LOK JAGRUTI UNIVERSITY (LJU)

INSTITUTE OF ENGINEERING & TECHNOLOGY

Department of Computer Science and Design (703)

Bachelor of Engineering (B.E.) – Semester – I

Course Code:	**117032191**
Course Name:	**Computer Programming using Java-1**
Category of Course:	Engineering Science Course (ESC)
Prerequisite Course:	Mathematics - I (117031191)

Teaching Scheme				
Lecture (L)	**Tutorial (T)**	**Practical (P)**	**Credit**	**Total Hours**
3	**0**	**6**	**6**	**30**

Syllabus

Unit No.	Topic	Prerequisite Topic	Successive Topic	Teaching Hours
01	**Introduction to Computer**			**2 (5%)**
	1.1 Basic Model of Von-Neumann Architecture		Introduction to Data Structure (017033292 - Unit 1.1)	
	1.2 Programming Language - Machine, Assembly and High Level Languages, Assembler, Compiler, Interpreter			
	1.3 Algorithm and Flowchart			
02	**Basics of Java**			**3 (10%)**
	2.1 Features of Java, Byte Code and Java Virtual Machine, JDK, JRE, Basic Structure of Program, Comments- Single Line and Multiline	---		
	2.2 Compiling and Execution of a Simple Java Program			
	2.3 Identifiers, Constants, Separators, Variables, Keywords,Literals			
	2.4 Data Types- Primitive | Non-Primitive, Typecasting	Identifiers (117032191 – Unit-2)	Introduction to Data Structures (017033292- Unit-1.1)	
	2.5 Classification of Operators - Arithmetic, Relational, Logical, Assignment, Increment/Decrement, Bitwise (&, |, ^ , <<, >>, ~), and Ternary Operators,String concatenation operator(+), Evaluation of Expression, Precedence and Associativity		Introduction to Python (017032491-Unit-1.1)	
	2.6 Command line argument, Use of Wrapper Class, Scanner class and its methods: (nextInt(), nextDouble(), nextFloat(), nextBoolean(), next(),nextLine()). Math class and its Methods : min(), max(), pow(), sqrt(), cbrt(), ceil(), floor(), Math.PI, Math.E, abs(), random(),	---		
03	**Decision Making Statements and Branching**			**3 (10%)**
	3.1 Simple if, If-else, Nested if, if else…if ladder	Operators (117032191 – Unit-2)	Unix/Linux Operating System (017033301- Unit 10), Output Primitives (017033501-Unit 2)	
	3.2 Switch Statement			
04	**Looping Constructs**			**3 (10%)**
	4.1 Entry Controlled Loop: while loop, for loop. Exit Controlled Loop: do-while.	Operators (117032191 – Unit-2)	JavaScript (017033691-Unit-4.1,4.2) Output Primitives (017033501-Unit 2)	
	4.2 Nesting of Loops, Different patterns using nested loop	Loops (117032191 – Unit-2)		
	4.3 Break and Continue Statements			
05	**Arrays**			**3 (10%)**
	5.1 One Dimensional Arrays - Declaration and Initialization, Print elements of array using enhanced for loop (for each loop)	Data Types (117032191 – Unit-2)	Introduction to Data Structures (017033292- Unit-1.3), JavaScript (017033691-Unit-4.1, 4.2), Output Primitives, 2D Viewing (017033501-Unit-2.1,2.2,2.3,2.4,4.1,4.2,4.3)	
	5.2 Two Dimensional Arrays - Declaration and Initialization, Matrix Operations (for each loop)	One Dimensional Arrays (117032191– Unit-5)	JavaScript (017033691-Unit-4.1, 4.2), Output Primitives, 2D Viewing (017033501-Unit-2.1,2.2,2.3,2.4,4.1,4.2,4.3) Introduction to Data Structure, Stack-1, Queue-1 (017033292 - Unit 1.3,2.1,4.1)	
	5.3 Ragged Array	Two Dimensional Arrays (117032191– Unit-5)		

