MANIPAL INSTITUTE OF TECHNOLOGY
MANIPAL
(A constituent unit of MAHE, Manipal)

COURSE PLAN FOR ODD SEM 2022-23

Department	:	Data Science & Computer Applications
Course Name & code	:	Object Oriented Programming Lab & DSE 2161
Semester & branch	:	III Semester B.Tech & DSE
Name of the faculty	:	Dr. Vidya Rao and Dr. Nandini Kumari

No of contact hours/week:

L	T	P	C
0	0	3	1

ASSESSMENT PLAN

Course Outcomes (COs)

	At the end of this course, the student should be able to:	**No. of Contact Hours**	**Marks**
CO1:	Implement object-oriented programming concepts in Java	6	18
CO2:	Write, compile and debug programs written in Java	6	18
CO3:	Learn the usage of abstract classes and interfaces in a Java program	12	36
CO4:	Implement exception handling, multithreading, string programs, input/output streams in Java	6	18
CO5:	Develop simple GUI applications using Java swings	3	10
	Total	33	100

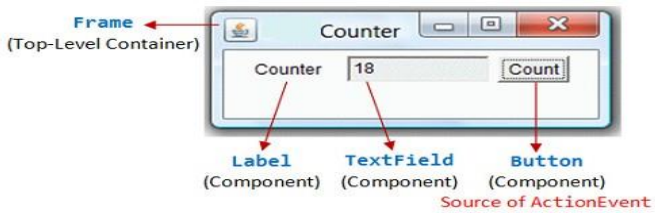
1. Continuous Evaluation	60%
Record: 4M, Viva: 6M and Execution: 10M Internal Marks: 3 * 20 = 60 Marks	
2. Lab Examination	40%
• End Semester Lab evaluation: 40 Marks, write Up: 15 Marks, execution: 25 Marks, • Total: 15+25 = 40 Marks Examination of 2 hours duration (Max. Marks: 40)	

Course Plan

L. No.	Topics	Course Outcome Addressed
L1	**Java Features & Simple Programs Using Control Structures** 1. Create a class name "GreatestNumber" and define a method that displays the greatest among the three-given number. 2. A Taxi service offers a new service based on travel distance. Write a Java program to calculate the total distance travelled by considering following charges. First 5 KM = INR 10, Next 15 KM = INR 8, Next 25 KM = INR 5. 3. For given a 9-digit registration number of DSE student, identify the year of joining. Assuming the first two digits specify the year of joining. 4. For a given date of birth of a person, calculate the date of retirement by taking years of service as input.	CO1
L2	**Java Data Types, Type Conversions, Operators** 1. Write a Java program to read an int number, double number and a char from the keyboard and perform the following conversion: int to byte, char to int, double to byte, double to int. 2. Add two numbers using bitwise operator and check is the output is a even or odd number. [Hint: use left shift and right shift bitwise operators] **3.** By considering a string and a number, perform swap of string to int and int to string. [Hint: a = "hello", b = 123, ==> (swap to) a = 123, b = "hello"] **Practice Questions:** 1. Write a Java program to accept number of hours worked, hourly rate and calculates the salary for an employee according to the following criteria: The company pays straight time for the first 40 hours worked by each employee and time and a half for all hours worked in excess of 40 hours. 2. Perform multiplication and division by 2^n using shift operators.	CO1
L3	**Control Statements:** 1. Write a Java Program to check if the person is eligible for voting. Consider three test cases. 2. For a given rectangle find the area. [Hint: (length x breadth) and circumference (2(length + breadth)].	CO1

	3. Display the corresponding day of the date, for example, if the day is 1 then they should display Monday. If the day is greater than 7, display invalid day. 4. Using Java control statements write a program to check if the given input number is a prime number of not? **Practice Questions:** 1. Write a Java program to display the numbers in the following format using nested for loop. 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5 2. Write a Java program to generate prime numbers between n and m.(Hint: A prime number is a natural number greater than 1 that has no positive divisors other than 1 and itself. Eg: 2, 3, 5,7,11 etc.)	
L4	**<u>Arrays,</u>** 1. Given the float array, return the sum of all elements in it. 2. Return the sum of the first, middle and last elements in the array if the length of the array is odd, if the length is even then the sum of the first and last element should be displayed. [Hint: import required java.util.Array package] 3. Demonstrate the 2x2 matrix multiplication using Arrays. **Practice Questions:** 1. Consider an integer array with random values as 12, 67, 87,34, 90, 23, 67, 88 and output the sorted array. 2. Calculate the sum and average of array containing elements as 12, 67, 87,34, 90, 23, 67, 88.	CO2
L5	**<u>Classes and Methods</u>** 1. A post office charges parcel senders according to the weight of the parcel. For each parcel having weight 2 Kg or less, the charge is Rs 32.50 and for each extra Kg, there is an additional charge of Rs 10.50. Create a class with necessary fields for storing customer name, parcel id and parcel weight. Use constructor & methods for storing & processing the parcel details. Create an array of objects to process the charges of 'n' parcels. 2. Create a class containing an array of integers as its data members. Add the following methods: • A ***default constructor.*** • A ***parameterized constructor*** which initializes the array of the object. • A method called ***print*** to print the array contents. • A method called ***search*** to search for an element in the array. • A method called ***compare*** which compares 2 objects for equality. 3. Demonstrate constructor overloading, method overloading, and the usage of ***this*** keyword.	CO2

