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-203

**Панчук Олександр**

Прийняв: асист. каф. АСУ

**Скрибайло-Леськів Д. Ю.**

Львів – 2022

***Лабораторна робота № 1***

**Тема роботи:** **“Основи Java”**

**Мета роботи:** навчитися імплементовувати класи, методи та принципи ООП в об’єктно-орієнтованій мові програмування Java.

**Завадання**

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 оголосіть 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 документацію.

**Індивідуальне завдання (варіант 18)**

*Визначити, які числа серед перших N чисел Люка закінчуються на задану цифру.*

1. Я ознайомився з поняттям чисел Люка та визначив алгоритм їх знаходження.

The first few Lucas numbers are 2, 1, 3, 4, 7, 11, 18, 29, 47, 76, 123, 199, 322, 521, 843, 1364, 2207, 3571, 5778, 9349 ....

N повино бути иеншим рівним 90, тому що число ряду чисел Люка під заданм N - це максимально допусиме значення для типу long.

1. Код програми:

package lpnu.ap.lab1;  
  
import java.util.Arrays;  
import java.util.Scanner;  
  
public class LucasNumber {  
 private int num;  
 private long valueNum;  
 private int digit;  
 public LucasNumber(int num, int digit) {  
 this.num = num;  
 this.digit = digit;  
 getResult();  
 }  
  
 public int getNum() { return num; }  
  
 public int getDigit() { return digit; }

public long getValueNum() { return valueNum; }  
  
 public long [] getResult() {  
 long [] array = new long[num];  
 long n1 = 1, n2 = 3;  
 int j = 0;  
 if (n1 == digit) array[j++] = n1;  
 if (n2 == digit) array[j++] = n2;  
  
 for (int i = 0; i < num - 2; i++) {  
 long temp = n1;  
 n1 = n2;  
 n2 += temp;  
  
 if (n2 % 10 == digit) array[j++] = n2;  
 }  
 this.valueNum = n2;  
 long [] result = Arrays.*copyOf*(array, j);  
 return result;  
 }  
  
 public static void main(String[] args) {  
  
 int indexLimit = 100;  
 int index = 0;  
 LucasNumber [] numbers = new LucasNumber[indexLimit];  
  
 System.*out*.println("Welcome to LucasNumbers programme!!! (N <= 90) Input '-1' to finish!");  
 while (true) {  
  
 Scanner scanner = new Scanner(System.*in*);  
 System.*out*.print("Input 'N': ");  
 int N = scanner.nextInt();  
 if (N < 0 || index + 1 == indexLimit || N > 90) break;  
 System.*out*.print("Input 'digit': ");  
 int digit = scanner.nextInt();  
  
 numbers[index++] = new LucasNumber(N, digit);  
 }  
  
 for (int i = 0; i < index; i++) {  
 System.*out*.println("\nTest " + (i + 1) +

"\n\tValue of ( N = " +

numbers[i].getNum() + " ) = " +

numbers[i].getValueNum() +

"\n\tDigit = " +

numbers[i].getDigit()+

"\nResult: ");

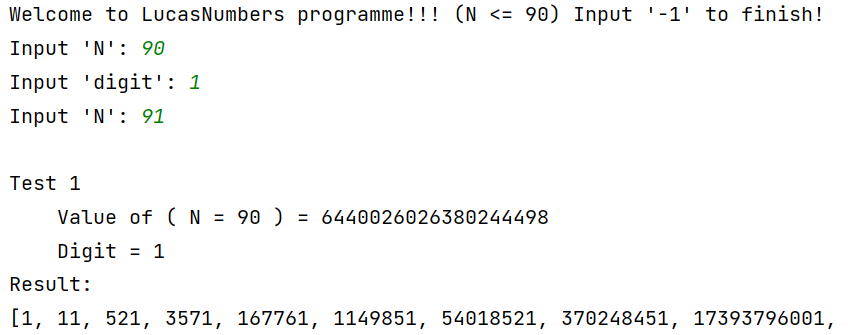
if (numbers[i].getResult().length != 0)

