CO1):

1. Define a class ‘product’ with data members pcode, pname and price. Create 3 objects of the class and find the product having the lowest price.
2. Read 2 matrices from the console and perform matrix addition.

3. Add complex numbers
4. Read a matrix from the console and check whether it is symmetric or not.
5. Create CPU with attribute price. Create inner class Processor (no. of cores, manufacturer) and static nested class RAM (memory, manufacturer). Create an object of CPU and print information of Processor and RAM.

Course Outcome 2 (CO2)

1. Program to Sort strings
2. Search an element in an array.
3. Perform string manipulations
4. Program to create a class for Employee having attributes eNo, eName eSalary. Read n employ information and Search for an employee given eNo, using the concept of Array of Objects.

Course Outcome 3(CO3):

1. Area of different shapes using overloaded functions
2. Create a class 'Employee' with data members Empid, Name, Salary, Address and constructors to initialize the data members. Create another class 'Teacher' that inherit the properties of class employee and contain its own data members department, Subjects taught and constructors to initialize these data members and also include display function to display all the data members. Use array of objects to display details of N teachers.
3. Create a class 'Person' with data members Name, Gender, Address, Age and a constructor to initialize the data members and another class 'Employee' that inherits the properties of class Person and also contains its own data members like Empid, Company_name, Qualification, Salary and its own constructor. Create another class 'Teacher' that inherits the properties of class Employee and contains its own data members like Subject, Department, Teacherid and also contain constructors and methods to display the data members. Use array of objects to display details of N teachers.
4. Write a program has class Publisher, Book, Literature and Fiction. Read the information and print the details of books from either the category, using inheritance.
5. Create classes Student and Sports. Create another class Result inherited from Student and Sports. Display the academic and sports score of a student.

6. Create an interface having prototypes of functions area() and perimeter(). Create two classes Circle and Rectangle which implements the above interface. Create a menu driven program to find area and perimeter of objects.

7. Prepare bill with the given format using calculate method from interface.

Order No.

Date :

Product Id	Name	Quantity	unit price	Total
101	A	2	25	50
102	B	1	100	100
			Net. Amount	150

Course Outcome 4 (CO4):

1. Create a Graphics package that has classes and interfaces for figures Rectangle, Triangle, Square and Circle. Test the package by finding the area of these figures.
2. Create an Arithmetic package that has classes and interfaces for the 4 basic arithmetic operations. Test the package by implementing all operations on two given numbers
3. Write a user defined exception class to authenticate the user name and password.
4. Find the average of N positive integers, raising a user defined exception for each negative input.
5. Define 2 classes; one for generating multiplication table of 5 and other for displaying first N prime numbers. Implement using threads. (Thread class)
6. Define 2 classes; one for generating Fibonacci numbers and other for displaying even numbers in a given range. Implement using threads. (Runnable Interface)
7. Producer/Consumer using ITC
8. Program to create a generic stack and do the Push and Pop operations.
9. Using generic method perform Bubble sort.
10. Maintain a list of Strings using ArrayList from collection framework, perform built-in operations.
11. Program to remove all the elements from a linked list
12. Program to remove an object from the Stack when the position is passed as parameter
13. Program to demonstrate the creation of queue object using the PriorityQueue class
14. Program to demonstrate the addition and deletion of elements in deque
15. Program to demonstrate the creation of Set object using the LinkedHashset class
16. Write a Java program to compare two hash set

17. Program to demonstrate the working of Map interface by adding, changing and removing elements.
18. Program to Convert HashMap to TreeMap

Course Outcome 5 (CO5):

1. Program to draw Circle, Rectangle, Line in Applet.
2. Program to find maximum of three numbers using AWT.
3. Find the percentage of marks obtained by a student in 5 subjects. Display a happy face if he secures above 50% or a sad face if otherwise.
4. Using 2D graphics commands in an Applet, construct a house. On mouse click event, change the color of the door from blue to red.
5. Implement a simple calculator using AWT components.
6. Develop a program that has a Choice component which contains the names of shapes such as rectangle, triangle, square and circle. Draw the corresponding shapes for given parameters as per user's choice.
7. Develop a program to handle all mouse events and window events
8. Develop a program to handle Key events.

Course Outcome 6 (CO6):

1. Program to list the sub directories and files in a given directory and also search for a file name.
2. Write a program to write to a file, then read from the file and display the contents on the console.
3. Write a program to copy one file to another.
4. Write a program that reads from a file having integers. Copy even numbers and odd numbers to separate files.
5. Client server communication using Socket – TCP/IP
6. Client Server communication using DatagramSocket - UDP

Syllabus:

Classes and Objects, Constructors, Method Overloading, Access Modifiers, Arrays and Strings, Inheritance, Interfaces, Abstract classes, Dynamic Method Dispatch, String, Packages, Introduction to java.util, Collection framework, User defined packages, Exceptions, Multithreading, Applets, Graphics, File, Generic programming, Socket Programming

Reference Books

1. Herbert Schildt, "***Java The Complete Reference***", Seventh Edition, Tata McGraw-Hill Edition

2. C. Thomas Wu, "***An introduction to Object-oriented programming with Java***", Fourth Edition, Tata McGraw-Hill Publishing company Ltd.

3. Cay S. Horstmann and Gary Cornell, "***Core Java: Volume I – Fundamentals***", Eighth Edition, Sun Microsystems Press.

4. K. Arnold and J. Gosling, "***The JAVA programming language***", Third edition, Pearson Education.

5. Paul Deitel and Harvey Deitel, "***Java, How to Program***", Tenth Edition, Pearson Education

6. Rohit Khurana, "***Programming with Java***", Vikas Publishing, 2014.

7. Timothy Budd, "***Understanding Object-oriented programming with Java***", Updated Edition, Pearson Education.

8. Y. Daniel Liang, "***Introduction to Java programming***", Seventh Edition, Pearson Education.

Web Reference

- https://www.hackerrank.com/domains/java
- https://www.geeksforgeeks.org/java-tutorial/
- https://www.w3resource.com/java-tutorial/
- https://www.w3resource.com/java-exercises/
- https://nptel.ac.in/courses/106/105/106105191/
- https://nptel.ac.in/noc/courses/noc20/SEM1/noc20-cs08/
- https://www.coursera.org/learn/object-oriented-java
- https://www.edx.org/course/object-oriented-programming-in-java-2

Course Contents and Lab Schedule

Topic	No. of hours
1. Classes and Objects.	3
2. Constructors, Method Overloading, Access Modifiers	2
3. Arrays and Strings.	4
4. Inner class – static and non-static	2
5. Inheritance, Multiple inheritance - implementation using interfaces	3
6. Method overriding, Abstract classes, Dynamic Method Dispatch	3
7. Interfaces and Packages, StringBuffer class	3
8. Introduction to java.util package – Vector, Scanner, StringTokenizer	2
9. Collection framework – ArrayList, LinkedList, Stack, Queue, Set, Map	3
10. User defined packages	2
11. Exceptions – User defines exceptions	2
12. Multithreading – Thread class	2
13. Inter Thread Communication	2
14. Generic programming	2
15. Applets, Graphics – 2D	3
16. Event handling in Applet	3
17. File	3
18. Socket Programming	3