



Hindusthan College of Engineering and Technology

An Autonomous Institution Affiliated to Anna University | Approved by AICTE, New Delhi
Accredited with 'A' Grade by NAAC | Accredited by NBA (ECE, MECH, EEE, IT & CSE)



Valley Campus, Pollachi Highway, Coimbatore 641 032. | www.hicet.ac.in

Course Information Sheet (CIS)

1. **Academic Year** : 2023 – 24 ODD Semester
2. **Name of Course Coordinator** : Dr RAMYA DEVI M
3. **Department** : Computer Science and Engineering
4. **Programme** : B.E CSE
5. **Class and semester** : II and III
6. **Course code and title** : 22CS3251/ Object Oriented Programming using Java
7. **Regulations** : R2022
8. **Course Category** : PC
9. **Contact hours** : 60
10. **Type of course** : Theory + Lab Component
11. **Credit** : 3
12. **Course Attainment level** :
 - Level I : 72 - 81%,
 - Level II : 82 - 91%
 - Level III : >91%

13. **Course pre-requisites** :

13. **Course Learning Objectives (CLO)** :

- 1) To conceptualize the basic Java based code for solving low complexity problems.
- 2) To study the Object-Oriented Features in Java for solving medium complexity problems.
- 3) To understand the polymorphism, abstraction, inheritance and interface concepts
- 4) To develop Java Applications using Multithreading, Packages and Collections.
- 5) To familiarize the concepts of data structures using java.

14. **Course Outcomes (COs)** :

Upon successful completion of this course, the student will be able to:

- CO1 - Apply Java based code for solving low complexity problems
- CO2 - Utilize Object Oriented Features in Java for solving medium complexity problems
- CO3 - Exploit polymorphism, abstraction, inheritance and interfaces in Java.
- CO4 - Develop Packages, Collections and Multi-Threaded Java Applications.
- CO5 - Utilize appropriate Java Classes to solve data structure-based problems

15. Syllabus:

UNIT I - INTRODUCTION TO JAVA and OOP		7+2
CO1	<p>Object Oriented Programming- First java program Hello World- JVM architecture JDK-JRE- identifiers-variables-comments-command line arguments-operators in java-control structures- Series and patterns-strings -immutable string-string operations-String Buffer class-StringBuilder class-String Joiner-String Tokenizer.</p> <p>Programs:</p> <ol style="list-style-type: none"> Write a program to generate the given pattern. If $n = 5$, <pre> 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1 </pre> Petya started to attend programming lessons. On the first lesson, his task was to write a simple program. The program was supposed to do the following: in the given string, consisting of uppercase and lowercase letters, it: <ul style="list-style-type: none"> Deletes all the vowels. Inserts a character "." before each consonant. Replaces all uppercase consonants with corresponding lowercase ones. 	

UNIT II - ARRAYS, CLASS AND INHERITANCE		7+2
CO2	<p>Introduction to Arrays in java-Arrays class-declaration and initialization of an array-2D array declaration and initialization -Multi-dimensional array-Classes and objects-naming convention in java-methods-access modifiers-constructors- copy constructors -singleton class- object class-inner class-abstract class- Throwable class- types of inner class- static and non-static nested class-Inheritance-Types of inheritance, Difference between inheritance in C++ and java</p> <p>Programs:</p> <ol style="list-style-type: none"> A magic square is an arrangement of numbers (usually integers) in a square grid, where the numbers in each row, and in each column, and the numbers in the forward and backward main diagonals, all add up to the same number. Write a program to find whether a given matrix is a magic square or not. <p>Input Format: The input consists of $(n*n+1)$ integers. The first integer corresponds to the number of rows/columns in the matrix. The remaining integers correspond to the elements in the matrix. The elements are read in row wise order, first row first, then second row and so on. Assume that the maximum value of m and n is 5.</p> A company maintains a database that has the details of all the employees. There are two levels of employees where level 1 is the top management having salary more than 100 dollars and level 2 is the staffs who are getting a salary less than 100 dollars. Create a class named Employee with empId and salary as attributes. Create another class empLevel that extends employee and categorizes the employee into various levels. <p>Input Format: The input should contain only the employee id and salary of the employee separated by space. Employee id should be of integer type and salary float type.</p> 	

Output Format: The output of the program must display the employee id, salary, and level of the employee one below the other in the same order.