06	**Strings**			**3 (10%)**
	6.1 String class and its methods (charAt(), length(), concat(), equals(), equalsIgnoreCase(), compareTo(), compareToIgnoreCase(), toUpperCase(), toLowerCase(), split(), replace(),toString(),startsWith(),endsWith(),indexOf(),toCharArray(),trim())	One Dimensional Arrays (117032191– Unit-5)		
	6.2 StringBuffer class and its method (append(), insert(), replace(), delete(), reverse(),capacity())	One Dimensional Arrays (117032191– Unit-5)		
	6.3 Difference between String and StringBuffer class	String class and StringBuffer class (117032191– Unit-6)		
07	**Class, Object and Method Part 1**			**3 (10%)**
	7.1 General form of class, Declaring object, Characteristics of OOP, Scope & Life time of Variable, Static variable, Instance variable	One Dimensional Arrays (117032191– Unit-5)		
	7.2 Introducing Method , Method declaration, calling and signature of method	Data Types (117032191 – Unit-2)	JavaScript, Node.JS (017033691-Unit-4.1,4.2,5.1,5.2,8.1,8.2,8.3)	
	7.3 Categories of Method, Recursion	Method (117032191 – Unit-7)	Inter Process Communication, Deadlock Handling (017033301-Unit 3,4)	
08	**Class, Object and Method Part 2**			**3 (10%)**
	8.1 Method overloading	Method (117032191 – Unit-7)		
	8.2 Actual and Formal Arguments	Categories of Method (117032191 – Unit-7)		
	8.3 Passing Arrays to Method, array of objects	One Dimensional Arrays (117032191– Unit-5), Method (117032191 – Unit-7)		
	8.4 Call by Value and Call by Reference,	Method (117032191 – Unit-7)		
	8.5 Returning object, Object as Parameter	Categories of Method (117032191 – Unit-7)		
09	**Constructor and keywords**			**3 (10%)**
	9.1 Types of constructor: Usage, Types of constructor: Default, Parameterized and Copy Constructor, Constructor Overloading	Class and Method (117032191 – Unit-7)		
	9.2 this keyword and its uses	Keywords (117032191 – Unit-2)		
	9.3 static keyword: method, block, class, instance block	Scope & Life time of Variable, Keywords (117032191 – Unit-2)		
	9.4 Nested class: static inner and Non static inner class	Class (117032191 – Unit-7)	Singly Linked List (017033292- Unit-6.2), Doubly and Circular Link List(017033292- Unit-7.1, 7.2)	
10	**Inheritance**			**4 (15%)**
	10.1 Introduction of inheritance and its types	Class and Method (117032191 – Unit-7)		
	10.2 Inheriting Data members and Methods	Inheritance (117032191 – Unit-10)		
	10.3 Method overriding	Method (117032191 – Unit-7)		
	10.4 super and final keyword	Keywords (117032191 – Unit-2)		

Major Components/ Equipment	
Sr. No.	**Component/Equipment**
1	Computer
2	Notepad ++, Java Compiler

Sr No.	Practical Title	Link to Theory Syllabus
1	WAP to calculate the Area of a Circle, Area and Perimeter of rectangle, Area of Triangle.	Unit-2
2	WAP that reads two nos. from key board and gives their addition, subtraction, multiplication, division and modulo.	Unit-2
3	WAP to enter three subject marks, and calculate total, percentage of student and display the same in proper format.	Unit-2
4	WAP to convert days into months and days.	Unit-2
5	WAP that converts Fahrenheit temperature to centigrade and vis-a-versa Cent.=(5* (fahr.-32))/9	Unit-2
6	WAP to swap two numbers with and without using temporary variable	Unit-2
7	WAP to determine a given number is ‘odd’ or ‘even’ and print the following message “Number is ODD” or “Number is Even” (i) Without using else option. (ii) With else option	Unit-3
8	WAP to accept three numbers from user and Print Maximum number	Unit-3
9	WAP to print grade of a student using following rules : Percentage >70 means Grade A Percentage 60-70 means Grade B Percentage 50-60 means Grade C Percentage <50 means Grade F	Unit-3
10	WAP to perform addition, multiplication, subtraction and division with Switch statement.	Unit-3
11	WAP that reads number from 1 to 7 and accordingly it should print MONDAY to SUNDAY.	Unit-3
12	WAP to enter a character and check whether it is a vowel or consonant using switch statement	Unit-3
13	WAP to check whether entered character is vowel or not.	Unit-3
14	WAP to display “Hello” five times.	Unit-4
15	WAP to display 1-10 numbers, 20-30 numbers	Unit-4
16	WAP to display multiplication table.	Unit-4
17	Assume that you want to make the sum of 1 to 100. Give the necessary code to perform the same using (1) For loop (2) While loop (3) Do-while loop	Unit-4
18	WAP for finding sum of 1 to k. The number k should be read from the keyboard using Command line argument.	Unit-4
19	WAP to print multiple of N from given range of unsigned integers. For example, if N=5 and range is [17, 45] it prints 20, 25, 30, 35, 40, 45. Take input using Scanner class	Unit-4
20	WAP to find sum of all integers greater than 100 & less than 200 and are divisible by 5.	Unit-4
21	WAP to count ODD and EVEN numbers from given 10 numbers	Unit-4
22	WAP to find the sum of first N odd numbers.	Unit-4
23	WAP to find 1+3/5+5/7+7/9+… series. Print addition of first N part.	
24	WAP to find 1+1/2+1/3+1/4+……+1/N series.	Unit-4
25	WAP to find Factorial of a number.	Unit-4
26	WAP to generate Fibonacci series of numbers	Unit-4
27	WAP to reverse a number.	Unit-4
28	WAP to calculate sum of digits (Ex: 123 => so sum of digit = 1+2+3=6)	Unit-4
29	WAP to find out Armstrong Numbers. Example: - 153 is an Armstrong Number.	Unit-4
30	WAP to check whether the given number is Prime or not. OR Write a Java program to find and print prime numbers between the numbers 1 to n, where the number n should be read from the keyboard.	Unit-4
31	WAP to check whether a number is a perfect number or not. (e.g. 123 is a perfect no i.e.1+2+3=1*2*3)	Unit-4
32	WAP to find out sum of first and last digit of a given number	Unit-4
33	WAP to print following pattern using loop statement for n row. * * * * * * * * * * * * * * * 1 1 3 1 3 5 1 3 5 7 1 0 1 1 0 1 0 1 0 1 1 2 3 4 5 6 7 8 9 10	Unit-4