L6	**String handling** 1. Create Book class with field name, id, price with a constructor and get methods for all fields. [hint: constructor will be Book(id, name, price), methods will be getID(), getName() and getPrice().] 2. Create a class as Student containing ID, Marks (array of 5). Now create methods for students to find the total and print the student score. Identify if the student is passed or failure with a minimum mark as 40M. 3. Write a menu driven program to do the following: -To compare two strings -To convert the uppercase character to lower and vice-versa -To display whether an entered string is a substring of the other or not -If the entered string is a substring of the other, replace it with "Hello"	CO3
L7	**Exception handling** 1. Given an integer array of size 3. Read the input from the keyboard and store in the array if it is of numeric type, otherwise display the exception message as "*Enter a numeric value*". Repeat this step until the user enters 3 numeric values. 2. Given a list of filenames stored in an array of strings. Check whether all the filenames have the same extension or not. Otherwise throw a user-defined exception, namely, *Invalid_File_Extn*. Define a custom exception handler to display the error message. 3. Demonstrate generic exception handler for processing all kinds of exceptions.	CO4
L8	**Inheritance and Access Modification** 1. Create Area class with variable height. Create a triangle class that extends area class with variables base and method to calculate the area. Create a rectangle class that extends area class with variable width and method to calculate area. Now create a triangle and rectangle objects and print their areas. 2. Create two classes as Vehicle and Car where car is single inheritance subclass of vehicle. Now create an object for the car and print the details. Note that data member and methods can be defined in both classes. 3. In a single program demonstrate default, public, protected, and private access modifiers. **Practice Questions:** 1. Write a java program to store student record of college named "MIT". Class Student_Detail should contain name, id, college_name as its members.display_details() method should display the details of the students. 2. Demonstrate the usage of Protected access modifier in Java. 3. Demonstrate the default access modifier in Java. 4. Create a base class "Game" with method called "type" that prints "indoor & outdoor games". Create a subclass cricket with a method called "type" that prints "cricket is an outdoor game". Create one more subclass of "Game" called "chess" with a method "type" that prints "chess is an indoor game". Write a complete Java program for the above to understand the Dynamic method dispatch concept.	CO3
L9	**Interface & Abstract Class**	CO3

	1. Write a java program to store student record of college named "MIT". Class Student_Detail should contain name, id, college_name as its members.display_details() method should display the details of the students. 2. Create an interface called "sports" with methods getNumberOfGoals and dispTeam. Create classes Hockey and football that uses the interface "sports". Write the appropriate code for the methods. **Practice Questions:** 1. Write a counter program to count the number of objects created. 2. Write a program to compute the area of a square and a triangle by using abstract class.	
L10	**<u>File handling</u>** 1. To count the number of characters, vowels, lines and words in a given file.[Hint :Use read method] 2. Write a program to display all the files and directories of a directory using File object. 3. Write a menu driven program to do the following: Write to a file, read from the file, copy bytes from one file to another file [Hint: Use read and write methods] 4. To read and write primitive data using random access file and append some information. [Hint: Use RandomAccessFile class]	CO4
L11	**<u>Multithreaded Programming & Swings</u>** 1. Develop a multithreaded program to perform the following tasks: Main thread → Input two integer arrays Arr1[] and Arr2[]. Display the results obtained by Thread-1 and Thread-2. Thread-1 → Compute the union of Arr1[] and Arr2[]. Thread-2 → Find the intersection of Arr1[] and Arr2[]. 2. Write a menu driven program to create thread using runnable interface and inheriting thread class. [Hint: Make use of Extends and Implements keywords] 3. Write an GUI application(Counter) as shown in the Fig. Each time the "Count" button is clicked, the counter value shall increase by 6.  **Practice Questions:** 1. Write a program to create multiple threads. [Hint: Multiple instances of thread] 2. Write a java application using swings(hello world swing) to display "HELLO WORLD"	CO5
L12	**End Sem Exams**	

References:

1.	Patrick Naughton and Herbert Schildt,The Complete Reference -Java 2, 8th Edition, Tata McGrawHill, 2000
2.	E.Balaguruswamy, Programming with Java A Primer, 2nd Edition, Tata McGrawHill, 2000
3.	Aaron Walsh and John Fronckowiak, Java Programming Bible,1st Edition, IDG Books, India, 2000
4.	C Xavier Java Programming A Practical Approach – 1stEdition,TataMcGrawHill Publishing Co Ltd., 2011
5.	Dr. G.T.Thampi , Object Oriented Programming In Java , DreamTech, 2009

Submitted by: Dr. Vidya Rao and Dr. Nandini Kumari

(Signature of the faculty)

Date: 09-08-2022

Approved by: Dr. Radhika M Pai

(Signature of HOD)

Date: 09-08-2022

FACULTY MEMBERS TEACHING THE COURSE (IF MULTIPLE SECTIONS EXIST):

FACULTY	SECTION	FACULTY	SECTION
Dr. Raghuram Holla	A		
Dr. Vidya Rao	B		