UNIT III - ABSTRACTION, POLYMORPHISM AND INTERFACES

7+2

CO3 Abstraction in java -abstract class-control abstraction-data hiding vs abstraction, encapsulation- Runtime polymorphism-compile time vs run time polymorphism, constructor overloading- constructor chaining-private constructors and singleton class- Methods-different method calls-method overriding-method overloading, method overloading vs method overriding. Interfaces-interfaces and inheritance-class vs interface-Functional interface-nested interface-Marker interface-Comparator interface.

Programs:

1. Write a Java program to demonstrate method overriding and dynamic method dispatch. Create a class named 'Animal' with a method named 'Print' that prints "Animal" to the console. Next, create two subclasses named 'Dog' and 'Cat' that inherit from the 'Animal' class and override the 'Print' method to print "Dog" and "Cat" to the console, respectively. In the 'Main' class, declare a variable 'a' of type 'Animal' and initialize it with a new object of the 'Dog' class. Call the 'Print' method on the 'a' variable and observe that "Dog" is printed to the console. Next, set the 'a' variable to a new object of the 'Cat' class and again call the 'Print' method. Observe that "Cat" is printed to the console this time.

Input Format No console input.

Output Format Print the String from subclass named Dog and Cat in separate lines.

2. Write a java program to create an interface called "ShapeCalculator" that has a method called "calc(int n)". Then, create two classes called "Square" and "Circle" that implement the "ShapeCalculator" interface and implement the "calc(int n)" method. Your program should calculate the area and perimeter of both squares and circles.

Input Format: The input to your program will be a single integer that represents the side of the square and the radius of the circle.

UNIT IV- MULTITHREADING, PACKAGES AND COLLECTIONS

7+2

CO4 Threads-lifecycle and stages of a Thread-Thread priority-main Thread-Runnable interface-naming thread-start () method-Java packages-built in packages-user defined packages-Collections-List interface-Queue interface-Map interface-Set-Iterator, Comparator-JDBC-connectivity with JDBC-DriverManager-Statement-JDBC Exceptions.

Programs:

1. Write a java program that implements a multi-thread application that has three threads.
 - First thread generates random integer every 1 second and if the value is even,
 - second thread computes the square of the number and prints. If the value is odd,
 - the third thread will print the value of cube of the number.
2. Simple OPAC system for library using event-driven paradigms with JDBC

UNIT V - DATA STRUCTURE IN JAVA

CO5	<p>Arrays-Linked list- implementation of linked list-stack-implementation of stack operation. Queue-implementation of queue operations-Tree-Binary search tree implementation- Graphs-shortest path algorithm using java.</p> <p>Programs:</p> <ol style="list-style-type: none"> Write a program to evaluate an expression entered in "postfix" form using stack concept. Write a program to implement single source shortest path algorithm.
-----	--

Total Instructional Hours - 60

16. Text books and Reference books:

TEXT BOOKS

- Herbert Schildt, "JAVA The Complete Reference", 10th Edition, McGraw Hill Education, 2017.
- Cay S. Horstman and Gary Cornell, "Core Java Volume I—Fundamentals", 11th Edition, Prentice Hall, 2018.

REFERENCE BOOKS

- Cay Horstman, "Big Java: Early Objects", 6th Edition, Wiley Publications, 2016.
- Ken Arnold, James Gosling, and David Holmes, "The Java Programming Language", 4th edition, Addison Wesley, 2005

17. Course plan:

S.No	Name of the Topic	No of Hours	Cum. Hours	Teaching Methods	Teaching Aids	Text/Reference books
UNIT I - INTRODUCTION TO JAVA and OOP						
GROUP I						
1	Object Oriented Programming	1	1	Lecture	PPT, Video	T1, R2
2	First java program Hello World	1	2			
3	JVM architecture-JDK-JRE	1	3			
4	Identifiers - variables - comments - Command line arguments	1	4			
GROUP II						
5	Operators	1	5	Lecture Flipped Class	PPT, Video, Quiz	T1, T2
6	Control structures- Series and patterns	1	6			
7	Strings -immutable string - string operations	1	7			
8	String Buffer class -StringBuilder class - String Joiner - String Tokenizer	1	8			
GROUP III						
9	<p>Write a program to generate the given pattern. If n = 5,</p> <pre> 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1 </pre>	2	10	Lecture Flipped Class	PPT	T1, T2