System.*out*.println(Arrays.*toString*(numbers[i].getResult()));  
 else  
System.*out*.println("There are no such numbers that end in a given digit!");

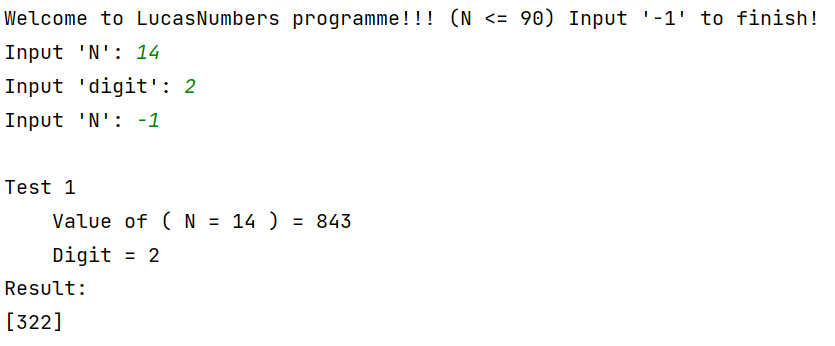
}  
 }  
}

Посилання на проект у [github.com/olex4ndr/AP-Lab-1](https://github.com/olex4dr/AP-Lab-1)

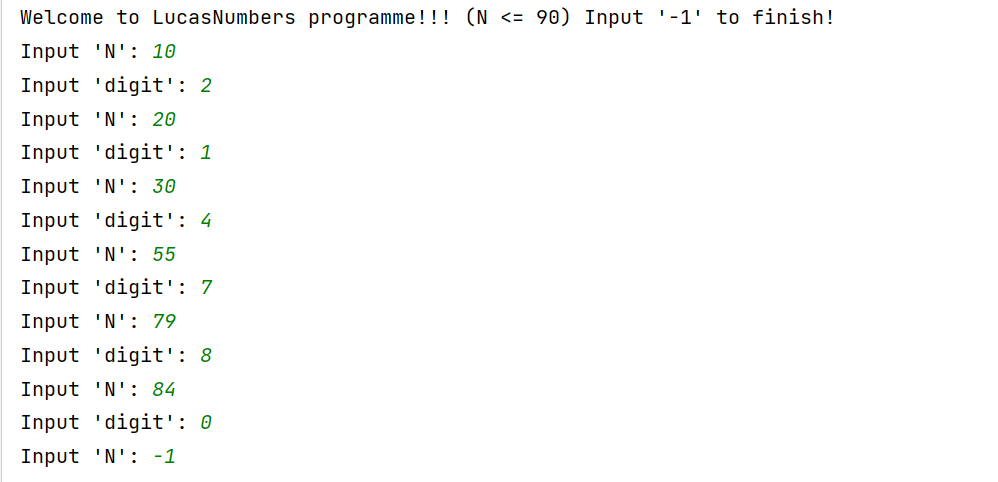
1. Результати виконання програми:
2. Перевірка виходу з циклу при введені некоректних даних (N > 90):

