LAB MANUAL
Department of Computer Science and Engineering (GEHU)

PROGRAMMING IN JAVA LAB (PCS 408)

Vision and Mission of the Department of Computer Science and Engineering

To become a Center of Excellence in the computer sciences and information technology discipline with a strong research and teaching environment that adapts swiftly to the challenges of the 21st century.

Mission

M1. To provide qualitative education and generate new knowledge by engaging in cutting-edge research and by offering state-of-the-art undergraduate, postgraduate and doctoral programs, leading to careers as Computer and IT professionals in the widely diversified domains of industry, government and academia.

M2. To promote a teaching and learning process that yields advancements in state-of-the-art in computer science and information technology, resulting in integration of research results and innovations into other scientific disciplines leading to new technologies and products.

M3. To harness human capital for sustainable competitive edge and social relevance by inculcating the philosophy of continuous learning and innovation in Computer Science and IT.

Program Educational Objectives (PEOs)

PEO 1. To produce students employable towards building a successful career based on sound understanding of theoretical and applied aspects as well as methodology to solve multidisciplinary real life problems.

PEO 2. To produce professional graduates ready to work with a sense of responsibility, ethics and enabling them to work efficiently individually and also as a team.

PEO 3. To impart the competency in students so that they are able to pursue higher studies and research in areas of engineering and other professionally related fields.

PEO 4. To inculcate ability to adapt to the changing technology through continuous learning.

Program Outcomes

PO1.Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2. Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3. 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.

PO4. 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.

PO5. Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.

PO6. 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.

PO7. 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 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.

PROGRAM SPECIFIC OUTCOMES (PSOs)

PSO1. Ability to analyze, design, implement, and test software systems based on requirement specifications and development methodologies of software systems.

PSO2. Apply computer science theory blended with engineering mathematics to solve computational tasks and model real world problems using appropriate programming language, data structure, and algorithms.

PSO3. Ability to explore technological advancements in various domains, evaluate its merits and identify research gaps to provide solution to new ideas and innovations.

Course Outcomes

After completion of the course students will be able to

1. Understand the object-oriented approach in programming along with the purpose and usage principles of inheritance, polymorphism, encapsulation and method overloading etc.

2. Demonstrate ability to test and debug Java programs using IDE
3. Analyze, design and develop small to medium sized application programs that demonstrate professionally acceptable programming standards
4. Demonstrate skills of developing event-driven programs using graphical user interfaces
5. Design applications for accessing databases using Java features.

Problem Statements

List of Programs

Topic	Concept and Problem statement		
(i)	Taking input from Command line and convert objects into primitive data type		Page No.
	1.	Write a java program to take input as a command line argument. Your name, course, universityrollno and semester. Display the information. Name: UniversityRollNo: Course: Semester:	
(ii)	Concepts of Java Control statements, Conditional statements, loops and iterations		
	2.	Using the switch statement, write a menu-driven program to calculate the maturity amount of a bank deposit. The user is (i) Term Deposit (ii) Recurring Deposit For option (i) accept Principal (p), rate of interest (r) and time period in years (n). Calculate and output the maturity amount (a) receivable using the formula a = p[1 + r / 100]n. For option (ii) accept monthly installment (p), rate of interest (r) and time period in months (n). Calculate and output the maturity amount (a) receivable using the formula a = p * n + p * n(n + 1) / 2 * r / 100 * 1 / 12. For an incorrect option, an appropriate error message should be displayed. [Use Scanner Class to take input]	
	3.	Program to find if the given numbers are Friendly pair or not (Amicable or not). **Friendly Pair** are two or more numbers with a common abundance.	25

		Input & Output format: • Input consists of **2 integers**. • The first integer corresponds to number 1 and the second integer corresponds to number 2. • If it is a Friendly Pair display **Friendly Pair** or displays **Not Friendly Pair**. For example,**6** and **28** are **Friendly Pair**. (Sum of divisors of 6)/6 = (Sum of divisors of 28)/28. Steps to check whether the given numbers are friendly pair or not • Input the numbers num1 and num2. • Initialize sum1 = sum2 = 0. • sum1 = sum of all divisors of num1. • sum2 = sum of all divisors of num2. • If (sum1 == num1) and (sum2 == num2), then print **"Abundant Numbers"**. • Else, print **"Not Abundant Numbers"**. Program to check whether the given numbers are friendly pair or not	
	4	Program to replace all 0's with 1 in a given integer. Given an integer as an input, all the 0's in the number has to be replaced with 1. For example, consider the following number Input: 102405 Output: 112415 Input: 56004 Output: 56114 Steps to replace all 0's with 1 in a given integer • Input the integer from the user. • Traverse the integer digit by digit. • If a '0' is encountered, replace it by '1'. • Print the integer.	