10	<p>Petya started to attend programming lessons. On the first lesson, his task was to write a simple program. The program was supposed to do the following: in the given string, consisting of uppercase and lowercase letters, it:</p> <ul style="list-style-type: none"> ▪ Deletes all the vowels. ▪ Inserts a character "." before each consonant. ▪ Replaces all uppercase consonants with corresponding lowercase ones. 	2	12			
Scheduled completion of Unit I: 12 hours						
UNIT II – ARRAYS, CLASS AND INHERITANCE						
GROUP I						
11	Introduction to Arrays in java-Arrays class-declaration and initialization of an array	1	13	Lecture Flipped Class	PPT, Video, Quiz	T1, T2
12	2D array declaration and initialization - multi-dimensional array	1	14			
13	Classes and objects - naming convention in java methods - access modifiers	1	15			
14	Constructors- copy constructors - singleton class- object class	1	16			
GROUP II						
15	Inner class	1	17	Lecture Flipped Class	PPT, Video, Quiz, Assignm ent	T1, T2
16	Abstract class- Throwable class- types of inner class- static and non-static nested class	1	18			
17	Inheritance-Types of inheritance	1	19			
18	Difference between inheritance in C++ and java	1	20			
GROUP III						
19	A magic square is an arrangement of numbers (usually integers) in a square grid, where the numbers in each row, and in each column, and the numbers in the forward and backward main diagonals, all add up to the same number. Write a program to find whether a given matrix is a magic square or not. Input Format: The input consists of (n*n+1) integers. The first integer corresponds to the number of rows/columns in the matrix. The remaining	2	22	Lecture Flipped Class	PPT	T1, T2

	integers correspond to the elements in the matrix. The elements are read in row wise order, first row first, then second row and so on. Assume that the maximum value of m and n is 5.					
20	A company maintains a database that has the details of all the employees. There are two levels of employees where level 1 is the top management having salary more than 100 dollars and level 2 is the staffs who are getting a salary less than 100 dollars. Create a class named Employee with empId and salary as attributes. Create another class empLevel that extends employee and categorizes the employee into various levels. Input Format: The input should contain only the employee id and salary of the employee separated by space. Employee id should be of integer type and salary float type.	2	24	Lecture Flipped Class	PPT	T1, T2
Scheduled completion of Unit II: 12 hours						
UNIT III - ABSTRACTION, POLYMORPHISM, AND INTERFACES						
GROUP I						
21	Abstraction in java -abstract class-control abstraction-data hiding vs abstraction - encapsulation	1	25	Lecture, Flipped Class	PPT, Video, Seminar	T1
22	Runtime polymorphism-compile time vs run time polymorphism	1	26			
23	Constructor overloading - constructor chaining-private constructors and singleton class	1	27			
24	Methods - different method calls - method overriding - method overloading - Method overloading vs method overriding	1	28			
GROUP II						
25	Interfaces-interfaces and inheritance	1	29	Lecture, Flipped Class	PPT, Video, Quiz, Assignment	T1
26	Class vs interface-Functional interface	1	30			
27	Nested interface-Marker interface	1	31			
28	Comparator interface	1	32			
GROUP III						