	* ## *** #### ***** ###### ******* 1 2 2 3 3 3 4 4 4 4 5 5 5 5 5 ***** **** *** ** * 54321 4321 321 21 1 1 2 3 4 5 2 3 4 5 3 4 5 4 5 5	
34	WAP to Print following pattern using loop statement for n row * * * * * * * * * * 1 A B 1 2 3 A B C D 1 2 3 4 5 1 A B 2 3 4 C D E F 5 6 7 8 9 1 2 2 3 3 3 4 4 4 4 1 1 2 3 1 2 3 4 5 1 2 3 4 5 6 7 1 2 3 4 5 6 7 8 9	Unit-4
35	WAP to read 10 numbers from user and find Sum, Maximum, Minimum and Average of them	Unit-5
36	WAP which declares array of 10 integers, enter data and sum all the elements which are even. Also find maximum number from them.	Unit-5
37	WAP to accept array of N integers and find Largest odd number as well as largest even number and display them.	Unit-5
38	WAP read in an array of integers and print its elements in reverse order.	Unit-5
39	WAP to add two 3 x 3 Matrix or 2x2 matrix	Unit-5
40	WAP to find maximum element from 3*3 Matrices	Unit-5
41	WAP to find the minimum value from the array of 3 x 3.	Unit-5
42	WAP to count number of positive, negative and zero elements from 3 x 3 matrix.	Unit-5
43	WAP to read two matrix from the user and store the multiplication of two matrix in the resultant matrix. i.e. C=A * B	Unit-5
44	WAP to display transpose of given 3*3 matrix.	Unit-5
45	WAP to check given string is palindrome or not.	Unit-6
46	WAP to reverse the input string.	Unit-6
47	WAP to concatenate two strings without using built in function.	Unit-6
48	WAP to accept a string and count the number of vowels present in a string	Unit-6

49	Write method headers (not the bodies) for the following methods: a. Return a sales commission, given the sales amount and the commission rate. b. Display the calendar for a month, given the month and year. c. Return a square root of a number. d. Test whether a number is even, and returning true if it is. e. Display a message a specified number of times. f. Return the monthly payment, given the loan amount, number of years, and annual interest rate. g. Return the corresponding uppercase letter, given a lowercase letter.	Unit-7
50	A pentagonal number is defined as n(3n–1)/2 for n = 1, 2, . . ., and so on. Therefore, the first few numbers are 1, 5, 12, 22, ... Write a method with the following header that returns a pentagonal number: *public static intgetPentagonalNumber(int n)* Write a test program that uses this method to display the first 100 pentagonal numbers with 10 numbers on each line.	Unit-7
51	Write a method with the following header to display an integer in reverse order: *public static void reverse(int number)* For example, reverse(3456) displays 6543. Write a test program that prompts the user to enter an integer and displays its reversal.	Unit-7
52	WAP to find GCD of the 2 numbers using recursion	Unit-7
53	WAP to calculate nCr using recursion. nCr = n! / (r! * (n-r)!)	Unit-7
54	WAP to generate Fibonacci series of n given numbers using recursion.	Unit-7
55	WAP to find sum of digit of given number using recursion.	Unit-7
56	Write a program which takes five numbers as command line argument from user, store them in one dimensional array and display count of negative numbers.	Unit-8
57	Write a program that creates and initializes a four integer element array. Calculate and display the average of its values.	Unit-8
58	Write a program that prompts the user to enter the number of students, the students' names, and their scores, and prints student names in decreasing order of their scores.	Unit-8
59	Write a program that creates an integer array and then uses a for loop to check whether the array is sorted from smallest to largest. If so, it prints "sorted" otherwise it prints "Not sorted"	Unit-8
60	Write a java program to calculate total and average to five values. Pass input values as constructor parameter.	Unit-9
61	Write a class named Rectangle to represent a rectangle. It contains following members: DATA: width(double) and height (Double) that specify the width and height of the rectangle. Methods: 1. A no-arg constructor that creates a default rectangle. 2. A constructor that creates a rectangle with the specified width and height. 3. A method named getArea() that returns the area of this rectangle. 4. A method named getPerimeter() that returns the perimeter.	Unit-9
62	Write a JAVA program to create a super class called figure that storesthe dimensions of a two dimensional object. It also defines a method called area () that computes the area of an object. The program derives two sub classes from figure. The first is rectangle and the second is Triangle. Each of these subclasses overrides area (), so that it returns the area of a rectangle and a triangle respectively	Unit-10

