



Object Oriented Programming using Java Laboratory (DJS22FEL22)

F.Y B. Tech, Semester: II

Experiment list

(AY: 2023-24)

1. To implement Java control statements, loops, and command line arguments

- a. Given an integer, n, perform the following conditional actions:
 - If n is odd, print Weird
 - If n is even and in the inclusive range of 2 to 5, print Not Weird
 - If n is even and in the inclusive range of 6 to 20, print Weird
 - If n is even and greater than 20, print Not Weird
- b. WAP to find largest of 3 numbers using nested if else and nested ternary operator.
- c. Write a Java program that reads a positive integer from **command line** and count the number of digits the number (less than ten billion) has.
- d. Write a menu driven program using switch case to perform mathematical operations.
- e. WAP to find grade of student from input marks using if else ladder and switch case.
- f. WAP to print the sum of following series $1+1/2^2+1/3^2+1/4^2+\dots+1/n^2$
- g. WAP to display the following patterns:

```
1
2  1
1  2  3
4  3  2  1
1  2  3  4  5
6  5  4  3  2  1
1  2  3  4  5  6  7
```

A

CB

FED

JHGH



2. To implement Arrays

- a. You have been given an array of positive integers A_1, A_2, \dots, A_n with length N and you have to print an array of same length (N) where the values in the new array are the sum of every number in the array, except the number at that index.

i/p 1 2 3 4

For the 0th index, the result will be $2+3+4=9$, similarly for the second, third and fourth index the corresponding results will be 8, 7 and 6 respectively.

i/p 4 5 6

o/p 11 10 9

- b. The annual examination results of 5 students are tabulated as follows:

Roll No	Subject1	Subject2	Subject3

WAP to read the data and determine the following

Total marks obtained by each student

The student who obtained the highest total marks

- c. WAP to display following pattern using irregular arrays (jagged arrays).

1

1 2

1 2 3

3. To implement Strings

- a. WAP to find out number of uppercase & lowercase characters, blank spaces and digits from the string.
- b. WAP to count the frequency of occurrence of a given character in a given line of text.
- c. WAP to check if a string is a palindrome or not using inbuilt functions.

4. To implement collections (Array List/ Vectors)

- a. WAP to accept students name from command line and store them in vector.



- b. WAP to add n strings in a vector array. Input new string and check if it is present in the vector.
If present delete it else add to the vector

5. To implement class with members and methods (static, non-static, recursive and overloaded methods)

- Create a class employee with data member's empid, empname, designation and salary. Write methods getemployee() to take user input, showgrade() to display grade of employee based on salary, showemployee() to display details of employee.
- WAP to display area of square and rectangle using the concept of **overloaded functions**.
- WAP to find value of y using recursive function (static), where $y=x^n$
- WAP to count the number of objects made of a particular class using static variable and static method and display the same.

6. To implement array of objects and passing/ returning objects

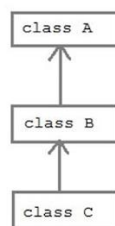
- WAP to arrange the names of students in descending order of their total marks, input data consists of students details such as names, ID.no, marks of maths, physics, chemistry. (Use array of objects)
- WAP to perform mathematical operations on 2 complex numbers by passing and returning object as argument. Show the use of this pointer.

7. To implement Constructors and constructor overloading

- WAP to count the no. of objects created of a class using constructors.
- WAP to display area of square and rectangle using the concept of overloaded constructor (use parameterized, non-parameterized and copy constructor).

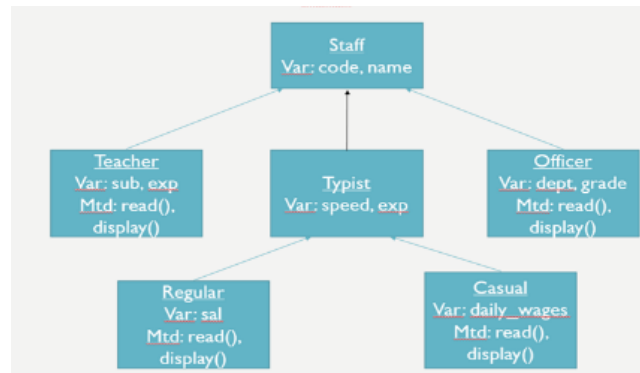
8. To implement Inheritance and super keyword

- WAP to demonstrate the role of Constructors in inheritance in the following class diagram.



