

Міністерство освіти і науки України  
Національний університет «Львівська політехніка»  
Інститут комп'ютерних наук та інформаційних технологій  
Кафедра автоматизованих систем управління



**Звіт**  
до лабораторної роботи №1  
з дисципліни  
*Прикладне програмування*  
на тему:  
**«Основи Java»**

Виконав: студент КН-206  
**Филипчук Богдан**

Прийняв: Скрибайло-Леськів Д. Ю.

# **Лабораторна робота № 1**

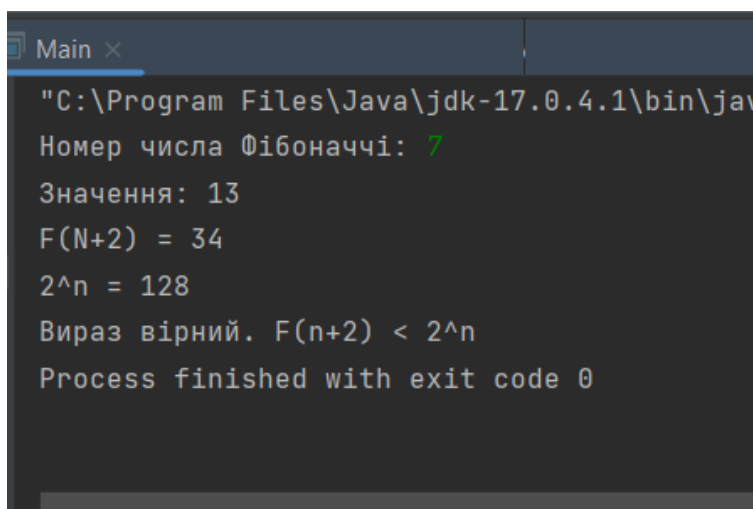
## **Варіант №26**

**Тема роботи:** Основи Java

### **Завдання лабораторної роботи**

1. Згідно з індивідуальним завданням напишіть програму (з використанням об'єктно-орієнтованого підходу), що відповідає наступним вимогам:
  - Програма визначає і опрацьовує задані числа Фібоначчі (1, 1, 2, 3, 5, ...) або Люка (1, 3, 4, 7, 11, ...). Номер числа та інші вхідні дані користувач має можливість передати через командний рядок, або в іншому випадку вводить з клавіатури.
  - Створіть клас, який містить дані про номер та значення числа Фібоначчі або Люка, методи доступу до даних та інші методи, відповідно до індивідуального завдання.
  - У головній функції оголошіть об'єкти або масив об'єктів створеного класу, які зберігають та опрацьовують введені дані.
  - Програма виводить на екран вхідні дані і результати обчислень та перевірок із відповідними підказками.
  - Скомпілюйте та запустіть програму через командний рядок та за допомогою обраного середовища розробки.
2. Продокументуйте програму з п. 1 з використанням JavaDoc і згенеруйте на основі нього документації

**Індивідуальне завдання:** Визначити  $N$ -е число Фібоначчі і перевірити, чи  $F_{n+2} < 2^n$



```

Main x
"C:\Program Files\Java\jdk-17.0.4.1\bin\jav
Номер числа Фібоначчі: 7
Значення: 13
F(N+2) = 34
2^n = 128
Вираз вірний. F(n+2) < 2^n
Process finished with exit code 0

```

```

import java.util.ArrayList;

/**
 * Клас для представлення чисел Фібоначчі.
 * Це функція, у котрій кожне наступне число є сумою двох попередніх.
 *
 * @author Богдан Филипчук
 */
public class FibonacciNumb {
    /**
     * Масив чисел Фібоначчі
     */
    private ArrayList<Integer> numbers = new ArrayList<Integer>();

    public FibonacciNumb() {
        numbers.add(1);
    }

    /**
     * Метод шукає N-е число Фібоначчі, де N - натуральне число
     *
     * <p>
     * <strong>Увага:</strong> N - повинна бути більше нуля.
     *
     * @param N порядковий номер числа Фібоначчі
     * @return N-е число Фібоначчі
     */
    public int getNumbers(int N) {
        if(N < 1) {
            System.out.println("Неправильно введений порядковий номер");
            return 0;
        }
        if (N > numbers.size())
            FillArray(N);
        return numbers.get(N-1);
    }

    /**
     * Заповнює поле класу <code>FibonacciNumb</code> числами Фібоначчі.
     *
     * <p>
     * Так як поле класу <code>FibonacciNumb</code> це екземпляр класу
     * <code>ArrayList</code>,
     * то доцільно заповнювати масив кількістю чисел, до числа, яке запрошує
     * користувач.
     * Це є обов'язковим для розрахунку всі чисел Фібоначчі, оскільки кожен
     * наступний член масиву
     * є сумою двох попередніх.
     *
     * @param N порядковий номер числа Фібоначчі, до якого слід заповнити масив
     * @see ArrayList
     */
    public void FillArray(int N) {
        while (numbers.size() < N) {
            int secNumber;
            if (numbers.size() == 1)
                secNumber = 0;
            else
                secNumber = numbers.get(numbers.size() - 2);

            int toAdd = numbers.get(numbers.size() - 1) + secNumber;
            numbers.add(toAdd);
        }
    }
}

```

```

import java.util.Scanner;

//26-ий варіант
public class Main {
    /**
     * Перевіряє вираз
     * <p>
     * Вираз -  $F(N+2)$  менше  $2^N$ , де
     * <p>
     *  $F(N)$  - формула яка пояснює числа Фібоначчі
     * <p>
     *  $N$  - порядковий номер числа Фібоначчі
     *
     * @param fibonacciNumb екземпляр класу <code>FibonacciNumb</code>
     * @param N порядковий номер числа Фібоначчі, яке запрошує користувач
     * @return true якщо вираз вірний, false - якщо ні
     */
    public static boolean CheckExpression(FibonacciNumb fibonacciNumb, int N) {
        int first = fibonacciNumb.getNumbers(N + 2);
        System.out.println("F(N+2) = " + first);
        int second = 2 << N;
        System.out.println("2^n = " + second);
        return first < second;
    }

    public static void main(String[] args) {
        FibonacciNumb fibonacciNumb = new FibonacciNumb();
        Scanner scan = new Scanner(System.in);

        System.out.print("Номер числа Фібоначчі: ");
        int N = scan.nextInt();

        int get = fibonacciNumb.getNumbers(N);
        if (get == 0)
            return;
        System.out.println("Значення: " + get);

        if (CheckExpression(fibonacciNumb, N))
            System.out.print("Вираз вірний. F(n+2) < 2^n");
        else
            System.out.println("Вираз невірний. F(n+2) > 2^n");
    }
}

```

## Висновок

Метою даної лабораторної роботи було вивчити основи мови програмування Java. Для обчислення ряду чисел Фібоначчі описав окремий клас **FibonacciNumb**, який містить два методи: метод **getNumbers**, що повертає число Фібоначчі за порядковим номером, та метод **FillArray**, що заповнює динамічний масив числами Фібоначчі. Для перевірки виразу  $F_{n+2} < 2^n$  описав метод **CheckExpression**. Програма працює коректно, помилок не виникає.