Department of Computer Application
RCC Institute of Information Technology, Kolkata
Canal South Road, Kolkata - 700015

Assignment No.:1
Paper Name: OOP with Java **Paper Code: MCAN 293**
Semester: 1st Year 2nd Semester **Academic Session: 2022-2023**

LAB ASSIGNMENT

1. Write a Java program to print the largest value to be stored in an integer variable and to display the size occupied by a variable of type long.
2. Write a Java program that reads in two dates (using three integers for each) and displays a message showing the date that comes first.
3. Write a program in Java to solve the following problem -
   You are given a sorted (either in the increasing or in the decreasing order) sequence of numbers, ending with a -1. You can assume that are at least two numbers before the ending -1. You have to output the number of distinct elements in the sorted sequence. **Do not use arrays in the code.**
4. Write a program in Java to check if a given number is present in the last digit(s) of its square.
   For example, 25 is a such number as its square is 625 and 25 is present as the last digits. Few more example are: $5^2 = 2\mathbf{5}$, $6^2 = 3\mathbf{6}$, $76^2 = 57\mathbf{76}$, and $890625^2 = 79321\mathbf{890625}$, etc.
5. Write a program in Java to add two fractions and display their sum in the simplest form
   Examples Test cases:

```
Input: 1 2 3 2
Output: 2 1
Input:  1 3 3 9
Output: 2 3
Input: 1 5 3 15
Output: 2 5
```

6. Write a java program that prints the numbers that do not appear in Fibonacci series. Number of such terms to be printed should be given by the user.
7. Write a java program to print the second largest number among a list of numbers.
8. Euler's number e is used as the base of natural logarithms. It may be approximated using the following formula:

$$e = \frac{1}{0!} + \frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \ldots + \frac{1}{(n-1)!} + \frac{1}{n!}$$

   where *n* is sufficiently large. Write a program that approximates *e* using a loop that terminates when the difference between two successive values of *e* is less than 0.0000001.
9. Write a java program that reads a collection of positive and negative numbers and multiplies only the positive integers. Loop exit should occur when three consecutive negative values are read.
10. Write a java program to find the LCM of two given integers.