(iii)	Array in Java		
	5.	Printing an array into Zigzag fashion. Suppose you were given an array of integers, and you are told to sort the integers in a zigzag pattern. In general, in a zigzag pattern, the first integer is less than the second integer, which is greater than the third integer, which is less than the fourth integer, and so on. Hence, the converted array should be in the form of e1 < e2 > e3 < e4 > e5 < e6. Test cases: Input 1: 7 4 3 7 8 6 2 1 Output 1: 3 7 4 8 2 6 1 Input 2: 4 1 4 3 2 Output 2: 1 4 2 3	
	6.	The problem to rearrange positive and negative numbers in an array . Method: This approach moves all negative numbers to the beginning and positive numbers to the end but changes the order of appearance of the elements of the array. Steps:	

		1. Declare an array and input the array elements. 2. Start traversing the array and if the current element is negative, swap the current element with the first positive element and continue traversing until all the elements have been encountered. 3. Print the rearranged array. Test case: • Input: 1 -1 2 -2 3 -3 • Output: -1 -2 -3 1 3 2	
	7.	Program to find the saddle point coordinates in a given matrix. A saddle point is an element of the matrix, which is the minimum element in its row and the maximum in its column. For example, consider the matrix given below Mat[3][3] 1 2 3 4 5 6 7 8 9 Here, 7 is the saddle point because it is the minimum element in its row and maximum element in its column. Steps to find the saddle point coordinates in a given matrix 1. Input the matrix from the user. 2. Use two loops, one for traversing the row and the other for traversing the column. 3. If the current element is the minimum element in its row and maximum element in its column, then return its coordinates. 4. Else, continue traversing.	

(iv)	String Handling in Java		
	8.	Program to find all the patterns of 0(1+)0 in the given string. Given a string containing 0's and 1's, find the total number of 0(1+)0 patterns in the string and output it. 0(1+)0 - There should be at least one '1' between the two 0's. For example, consider the following string. **Input:** 01101111010 **Output:** 3 **Explanation:** **0110**1111010 - count = 1 011**01111**10 - count = 2 01101111**010**- count = 3 Step to find all the patterns of 0(1+)0 in the given string • Input the given string. • Scan the string, character by character. • If the given pattern is encountered, increment count. • Print count. Program to find all the patterns of 0(1+)0	
(v)	Class definition, creating objects and constructors		

	9.	Write a java program to create a class named '**Bank** ' with the following data members: Name of depositor Address of depositor Account Number Balance in account Class '**Bank**' has a method for each of the following: 1 - Generate a unique account number for each depositor For first depositor, account number will be 1001, for second depositor it will be 1002 and so on 2 - Display information and balance of depositor 3 - Deposit more amount in balance of any depositor 4 - Withdraw some amount from balance deposited 5 - Change address of depositor After creating the class, do the following operations 1 - Enter the information (name, address, account number, balance) of the depositors. Number of depositors is to be entered by user. 2 - Print the information of any depositor. 3 - Add some amount to the account of any depositor and then display final information of that depositor 4 - Remove some amount from the account of any depositor and then display final information of that depositor 5 - Change the address of any depositor and then display the final information of that depositor 6 - Randomly repeat these processes for some other bank accounts.	
	10.	Define a class **WordExample** having the following description: Data members/instance variables: private String strdata : to store a sentence. **Parameterized Constructor**	

		WordExample(String) : Accept a sentence which may be terminated by either'.', '? 'or '!' only. The words may be separated by more than one blank space and are in UPPER CASE. **Member Methods:** **void countWord():** Find the number of words beginning and ending with a vowel. **void placeWord():** Place the words which begin and end with a vowel at the beginning, followed by the remaining words as they occur in the sentence.	
(vi)	Method overloading (Compile time Polymorphism)		
	11.	Write a Java program to create a class called **ArrayDemo** and overload arrayFunc() function. **void arrayFunc(int [], int)** ➔To find all pairs of elements in an Array whose sum is equal to a given number : Array numbers= [4, 6, 5, -10, 8, 5, 20], target=10 **Output :** Pairs of elements whose sum is 10 are : 4 + 6 = 10 5 + 5 = 10 -10 + 20 = 10 **void arrayFunc(int A[], int p, int B[], int q)**➔Given two sorted arrays A and B of size p and q, Overload method arrayFunc() to merge elements of A with B by maintaining the sorted order i.e. fill A with first p smallest elements and fill B with remaining elements. Example: Input : int[] A = { 1, 5, 6, 7, 8, 10 } int[] B = { 2, 4, 9 } Output:	