29	Write a java program to demonstrate method overriding and dynamic method dispatch. Create a class named 'Animal' with a method named 'Print' that prints "Animal" to the console. Next, create two subclasses named 'Dog' and 'Cat' that inherit from the 'Animal' class and override the 'Print' method to print "Dog" and "Cat" to the console, respectively. In the 'Main' class, declare a variable 'a' of type 'Animal' and initialize it with a new object of the 'Dog' class. Call the 'Print' method on the 'a' variable and observe that "Dog" is printed to the console. Next, set the 'a' variable to a new object of the 'Cat' class and again call the 'Print' method. Observe that "Cat" is printed to the console this time. Input Format No console input. Output Format Print the String from subclass named Dog and Cat in separate lines.	2	34	Lecture Flipped Class	PPT	T1, T2
30	Write a java program to create an interface called "ShapeCalculator" that has a method called "calc(int n)". Then, create two classes called "Square" and "Circle" that implement the "ShapeCalculator" interface and implement the "calc(int n)" method. Your program should calculate the area and perimeter of both squares and circles. Input Format: The input to your program will be a single integer that represents the side of the square and the radius of the circle.	2	36	Lecture Flipped Class	PPT	T1, T2

Scheduled completion of Unit III: 12 hours

UNIT IV - MULTITHREADING, PACKAGES AND COLLECTIONS

GROUP I

31	Threads-lifecycle and stages of a Thread	1	37	Flipped class room, Lecture	PPT, Video, Quiz	T1, R1
32	Thread priority	1	38			
33	main Thread-Runnable interface-naming thread-start () method	1	39			

GROUP II

34	Java packages-built in packages	1	40	Lecture Flipped Class	PPT, Video, Quiz,	T1, T2
35	user defined packages	1	41			

36	Collections - List interface - Queue interface - Map interface - Set	1	42		Assignment	
37	Iterator Comparator	1	43			
38	JDBC - connectivity with JDBC-DriverManager-Statement-JDBC Exceptions.	1	44			
GROUP III						
39	Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.	2	46	Lecture Flipped Class	PPT	T1, T2
40	Simple OPAC system for library using event-driven paradigms with JDBC	2	48			
Scheduled completion of Unit IV: 12 hours						
UNIT V - DATA STRUCTURE IN JAVA						
GROUP I						
41	Arrays - Linked list - implementation of linked list	2	50	Lecture	PPT, Video, Seminar	T1, R3
42	Stack - implementation of stack operations	2	52			
43	Queue-implementation of queue operations-	1	53			
GROUP II						
44	Tree-Binary search tree implementation- algorithm	1	54	Lecture	PPT, Video	T1, T2
45	Graphs-shortest path using java.	2	56			
GROUP III						
46	Write a program to evaluate an expression entered in "postfix" form using stack concept.	2	58	Lecture Flipped Class	PPT	T1, T2
47	Write a program to implement single source shortest path algorithm.	2	60			
Scheduled completion of Unit V: 12 hours						

18. Weightage of unit contents:

- Factors considered,
- F1 - Number of periods allotted for teaching the unit and weightage per hour is equal 1.
 - F2 - Usefulness of the content matter of the unit in the students' learning point of view and its weightage equal to 1 if useful, otherwise zero.
 - F3 - Usefulness of the content matter of the unit in understanding other units of the same subject and its weightage equal to 1 if useful, otherwise zero.

F4- Usefulness of the content matter of the unit in understanding other subjects prescribed for the programme and its weightage equal to 1 if useful, otherwise zero.

Topic	F ₁	F ₂	F ₃	F ₄	A ₁ (Weightage)	A ₂ (%)
UNIT I - INTRODUCTION TO JAVA and OOP						
Object Oriented Programming		1	1	1	34	20.36
First java program Hello World		1				
JVM architecture-JDK-JRE		1				
Identifiers - variables - comments - Command line arguments		1	1	1		
Operators		1	1	1		
Control structures- Series and patterns		1	1	1		
Strings -immutable string - string operations		1	1	1		
String Buffer class -StringBuilder class - String Joiner - String Tokenizer.		1	1	1		
Write a program to generate the given pattern. 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1	12	1				
Petya started to attend programming lessons. On the first lesson, his task was to write a simple program. The program was supposed to do the following: in the given string, consisting of uppercase and lowercase letters, it: Deletes all the vowels. Inserts a character "." before each consonant. Replaces all uppercase consonants with corresponding lowercase ones.		1				
UNIT II ARRAYS, CLASS AND INHERITANCE						
Introduction to Arrays in java-Arrays class-declaration and initialization of an array		1	1	1	36	21.56
2D array declaration and initialization -multi-dimensional array		1	1	1		
Classes and objects - naming convention in java methods - access modifiers		1	1	1		
Constructors- copy constructors -singleton class- object class		1	1	1		
Inner class		1	1	1		
Abstract class- Throwable class- types of inner class- static and non-static nested class	12	1	1	1		
Inheritance-Types of inheritance		1	1	1		
Difference between inheritance in C++ and java		1				
A magic square is an arrangement of numbers (usually integers) in a square grid, where the numbers in each row, and in each column, and the numbers in the forward and backward main diagonals, all add up to the same number. Write a program to find whether a given matrix is a magic		1				

