Experiment Number: 2

Student Name: Garv Khurana **UID: 21BCS6615**

Branch: CSE - AIML **Section/Group: 21AML - 9 -A**

Semester: 3rd

Subject Name: Programming in Java **Subject Code: 21CSH-244**

Program 1

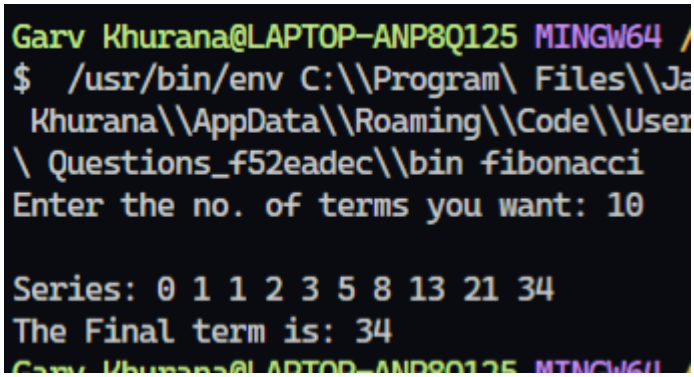
AIM: Display Fibonacci Series using Loop.

Program:

```
import java.util.*;
class fibonacci {
   static int fibo(int terms){
        int a = 0;
        int b = 1;
        int term = 0;
        if(terms == 1){
            System.out.print("\nSeries: "+ a);
            return a;
        } else if(terms == 2){
            System.out.print("\nSeries: " + a + " " + b);
            return b;
        } else{
            System.out.print("\nSeries: " + a + " " + b + " ");
```

```
                for(int i = 0; i < terms - 2; i++){
                        term = a + b;
                        a = b;
                        b = term;
                        System.out.print(term + " ");
    }
                return term;
            }
        }
        public static void main(String args[]){
            System.out.print("Enter the no. of terms you want: ");
            Scanner sc = new Scanner(System.in);
            int terms = sc.nextInt();
            int newTerm = fibo(terms);
            System.out.print("\nThe Final term is: " + newTerm);
        }
    }
```

Output:



Program 2

AIM: Display Fibonacci Series using recursion.

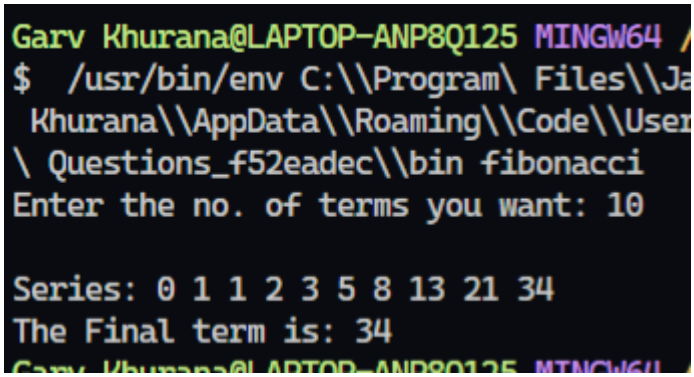
Program:

```
import java.util.*;
class fibonacci {
    static int recursive_fibo(int terms) {
        if (terms == 0 || terms == 1) {
            return terms;
        } else {
            return recursive_fibo(terms - 1) + recursive_fibo(terms - 2);
        }
    }
    public static void main(String args[]) {
        System.out.print("Enter the no. of terms you want: ");
        Scanner sc = new Scanner(System.in);
        int terms = sc.nextInt();
        int newTerm = recursive_fibo(terms - 1);

        System.out.print("\nThe Final term is: " + newTerm);

    }
}
```

Output:



Learning outcomes (What I have learnt):

1. **JAVA Syntax**
2. **Java Operators**
3. **Java Conditionals**
4. **Java Functions**
5. **Java Loops**

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty) :

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.	Viva		10
2.	Performance		12
3.	Worksheet		8