		Sorted Arrays: A: [1, 2, 4, 5, 6, 7] B: [8, 9, 10] **(Use Compile time Polymorphism Method Overloading)**	
(vii)	Method overriding (Runtime Polymorphism), Abstract class and Abstract method, Inheritance, Interfaces and Package.		
	12.	Write a java program to calculate the area of a rectangle, a square and a circle. Create an abstract class '**Shape**' with three abstract methods namely **rectangleArea()** taking two parameters, **squareArea()** and **circleArea()** taking one parameter each. Now create another class '**Area'** containing all the three methods rectangleArea(),squareArea() and circleArea() for printing the area of rectangle, square and circle respectively. Create an object of class Area and call all the three methods. **(Use Runtime Polymorphism)**	
	13.	Write a java program to implement abstract class and abstract method with following details: **Create a abstract Base Class Temperature** **Data members:** double temp; **Method members:** void setTempData(double) abstact void changeTemp() **Sub Class Fahrenheit (subclass of Temperature)** **Data members:** double ctemp; **method member:** Override abstract method changeTemp() to convert Fahrenheit temperature into degree Celsius by using formula C=5/9*(F-32) and display converted temperature	

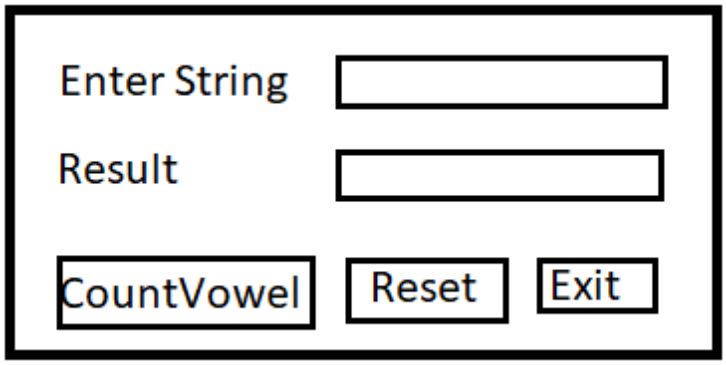
		Sub Class Celsius (subclass of Temperature) **Data member:** double ftemp; **Method member:** Override abstract method changeTemp() to convert degree Celsius into Fahrenheit temperature by using formula F=9/5*c+32 and display converted temperature	
	14.	Write a java program to create an interface that consists of a method to display **volume ()** as an abstract method and redefine this method in the derived classes to suit their requirements. Create classes called **Cone**, **Hemisphere** and **Cylinder** that implements the interface. Using these three classes, design a program that will accept dimensions of a cone, cylinder and hemisphere interactively and display the volumes. Volume of cone = (1/3)πr2h Volume of hemisphere = (2/3)πr3 Volume of cylinder = πr2h	
	15.	Design a **student management system** using Java packages. The system should manage information about students, courses, and their enrolments. You will organize the functionality across multiple packages and use object-oriented principles such as encapsulation. 1. In the **student** package, create a Student class with fields for studentId, name, and age. Include a constructor to initialize these fields and a getStudentInfo() method to return the student's details. 2. In the **course** package, create a Course class with fields for courseId, courseName, and credits. Include a constructor and a getCourseInfo() method to return the course's details. 3. In the **enrollment** package, create an Enrollment class that holds references to Student and Course objects. Include a constructor and a getEnrollmentInfo() method to return the enrollment details.	

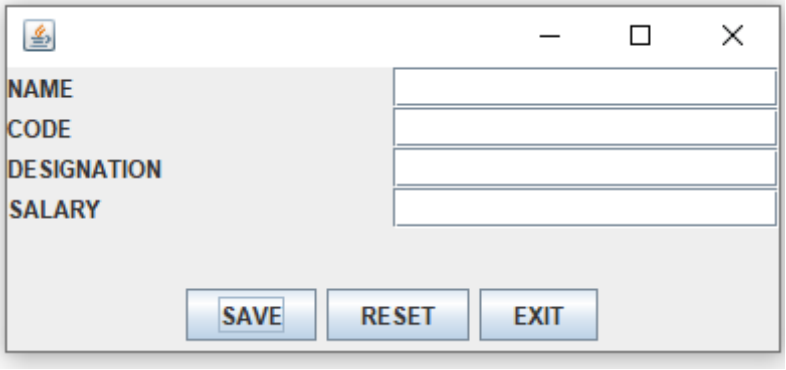
		4. In the default package, create instances of Student, Course, and Enrolment, then print details using the getStudentInfo(), getCourseInfo(), and getEnrollmentInfo() methods.	
(viii)	Exception handling		
	16.	Write a java program to accept and print the employee details during runtime. The details will include employee id, name, dept_ Id. The program should raise an exception if user inputs incomplete or incorrect data. The entered value should meet the following conditions: (i) First Letter of employee name should be in capital letter. (ii) Employee id should be between 2001 and 5001 (iii) Department id should be an integer between 1 and 5. If the above conditions are not met then the application should raise specific exception else should complete normal execution.	
	17.	Create a class MyCalculator which consists of a single method power(int, int). This method takes two integers, **n** and **p**, as parameters and finds $\mathbf{n^p}$. If either n or p is negative, then the method must throw an exception which says "n and p should be non-negative". **Input Format** Each line of the input contains two integers, n and p . Output Format Each line of the output contains the result ,if neither of n and p is negative. Otherwise the output contains "n and p should be non-negative".	