Wf. calle called called "Shan"

<p>square or not. Input Format: The input consists of $(n*n+1)$ integers. The first integer corresponds to the number of rows/columns in the matrix. The remaining integers correspond to the elements in the matrix. The elements are read in row wise order, first row first, then second row and so on. Assume that the maximum value of m and n is 5.</p>					
<p>A company maintains a database that has the details of all the employees. There are two levels of employees where level 1 is the top management having salary more than 100 dollars and level 2 is the staffs who are getting a salary less than 100 dollars. Create a class named Employee with empId and salary as attributes. Create another class empLevel that extends employee and categorizes the employee into various levels. Input Format: The input should contain only the employee id and salary of the employee separated by space. Employee id should be of integer type and salary float type.</p>	1				
UNIT III ABSTRACTION, POLYMORPHISM AND INTERFACES					
Abstraction in java -abstract class-control abstraction-data hiding vs abstraction - encapsulation	1	1	1		
Runtime polymorphism-compile time vs run time polymorphism	1	1	1		
Constructor overloading - constructor chaining-private constructors and singleton class	1	1	1		
Methods - different method calls -method overriding - method overloading - Method overloading vs method overriding	1	1	1		
Interfaces-interfaces and inheritance	1	1	1		
Class vs interface-Functional interface	1		1		
Nested interface-Marker interface	1		1		
Comparator interface	1	1	1		
<p>Write a Java program to demonstrate method overriding and dynamic method dispatch. Create a class named 'Animal' with a method named 'Print' that prints "Animal" to the console. Next, create two subclasses named 'Dog' and 'Cat' that inherit from the 'Animal' class and override the 'Print' method to print "Dog" and "Cat" to the console, respectively. In the 'Main' class, declare a variable 'a' of type 'Animal' and initialize it with a new object of the 'Dog' class. Call the 'Print' method on the 'a' variable and observe that "Dog" is printed to the console. Next, set the 'a' variable to a new object of the 'Cat' class and again call the 'Print' method. Observe that "Cat" is printed to the console this time. Input Format No console input. Output Format Print the String from subclass named Dog and Cat in separate lines.</p>	12	1			
				36	21.56

Write a java program to create an interface called "ShapeCalculator" that has a method called "calc(int n)". Then, create two classes called "Square" and "Circle" that implement the "ShapeCalculator" interface and implement the "calc(int n)" method. Your program should calculate the area and perimeter of both squares and circles. Input Format: The input to your program will be a single integer that represents the side of the square and the radius of the circle.							
UNIT IV - MULTITHREADING, PACKAGES AND COLLECTIONS							
Threads-lifecycle and stages of a Thread		1		1			
Thread priority		1		1			
main Thread-Runnable interface-naming thread-start () method		1		1			
Java packages-built in packages		1	1	1			
User defined packages		1	1	1			
Collections - List interface - Queue interface - Map interface - Set		1	1	1			
Iterator Comparator		1	1	1			
JDBC - connectivity with JDBC-DriverManager-Statement-JDBC Exceptions.	12	1	1	1			
Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.		1					
Simple OPAC system for library using event-driven paradigms with JDBC		1					
UNIT V DATA STRUCTURE IN JAVA							
Arrays - Linked list - implementation of linked list		1		1			
Stack - implementation of stack operations		1		1			
Queue-implementation of queue operations		1		1			
Tree-Binary search tree implementation-algorithm	12	1		1			
Graphs-shortest path using java.		1		1			
Write a program to evaluate an expression entered in "postfix" form using stack concept.		1		1			
Write a program to implement single source shortest path algorithm.		1		1			
					Total	167	100%
A₁ - Total weightage							
A₂ - % of Weightage							