Proposed Theory + Practical Evaluation Scheme by Academicians
(% Weightage Category Wise and it's Marks Distribution)

L:	**3**	**T:**	**0**	**P:**	**6**

Note: In Theory Group, Total 4 Test (T1+T2+T3+T4) will be conducted for each subject.
Each Test will be of 25 Marks.
Each Test Syllabus Weightage: Range should be 20% - 30%

Group (Theory or Practical)	Group (Theory or Practical) Credit	Total Subject Credit	Category	% Weightage	Marks Weightage
Theory	3	6	MCQ	20%	40
Theory			Theory Descriptive (Mainly Programming)	30%	60
Theory			Formulas and Derivation	0%	0
Theory			Numerical	0%	0
Expected Theory %	**50%**		**Calculated Theory %**	**50%**	**100**
Practical	3		Individual Project	25%	50
Practical			Group Project	15%	30
Practical			Internal Practical Evaluation (IPE)	10%	20
Practical			Viva	0%	0
Practical			Seminar	0%	0
Expected Practical %	**50%**		**Calculated Practical %**	**50%**	**100**
Overall %	**100%**			**100%**	**200**

Course Outcome	
	Upon completion of the course, Students will be able to

CO1	Understand basics of flowchart, algorithm, object oriented programming and decision making statements.
CO2	Apply the use of looping constructs, arrays and String.
CO3	Apply the concepts of classes, object and methods to create simple program.
CO4	Understand the concept of Constructor, keywords and Inheritance.
Suggested Reference Books	
1	Java : The Complete Reference, Herbert Schildt, McGrawHill
2	Core Java Volume I-Fundamentals, Cay S. Horstmann, Gary Cornell, Pearson India Education Services Pvt. Ltd.
3	Programming in JAVA, Second Edition , Sachin Malhotra & Saurabh Choudhary, Oxford
4	Java in 24 Hours, Cadenhead Rogers, Sams Publishing
5	Programming with Java, E Balagurusamy, Tata McGraw Hill

List of Open Source Software/Learning website	
1	https://www.w3schools.com/
2	http://www.c4learn.com/javaprogramming/
3	https://www.geeksforgeeks.org/java/
4	http://www.tutorialspoint.com/java/
5	https://www.javatpoint.com/java-tutorial

Practical Project/Hands on Project

Sr. No.	Project List	Linked with Unit
1	Airline reservation system, The main features of the airline reservation system are: Reservation and cancellation of the airline tickets Perform transaction management Maintain passenger records and report on the daily business transactions	All Units
2	Consider an array marks[20][5] which stores the marks obtained by 20 students in 5 subjects. Now write a program to a) Find average marks obtained in each subject b) Find average marks obtained by every student c) Find the number of students who have scored below 50 in their average d) Display the scores obtained by every student in proper format e) Display top five students on the basis of percentage f) Display grade in each subject g) Calculate SPI/CPI of every student	All Units
3	Store Management System is a simple console-based application. In this project, you can manage a typical ‘fashion wear’ department store. You can add goods, edit goods, search, delete and display the goods. Record the information (rate, quantity, name and code) of the added goods. You can search the goods by rate, code or quantity. And, similar goes for display; you can display the items by quantity, rate or code.	All Units
4	Library management System is a simple console-based application. You can add books, edit books, search, delete and display the books. Record the information (name of book, Author of book, Publisher of book, issued date/returned date of book). You can search the book by name, Author and Publication. And, similar goes for display; you can display the books according to name, Author and Publication.	All Units
5	Personal Diary Management System is a console application. In this project, user can keep their personal record like they do in a diary. You can keep records of the important things you do in your daily life, like meetings and various other tasks. You can handles password protection, add new diary record, modify and update an added record, delete or remove a record. Also you can modify/change a password, display added record in list.	All Units