		Sample Input 3 5 2 4 0 0 -1 -2 -1 3 **Sample Output** 243 16 java.lang.Exception: n and p should not be zero. java.lang.Exception: n or p should not be negative. java.lang.Exception: n or p should not be negative. Explanation In the first two cases, both **n** and **p** are positive. So, the power function returns the answer correctly. In the third case, both **n** and **p** are zero. So, the exception, "**n** and **p** should not be zero." is printed. In the last two cases, at least one out of **n** and **p** is negative. So, the exception, "**n** or **p** should not be negative." is printed for these two cases.	
(ix)	File Handling in Java		
	18.	Write a java file handling program to count and display the number of palindromes present in a text file "myfile.txt". Example: If the file "myfile.txt" contains the following lines, My name is NITIN Hello aaa and bbb word How are You ARORA is my friend Output will be => 4	
(x)	Multithreaded programming		
	19.	Write a program MultiThreads that creates two threads- one thread with the name CSthread and the other thread named ITthread. Each thread should display its respective name and execute after a gap of	

		500 milliseconds. Each thread should also display a number indicating the number of times it got a chance to execute.	
	20.	Write a java program for to solve producer consumer problem in which a producer produce a value and consumer consume the value before producer generate the next value.	
(xi)	Collection and Generic Framework:		
	21.	Write a method removeEvenLength that takes an ArrayList of Strings as a parameter and that removes all of the strings of even length from the list. **(Use ArrayList)**	
	22.	Write a method swapPairs that switches the order of values in an ArrayList of Strings in a pairwise fashion. Your method should switch the order of the first two values, then switch the order of the next two, switch the order of the next two, and so on. For example, if the list initially stores these values: {"four", "score", "and", "seven", "years", "ago"} your method should switch the first pair, "four", "score", the second pair, "and", "seven", and the third pair, "years", "ago", to yield this list: {"score", "four", "seven", "and", "ago", "years"} If there are an odd number of values in the list, the final element is not moved. For example, if the original list had been: {"to", "be", "or", "not", "to", "be", "hamlet"} It would again switch pairs of values, but the final value, "hamlet" would not be moved, yielding this list: {"be", "to", "not", "or", "be", "to", "hamlet"}	

	23.	Write a method called alternate that accepts two Lists of integers as its parameters and returns a new List containing alternating elements from the two lists, in the following order: • First element from first list • First element from second list • Second element from first list • Second element from second list • Third element from first list • Third element from second list • … If the lists do not contain the same number of elements, the remaining elements from the longer list should be placed consecutively at the end. For example, for a first list of (1, 2, 3, 4, 5) and a second list of (6, 7, 8, 9, 10, 11, 12), a call of alternate (list1, list2) should return a list containing (1, 6, 2, 7, 3, 8, 4, 9, 5, 10, 11, 12). Do not modify the parameter lists passed in.	
(xii)	Swing, Java FX and Event Handling:		
	24.	Write a GUI program to develop an application that receives a string in one text field, and count number of vowels in a string and returns it in another text field, when the button named “CountVowel” is clicked. When the button named “Reset” is clicked it will reset the value of textfield one and Textfield two . When the button named “Exit” is clicked it will closed the application.	

			
	25.	Design a JavaFX-based movie ticket booking system where users can select a movie, choose a showtime, and specify the number of tickets they wish to purchase. (a) The system should calculate the total cost based on the movie and the number of tickets, and display it in the UI. (b) Users can confirm their booking with a button, which will show a confirmation message. (c) Additionally, a reset button should allow users to clear all selections and start over. (d) Provide an exit button to close the application.	
(xiii)	Java Database Connectivity (JDBC)		
	26.	Create a database of employee with the following fields Name Code Designation Salary	

		 (a) Write a java program to create GUI java application to take employee data from the TextFields and store in database using JDBC connectivity. (b) Write a JDBC Program to retrieves all the records from employee database.	