19. Mapping syllabus with Bloom's Taxonomy LOT and HOT:

Lower Order Thinking		
R	Remembering	Students are expected to Recall the information through Recognizing, listing, describing, retrieving, naming, finding
U	Understanding	Students are expected to Explain an ideas or concepts through Interpreting, summarizing, paraphrasing, classifying, explaining
Ap	Applying	Students are expected to Use the information in another familiar situation through Implementing, carrying out, using, executing
Higher Order Thinking		
A	Analyzing	Students are expected to Break the information into parts to explore understandings and relationships through Comparing, organizing, deconstructing, interrogating, finding
E	Evaluating	Students are expected to Evaluate the Justifying a decision or course of action through Checking, hypothesizing, experimenting, judging
C	Creating	Students are expected to Generate new ideas, products, or ways of viewing things through Designing, constructing, planning, producing, inventing.

UNIT I – INTRODUCTION TO JAVA and OOP			
Sl.No	Name of the Topic	Process verb	Types of thinking
1	Object Oriented Programming, First java program Hello World, JVM architecture- JDK-JRE, Identifiers – variables – comments - Command line arguments	Explain, Compare, Infer	Understand CO1
2	Operators, Control structures- Series and patterns	Analyze, Evaluate, Create, Apply	Create CO1
3	Strings -immutable string – string operations, String Buffer class - StringBuilder class - String Joiner - String Tokenizer	Explain, Compare, Infer	Understand CO1
4	Write a program to generate the given pattern. If n = 5, 1 1 1 1 2 1 1 3 3 1 1 4 6 4 1 Petya started to attend programming lessons. On the first lesson, his task was to write a simple program. The program was supposed to do the following: in the given string, consisting of uppercase and lowercase letters, it: <ul style="list-style-type: none"> ▪ Deletes all the vowels. ▪ Inserts a character "." before each consonant. ▪ Replaces all uppercase consonants with corresponding lowercase 	Create	Create CO1

guit

ones.							
	R	U	Ap	A	E	C	Total
Type of thinking in Nos	-	2	-	-	-	2	4
Weightage,%	-	10.18	-	-	-	10.18	20.36

UNIT II ARRAYS, CLASS AND INHERITANCE

Sl.No	Name of the Topic	Process verb	Types of thinking
1	Introduction to Arrays in java-Arrays class-declaration and initialization of an array, 2D array declaration and initialization -multi-dimensional array,	Explain, Compare, Infer, Apply	Apply CO2
2	Classes and objects - naming convention in java methods - access modifiers, Constructors- copy constructors -singleton class- object class, Inner class, Abstract class- Throwable class- types of inner class-static and non-static nested class, Inheritance-Types of inheritance	Exemplify, Apply, Create, Develop	Create CO2
3	Difference between inheritance in C++ and java	Compare	Understand CO2
4	A magic square is an arrangement of numbers (usually integers) in a square grid, where the numbers in each row, and in each column, and the numbers in the forward and backward main diagonals, all add up to the same number. Write a program to find whether a given matrix is a magic square or not. Input Format: The input consists of (n*n+1) integers. The first integer corresponds to the number of rows/columns in the matrix. The remaining integers correspond to the elements in the matrix. The elements are read in row wise order, first row first, then second row and so on. Assume that the maximum value of m and n is 5. A company maintains a database that has the details of all the employees. There are two levels of employees where level 1 is the top management having salary more than 100 dollars and level 2 is the staffs who are getting a salary less than 100 dollars. Create a class named Employee with empId and salary as attributes. Create another class empLevel that extends employee and categorizes the employee into various levels. Input Format: The input should contain only the employee id and salary of the employee separated by space. Employee id should be of integer type and salary float type.	Create	Create CO2

	R	U	Ap	A	E	C	Total
--	----------	----------	-----------	----------	----------	----------	--------------

