ASSIGNMENT-4 DATE:11/7/24

1. Find the LCM and GCD of n numbers? Sample Input: N value = 2 Number 1 = 16 Number 2 = 20 Sample Output: LCM = 80 GCD = 4

CODE:

```
import java.util.Scanner;
public class Main {
 public static void main(String[] args) {
 Scanner input = new Scanner(System.in);
 System.out.print("Enter the number of elements: ");
 int n = input.nextInt();
 int[] numbers = new int[n];
 System.out.println("Enter the numbers:");
 for (int i = 0; i < n; i++) {
 numbers[i] = input.nextInt();
 }
 int gcd = numbers[0];
 int lcm = numbers[0];
 for (int i = 1; i < n; i++) {
 gcd = findGCD(gcd, numbers[i]);
 lcm = findLCM(lcm, numbers[i]);
 }
 System.out.println("GCD of the numbers is: " + gcd);
 System.out.println("LCM of the numbers is: " + lcm);
 }
 public static int findGCD(int a, int b) {
 if (b == 0) {
return a;
 }
 return findGCD(b, a % b);
 }
 public static int findLCM(int a, int b) {
 return (a * b) / findGCD(a, b);
 }
}
```

OUTPUT:

2.Write a program using function to calculate the simple interest. Suppose the customer is a senior citizen. He is being offered 12 percent rate of interest; for all other customers, the ROI is 10 percent.

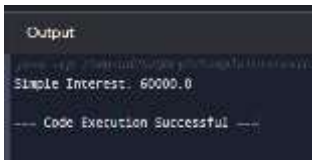
Sample Input:

Enter the principal amount: 200000
Enter the no of years: 3
Is customer senior citizen (y/n): n

CODE:

```
public class SimpleInterestCalculator {
   public static void main(String[] args) {
      double principal = 200000;
      int years = 3;
      char isSeniorCitizen = 'n';
      double rateOfInterest = (isSeniorCitizen == 'y') ? 0.12 : 0.10;
      double simpleInterest = (principal * rateOfInterest * years);
      System.out.println("Simple Interest: " + simpleInterest);
   }
}
```

OUPUT:



3. Write a program to print the Fibonacci series.

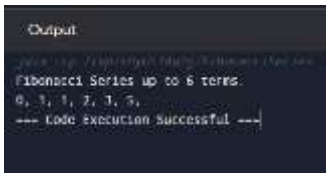
Sample Input:

Enter the n value: 6

CODE:

```
public class FibonacciSeries {
      public static void main(String[] args) {
         int n = 6, firstTerm = 0, secondTerm = 1;
         System.out.println("Fibonacci Series up to " + n + " terms:");
         for (int i = 1; i <= n; ++i) {
            System.out.print(firstTerm + ", ");
            int nextTerm = firstTerm + secondTerm;
            firstTerm = secondTerm;
            secondTerm = nextTerm;
         }
      }
   }
```

OUTPUT:



4. Java Program to Find Even Sum of Fibonacci Series Till number N?

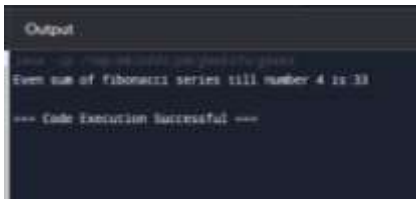
Sample Input: n = 4

Sample Output: 33

CODE:

```
import java.io.*;
class geeksforgeeks {
    static int Fib_Even_Sum(int N)
    {
            if (N <= 0)
                    return 0;
            int fib[] = new int[2*N+1];
            fib[0] = 0;
            fib[1] = 1;
            int s = 0;
            for (int j = 2; j <= 2 * N; j++) {
                    fib[j] = fib[j - 1] + fib[j - 2];
                    if (j % 2 == 0)
                            s += fib[j];
            }
            return s;
    }
    public static void main(String[] args)
    {
            int N=4;
            System.out.println(
                    "Even sum of fibonacci series till number " + N
                    + " is:" + +Fib_Even_Sum(N));
    }
}
```

OUTPUT:



5. Write a program to print the numbers from M to N by skipping K numbers in between?

Sample Input:

M = 50

N = 100

K = 7

Sample Output:

50, 58, 66, 74, …..

CODE:

```
public class SkipNumbers {
   public static void main(String[] args) {
      int M = 50;
      int N = 100;
      int K = 7;
      for (int i = M;i<=N;i+=K) {
         System.out.print(i+", ");
      }
   }
}
```

OUTPUT:

6.Write a program to print all the composite numbers between a and b?

Sample Input:

A = 12

B = 19

Sample Output

14, 15, 16, 18

CODE:

```
public class CompositeNumbers {
   public static void main(String[] args) {
      int A = 12;
      int B = 19;
      for (int i = A; i <= B; i++) {
         if (isComposite(i)) {
            System.out.print(i + ", ");
```

```
            }
        }
    }
    public static boolean isComposite(int num) {
        if (num <= 1) {
            return false;
        }
        for (int i = 2; i < num; i++) {
            if (num % i == 0) {
                return true;
            }
        }
        return false;
    }
}
```

OUTPUT:

7. Find the factorial of n?

Sample Input:

N = 4

Sample Output:

4 Factorial = 24

CODE:

```
import java.util.Scanner;
public class Main {
public static void main(String[] args) {
    int n = 4;
    int factorial = 1;
    for (int i = 1; i <= n; i++) {
        factorial *= i;
    }
    System.out.println(n + " Factorial = " + factorial);
}
}
```

OUTPUT:

8. Find the year of the given date is leap year or not

Sample Input:

Enter Date: 04/11/1947

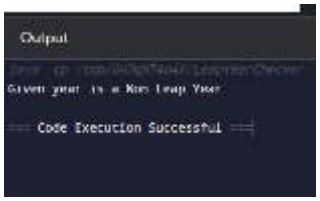
Sample Output:

Given year is Non Leap Year

CODE:

```
import java.util.Scanner;
public class LeapYearChecker {
   public static void main(String[] args) {
      String date = "04/11/1947";
      int year = Integer.parseInt(date.substring(date.lastIndexOf("/") + 1));
      if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0)) {
         System.out.println("Given year is a Leap Year");
      } else {
         System.out.println("Given year is a Non Leap Year");
      }
   }
}
```

OUTPUT:



9. Find the number of factors for the given number

Sample Input:

Given number: 100

Sample Output:

Number of factors = 9

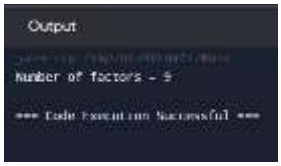
CODE:

```
public class Main {
   public static void main(String[] args) {
      int number = 100;
```

```
            int count = 0;
            for (int i = 1; i <= number; ++i) {
               if (number % i == 0) {
                  count++;
            }

               }
            System.out.println("Number of factors = " + count);
         }
      }
```

OUTPUT:



10. Write a program to print the given number is Perfect number or not?
    Sample Input:
        Given Number: 6
    Sample Output:

It's a Perfect Number

CODE:

```
public class PerfectNumber {
  public static void main(String[] args) {
    int givenNumber = 6;
    if (isPerfectNumber(givenNumber)) {
      System.out.println("It's a Perfect Number");
    }
  }
  public static boolean isPerfectNumber(int number) {
    int sum = 0;
    for (int i = 1; i < number; i++) {
      if (number % i == 0) {
        sum += i;
      }
    }
    return sum == number;
  }
}
```

OUTPUT: