JAVA LAB TRAINING

B.TECH-4th Semester

COURSE OUTCOME:

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

1) Apply object-oriented principles or features in the software design process to develop Java programs for real life applications.
2) Reduce the complexity of procedural language by employing different OOP technologies for developing robust and reusable software.
3) Apply some common object-oriented design patterns and give examples of their use.

Assignment-1 (Three Days)

Topic: JAVA fundamental, data types, operators, condition checking and loops.

Objective: The objective of this assignment is to learn how to write Simple JAVA Programs and ideas about java data types, operators, condition checking and loops.

1. Write a program to print your name.
2. Write a Java Program to print your college name by taking input through command line arguments.
3. Write a Java Program to take two inputs (name and college name) through command line argument and print as follows:
   <name> is a BTECH student of <college name> college.
   (e.g.: Input: Ram AOT
   Output: Ram is a BTECH student of AOT college)
4. Write a program to read the price of an item in the decimal form (like 175.95) and print the output in paise (like 17595 paise).
5. Write a program to convert the given temperature in *Fahrenheit* to *Celsius* using the following formula:
   C /5 = (F-32)/9.
6. Write a Java Program to add two Numbers. (By Keyboard Input or Command Line Input).
7. Write a Java Program to demonstrate arithmetic operations between two numbers using command line arguments.
8. Write a Java Program to check if a number is Even or Odd.
9. Write a Java Program to check Leap Year.
10. Write a Java Program to multiply a number by 2 and divide the same number by 4 using shift operator.
11. Write a Java Program to swap two numbers using the bitwise operator.
12. Write a program to calculate the Simple Interest (si) while your inputs are principle (p), time in years (n)

and rate of interest (r).

13. a) Write a Java Program to convert long to int.

b) Write a Java Program to find the ASCII value of a Character.

14. Write a java program to take an integer input from the user and print its reverse.

15. Write a java program to print Fibonacci series.

16. Write a java program to print the nth Fibonacci number. (n must be taken as an input).

17. Write a Java Program to find Sum of n-natural numbers.

18. Write a Java Program to check palindrome numbers.

19. Write a Java Program to display prime numbers from 1 to n.

20. Write a Java Program to check the perfect number.

21. Write a Java Program to check Armstrong number.

22. Write a Java Program to convert a binary number to octal number.

23. Write a Java Program to display the sequence ABA, BCB, CDC, DED,……..

24. Write a Java Program to display the sequence AMM, COO, EQQ ,..........

25. Write a Java Program to display the sequence A1, B2, C3, ..., …Y25,Z26, A1, B2,...............

26. Create a Java program that performs a fun mathematical operation using loops. Ask the user for a number and then apply the following operation until the result becomes a single-digit number: Multiply each digit of the number and repeat the process. Output each intermediate result.

```
(Test Cases,    I/P: 721
                O/P: Intermediate result: 14
                     Intermediate result: 4
                     Final result: 4
          I/P: 123
          O/P: Intermediate result: 6
               Final result: 6).
```

27. Write a function that returns the second last digit of the given number.

(For example, if the given number is 197, the second last digit is 9
Note1 - The second last digit should be returned as a positive number, i.e. if the given number is -197, the second last digit is 9.
Note2 - If the given number is a single digit number, then the second last digit does not exist. In such cases, the function should return -1.)

28. Develop a simple text-based game using Java loops that simulates a journey through a maze. The maze consists of rooms represented by numbers (e.g., 1, 2, 3, etc.). The player starts at Room 1 and can choose directions (north, south, east, west) to move to adjacent rooms. Implement a win condition when the player reaches Room 10.

29. Write a java program for reverse the digits of a given number and add it to the original number until a palindrome is obtained.

Also provide all intermediate results.

```
(Test Cases,    I/P: 174
                O/P: Intermediate result: 645
                     Intermediate result: 1191
                     Intermediate result: 3102
```

Intermediate result: 5115
Final result: 5115
I/P: 123
O/P: Intermediate result: 444
Final result: 444).

30. Write a java program to print Collatz Sequence.
(If number is even, then n = n / 2, If number is odd, then n = 3×n + 1. Repeat, until it becomes 1).
(Test Cases, I/P: 3 O/P: 3, 10, 5, 16, 8, 4, 2, 1
I/P: 6 O/P: 6, 3, 10, 5, 16, 8, 4, 2, 1
I/P: -6 O/P: ERROR).

31. Write a java program that simulates a basic number guessing game. The computer randomly selects a target number between 1 and 100, and the player has to guess the correct number within a limited number of attempts.