Type of thinking in Nos	-	1	1	-	-	2	4
Weightage,%	-	5.39	5.39	-	-	10.78	21.56
UNIT III ABSTRACTION, POLYMORPHISM AND INTERFACES							
Sl.No	Name of the Topic	Process verb			Types of thinking		
1	Abstraction in java -abstract class-control abstraction-data hiding vs abstraction - encapsulation, Runtime polymorphism-compile time vs run time polymorphism, Constructor overloading - constructor chaining-private constructors and singleton class, Methods - different method calls - method overriding - method overloading - Method overloading vs method overriding, Interfaces-interfaces and inheritance	Differentiate, Implement, Apply, Create			Create CO3		
2	Class vs interface	Compare			Understand CO3		
3	Functional interface, Nested interface-Marker interface, Comparator interface. Write a Java program to demonstrate method overriding and dynamic method dispatch. Create a class named 'Animal' with a method named 'Print' that prints "Animal" to the console. Next, create two subclasses named 'Dog' and 'Cat' that inherit from the 'Animal' class and override the 'Print' method to print "Dog" and "Cat" to the console, respectively. In the 'Main' class, declare a variable 'a' of type 'Animal' and initialize it with a new object of the 'Dog' class. Call the 'Print' method on the 'a' variable and observe that "Dog" is printed to the console. Next, set the 'a' variable to a new object of the 'Cat' class and again call the 'Print' method. Observe that "Cat" is printed to the console this time. Input Format No console input. Output Format Print the String from subclass named Dog and Cat in seperate lines. Write a java program to create an interface called "ShapeCalculator" that has a method called "calc(int n)". Then, create two classes called "Square" and "Circle" that implement the "ShapeCalculator" interface and implement the "calc(int n)" method. Your program should calculate the area and perimeter of both squares and circles. Input Format: The input to your program will be a single integer that represents the side of the square and the radius of the circle.	Implement, Apply, Create, Develop, Demonstrate			Create CO3		
		R	U	Ap	A	E	C
Type of thinking in Nos		-	1	-	-	-	2
							Total
							3

21.56
4

Weightage,%		-	7.19	-	-	-	14.37	21.56
UNIT IV MULTITHREADING, PACKAGES AND COLLECTIONS								
Sl.No	Name of the Topic	Process verb			Types of thinking			
1	Threads-lifecycle and stages of a Thread, Thread priority, main Thread-Runnable interface-naming thread-start () method	Explain, Compare, Discuss, Apply			Apply C04			
2	Java packages-built in packages, user defined packages	Differentiate, Implement, Apply, Create			Create C04			
3	<p>Collections - List interface - Queue interface - Map interface - Set, Iterator Comparator, JDBC - connectivity with JDBC-DriverManager-Statement-JDBC Exceptions. Write a java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.</p> <p>Simple OPAC system for library using event-driven paradigms with JDBC</p>	Explain, Compare, Infer, Apply			Apply C04			
		R	U	Ap	A	E	C	Total
Type of thinking in Nos		-	-	2	-	-	1	3
Weightage,%		-	-	13.96	-	-	6.98	20.95

UNIT V DATA STRUCTURE IN JAVA

Sl.No	Name of the Topic	Process verb			Types of thinking			
1	Arrays - Linked list - implementation of linked list, Stack - implementation of stack operations, Queue-implementation of queue operations- Tree-Binary search tree implementation- algorithm,	Explain, Compare, Discuss, Apply			Apply C05			
2	Graphs-shortest path using java.	Originate			Create C05			
3	<p>Write a program to evaluate an expression entered in "postfix" form using stack concept.</p> <p>Write a program to implement single source shortest path algorithm.</p>	Explain, Compare, Discuss, Apply			Apply C04			
		R	U	Ap	A	E	C	Total
Type of thinking in Nos		-	-	2	-	-	1	3
Weightage,%		-	-	10.38	-	-	5.19	15.57

	R	U	AP	A	E	C	TOTAL
UNIT 1	-	10.18	-	-	-	10.18	20.36
UNIT 2	-	5.39	5.39	-	-	10.78	21.56
UNIT 3	-	7.19	-	-	-	14.37	21.56

UNIT 4	-	-	13.96	-	-	6.98	20.5
UNIT 5	-	-	10.38	-	-	5.19	15.57
TOTAL	-	22.76	29.73	-	-	47.50	100
Lower Order Thinking						52.49	
Higher Order Thinking						47.50	