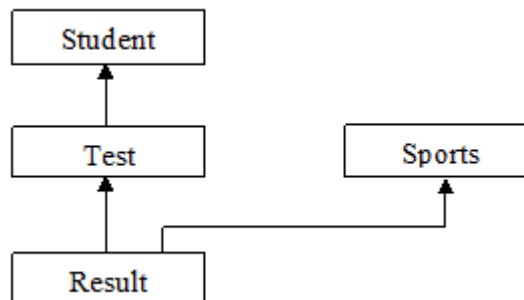


- b. WAP to create a super class having a variable. Let the variable be initialized to some value within a constructor. This class should have a method display () to display the initial value of the variable. Derive a sub class that accesses the constructor, variable and method of the super class using super keyword.
- c. Display data of the specialized classes given in the following class diagram.



9. To implement multiple inheritance using interfaces and method overriding

- a. Design an interface with a method reversal. This method takes a string as input and returns the reversed string. Create a class that implements the above interface.
- b. WAP to implement three classes namely Student, Test and Result. Student class has member as rollno, and read(). Test class has members as sem1_marks and sem2_marks and read(). Result class has member as total. Create an interface named sports that has a member score (). Derive Test class from Student and Result class has multiple inheritances from Test and Sports. Total is formula based on sem1_marks, sem2_mark and score. Use super keyword.



10. To implement dynamic polymorphism, final keyword and garbage collection



- a. Demonstrate using a suitable example that a base class reference variable can point to a child class object or a base class object using the concept of dynamic method dispatch (dynamic polymorphism).
- b. Adwita , 8th Grade student wants to write a functions to calculate simple interest, compound interest. She wants to keep same (final) rate of interest for every input of principal and time. She wants to ensure that the declared functions are not overridden in any subclasses and the class is not inherited by any other class. Help her to declare the variables methods and classes and write the code for the same using final keyword.
- c. WAP to create an object of a class, delete the same object by calling System. gc () and display a message that the “object has been deleted”.

11. To implement Abstract classes and packages

- a. Write an abstract class program to calculate area of circle, rectangle and triangle.
- b. WAP to create a package called vol having Cylinder class and volume (). WAP that imports this package to calculate volume of a Cylinder.

12. To implement exceptions in Java (read input using DataInputStream/ BufferedReader classes)

- a. WAP to implement inbuild exceptions
 - i. Checked exceptions (compile time exceptions)
 - a. ClassNotFoundException
 - b. IOException
 - ii. Unchecked exceptions (run time exceptions)
 - a. NumberFormatException
 - b. ArithmeticException
 - c. ArrayIndexOutOfBoundsException
 - d. NullPointerException
- b. Write a Java Program to Create a User Defined Exception class MarksOutOfBoundsException, If Entered marks of any subject is greater than 100 or less than 0, and then program should create a user defined Exception of type MarksOutOfBoundsException and must have a provision to handle it.



13. To implement Multithreading

- Write a multithreaded program a java program to print Table of Five, Seven and Thirteen using Multithreading (Use Thread class for the implementation).
- Write a multithreaded program to display /*/*/*/*/*/*/* using 2 child threads.
- Write a program to demonstrate thread methods: wait notify suspend resume join setpriority getpriority setname getname
- Write a multithreaded program that generates the Fibonacci sequence. This program should work as follows: create a class Input that reads the number of Fibonacci numbers that the program is to generate. The class will then create a separate thread that will generate the Fibonacci numbers, placing the sequence in an array. When the thread finishes execution, the parent thread (Input class) will output the sequence generated by the child thread. Because the parent thread cannot begin outputting the Fibonacci sequence until the child thread finishes, the parent thread will have to wait for the child thread to finish.
- WAP to prevent concurrent booking of a ticket using the concept of thread synchronization.

14. To implement basic Swing programs with event handling

- Write java program to create a registration form. Take Login id and Password from the user and display it on the third Text Field which appears only on clicking OK button and clear both the Text Fields on clicking RESET button.

- Write a program to create a basic calculator.
- Display the selected fields in Details after submit button is clicked