Assignment-2 (Two Days)

Topic: Class, Object and Constructor.

Objective: The objective of this assignment is to learn about Classes, Objects and Constructors.

1. Create a class ***Test,*** make instance variable [int x], method [void show()] and also put main method inside that class and use the instance variable and method from main.

2. Create a class ***Test1,*** make instance variables [int x], method [void show()] and use them from main, declared in different classes.

3. Create a class; put a method inside this class which will return a class reference of the same class and/or different class object.

4. Write a JAVA Program to create a *Student class* with proper attributes like roll no, name, stream, and college. Create two instances of that class from main (declared in different classes) and print them.

5. Design a class to represent a *Bank Account*. Include the following things:

Fields

- Name of the depositor
- Account number
- Type of account
- Balance amount in the account

Methods

- To assign initial values
- To deposit an amount
- To withdraw an amount after checking balance
- To display the name and balance

From main (declared in different classes) access them.

6. Do the problems ***15 to 20 of Assignment-1*** using the concept of class and object.

Assignment-3 (Two Days)

Topic: Polymorphism and Inheritance.

Objective: The objective of this assignment is to understand the concept of Polymorphism & Inheritance.

1. Write a java program to overload three methods named with addition() having different parameters.

2. Write a java program to overload the constructor named with Addition() having different parameters.

3. Overload the constructors for class Box for cube and cone and also display its volume.

4. [*] Repeat problem 3 for method overloading.

5. Create a class Student containing instance variables roll and name and a parameterized constructor. Create two objects of that class from the main class and print them (Hints: Override toString method of Object class).

6. Create a class EMP having instance variable name and id. Create its subclass (say Scientist) which has instance variable no_of_publication and experience. Now create its subclass, say Dscientist which has an instance variable award. Put a method: public String toString () { } in every class where you describe about the class and from the main create object of each class and print each object.

7. Create a class with a method void show () and make 3 subclasses of it and all subclasses have void show () method overridden and call those methods using their class references.

8. [*] Do the problem 6 using dynamic method dispatching.

Assignment-4 (Two Days)

Topic: Abstract class and abstract method

Objective: The objective of this assignment is to understand abstract methods, class and their uses.

1. Check without having any abstract method/s whether a class can be abstract; if so, then use that concrete method/s from another class having the main method.

2. Write a Java Program to implement the following:
Create an abstract base class TwoD that contains two instance variable length and breadth and an abstract method area(). Three subclasses triangle, square and rectangle inherits the base class TwoD and override the method area() according to their shape.

3. Write a Java Program to create a base class Animal having properties**:** consume food, capable of motion.
Two subclass Mammal and Reptile having features: covered with hairs and feed breast milk to its infants, can crawl have scales and cold blooded respectively.
Show the properties of Human and cow for Mammal category, snake and crocodile for reptile category adding one new feature for each of the classes.

4. Assume that a bank maintains two types of account for its customers, one called a savings account and other called a current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides a check book facility but no interest. Current account holders should also maintain a minimum balance (say Rs. 1000) and if the balance falls below this level a service charge is imposed (say Rs. 100).
Create a class **Account** that stores customer name, account number and type of account. From this class derive two classes **Curr_Acct** and **Savn_Acct** respectively to make them more specific to their requirements.

Include the necessary methods to achieve the following tasks:

a. Accept deposits from a customer and update the balance.
b. Display the balance.
c. Compute and deposit interest.
d. Permit withdrawal and update the balance.
e. Check for minimum balance, impose penalty, if necessary, and
update the balance. Use constructors to initialize the class members.

Assignment-5 (Two Days)

Topic: Array

Objective: The objective of this assignment is to understand the concept of **array** in java.

1. Write a Java Program to search an element in an array using Binary Search.
2. Write a Java Program to swap between the highest and lowest element in an array.
3. Write a java program that finds the second smallest element in an array and eliminate it.
   (For example, I/P: [9,3,1,6,4] O/P: [9,1,6,4]).
4. Write a Java program that rotates an array to the left by a given number of positions.
   For example, if an array is {1, 2, 3, 4, 5} and the rotation value is 2, then the output should be {3, 4, 5, 1, 2}.

5. Write a Java Program to implement:
   a) Bubble Sort
   b) Quick Sort
   c) Merge Sort

6. Write a Java Program to perform the following operation in matrix:
   a) Addition
   b) Multiplication
   c) Transpose

7. Write a Java Program to demonstrate irregular array.
8. Write a Java Program to demonstrate anonymous array.