20. Mapping course outcome with Bloom's Taxonomy LOT and HOT:

	R	U	Ap	A	E	C
C01		✓✓				✓✓
C02		✓✓				✓✓
C03		✓				✓
C04			✓✓			✓
C05			✓✓			✓

21. Mapping Course Outcome (CO) with Program Outcomes (PO) and Program Specific Outcomes (PSO):

	Program Outcomes	Descriptions
P01	Engineering knowledge	Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
P02	Problem analysis	Identify, formulate, research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
P03	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.
P04	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.
P05	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.
P06	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
P07	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

2095

		development.
PO8	Ethics	Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
PO9	Individual and team work	Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
PO10	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.
PO11	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
PO12	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	An ability to apply, design and develop principles of software engineering, networking, and database concepts for computer-based systems in solving engineering problems.
PSO2	An ability to understand, design and code engineering problems using programming skills.

PO&PSO →	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	3	3	3							3		1	3
CO2	3	3	3	3				1	1	1	3		1	3
CO3	3	3	3	3				1			3		1	3
CO4	3	3	3	3				1			3		1	3
CO5	3	3	3	3					1	1	3		2	3

3	High	2	Moderate	1	Low
---	------	---	----------	---	-----

22. Mapping with Programme Educational Objectives (PEOs):

Programme Educational Objectives:

PEO1 : To acquire knowledge in the latest technologies and innovations and an ability to identify, analyze, and solve problems in Computer Engineering.

PEO2 : To be capable of modeling, designing, implementing, and verifying a computing system to meet specified requirements for the benefit of society.

PEO3 : To possess critical thinking, communication skills, teamwork, leadership skills, and ethical behavior necessary to function productively and professionally.

Course	PEO1	PEO2	PEO3
---------------	-------------	-------------	-------------

Machine Learning Techniques	3	3	3
-----------------------------	---	---	---

3	High	2	Moderate	1	Low
---	------	---	----------	---	-----

23. Course assessment: (Direct Assessment Method)

Internal test: 15 Marks

Objective	To Identify What Students Have Learned and also to identify students strength and weakness
Product	Answer scripts
Frequency	Monthly
Format	Part -A 6 x 2 = 12 Marks Part - B 2 x 14 = 28 Marks Part -C 1 x 10 = 10 Marks Total marks = 50 Duration : 1 Hour and 30 Minutes
Evaluation	Based on answer given in the scripts
criteria	Pass mark - 50% Minimum pass percentage: 50% If not, remedial action will be taken.

Assignment: 5 marks

Objective	To enhance students' understanding of a particular reading
Product	Hand written assignment/tutorial sheets
Frequency	After completing one unit
Format	Important questions from each units
Evaluation	Based on rubrics
Criteria	No. of assignments: 3 Submit on or before the due date

Attendance: 5 marks

Objective	To make all students to attend the class throughout the course
Product	Record of class work
Frequency	All working days
Format	Record of class work format
Evaluation	Based on attendance earned by the students

Criteria	Marks will be awarded according to attendance percentage of students.
	91 and above 5
	86 - 90 4
	81 - 85 3

75 – 80	2
Less than 75	0

End semester exam: 75 marks

Objective	To assess the each student's knowledge of the course
Product	Result analysis
Frequency	Every Semester
Format	Part -A 10 x 2= 20 marks Part -B 5 x 14= 70 marks Part - C 1 x 10 = 10 Marks Total marks= 100 Duration : 3 Hours
Evaluation	Based on answer given in the scripts
Criteria	Minimum pass percentage: 50% If not, remedial action will be taken.


24. Course assessment: (Indirect Assessment Method)

Course Exit Survey: Course Exit Survey consists of few critical questions that evaluate the level of students' satisfaction level with curriculum and course being taught.

Prepared by,


Course Coordinator
(Dr Ramya Devi M)

Checked by,


Head of the Department
(Dr Shankar S)


Dean (Academics)

Approved by,


PRINCIPAL