

**Model Question Paper**  
**Second Semester MCA Degree Examination**  
**Programming in Java**

**Time: 3 Hours**

**Max. Marks: 100**

*Note: 1. Answer any FIVE full questions, choosing ONE full question from each module.*

*2. M: Marks, L: RBT (Revised Bloom's Taxonomy) level, C: Course outcomes.*

<b>Module -1</b>			<b>M</b>	<b>L</b>	<b>C</b>
<b>Q1</b>	a.	Develop a Java program to accept N numbers from the user and display the sum of all numbers, the largest number, and the average using arrays and methods. Illustrate how objects and methods are used in the program.	10	L3	CO1
	b.	Design a Java class Employee with data members such as employee ID, name, and salary, and implement constructors, methods, and access specifiers to initialize employee details, calculate annual salary, and display the details for at least three employees.	10	L3	CO1
<b>OR</b>					
<b>Q2</b>	a.	Write a recursive Java program to find the factorial of a number and generate the Fibonacci series up to N terms, and write down the differences between iterative and recursive approaches.	10	L3	CO1
	b.	Create a Java program to perform matrix addition and matrix multiplication using two-dimensional arrays with separate methods for each operation and display the formatted output.	10	L3	CO1
<b>Module- 2</b>					
<b>Q3</b>	a.	Develop a Java program to store marks of students in an array, calculate total and average marks, assign grades using control statements, and display the topper's details.	10	L3	CO2
	b.	Write a Java program using command-line arguments to accept two integers and an operator (+, -, *, /), perform the specified arithmetic operation, and display the result while handling invalid operators appropriately.	10	L3	CO2
<b>OR</b>					
<b>Q4</b>	a.	Create a Java program to analyze a given string and display the number of vowels, consonants, digits, spaces, and special characters using methods from the String and StringBuffer classes.	10	L3	CO2
	b.	Write a Java application to accept a sentence from the user, reverse each word individually, and display the modified sentence using loops, arrays, and string manipulation methods.	10	L3	CO2
<b>Module - 3</b>					
<b>Q5</b>	a.	Design a Java program using inheritance with a superclass Person and subclasses Student and Teacher. Implement methods to display the details and show how	10	L3	CO3

		method overriding can be performed.			
	b.	Create a Java program to demonstrate runtime polymorphism by defining a base class Shape and derived classes Circle, Rectangle, and Triangle. Implement an area() method in each class using dynamic binding.	10	L3	CO3
<b>OR</b>					
Q6	a.	Write a Java program to demonstrate multiple inheritance using interfaces by creating interfaces for Sports and Academic and a class StudentResult implementing both interfaces to calculate and display the final result of a student.	10	L3	CO3
	b.	Develop a Java application using abstract classes and interfaces to model a banking system and implement operations such as deposit, withdraw, and balance inquiry to show abstraction and interface implementation.	10	L3	CO3
<b>Module - 4</b>					
Q7	a.	Write a Java program to accept two numbers from the user, perform division, and handle divide-by-zero and invalid input exceptions using try, catch, and finally blocks.	10	L3	CO4
	b.	Design a Java program to implement a custom exception called InvalidAgeException where the program accepts age from the user, throws the exception if the age is less than 18, and otherwise allows registration.	10	L3	CO4
<b>OR</b>					
Q8	a.	Create a Java application to perform banking transactions and handle exceptions for insufficient balance, invalid withdrawal amount, and invalid account number using throw and throws appropriately.	10	L3	CO4
	b.	Develop a Java program to demonstrate nested try-catch blocks and rethrowing of exceptions while displaying appropriate messages for different exception types.	10	L3	CO4
<b>Module - 5</b>					
Q9	a.	Write a Java program to create three threads for displaying even numbers, odd numbers, and prime numbers within a given range and demonstrate thread creation using both the Thread class and the Runnable interface.	10	L2	CO5
	b.	Develop a Java program to demonstrate thread synchronization using a shared resource and illustrate how synchronization avoids data inconsistency in a banking transaction system.	10	L3	CO5
<b>OR</b>					
Q10	a.	Create a Java application using multithreading wherein one thread generates numbers, another thread computes square values, and a third thread computes cube values and displays the outputs concurrently.	10	L2	CO5
	b.	Write a Java program to demonstrate inter-thread communication using wait(), notify(), and notifyAll() methods by simulating a producer-consumer problem using threads.	10	L3	CO5

\_\*\*\*\*\*\_