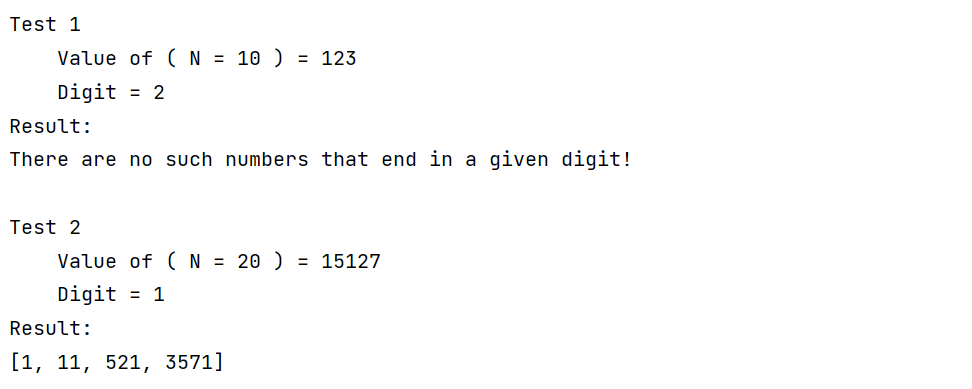


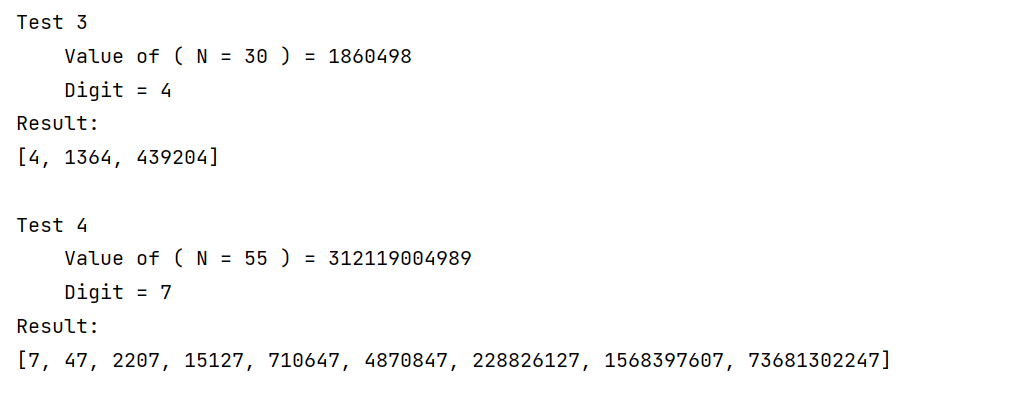
1. Перевірка виходу з циклу за бажанням користувача (N = -1):

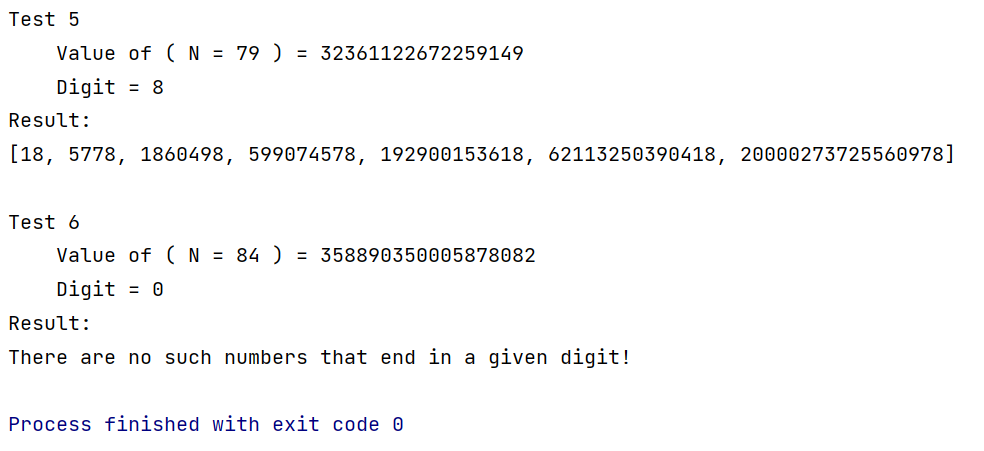


1. Введення довільних допустимих значень N:

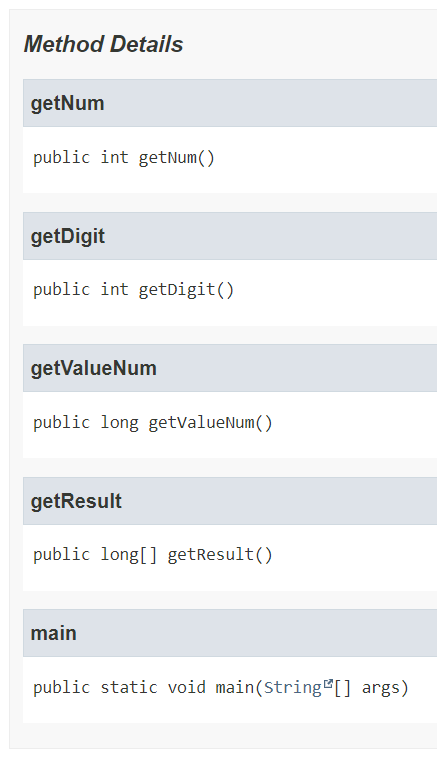
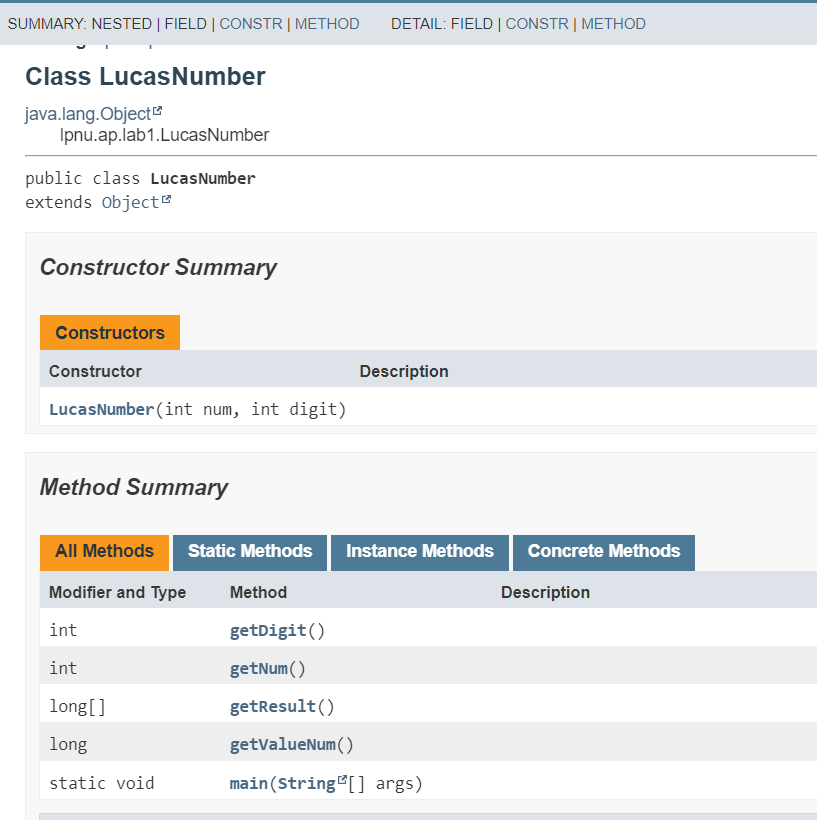


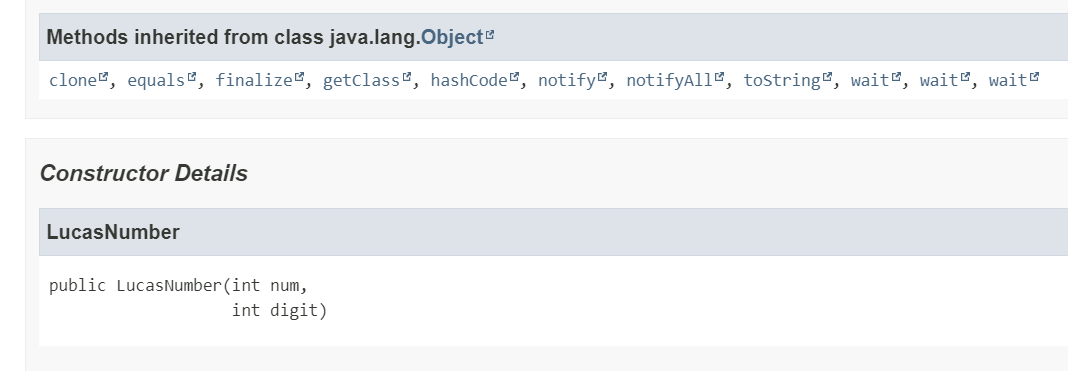






1. Далі, використовуючи засоби Intellij IDEA я згенерував “javadoc”.





**Висновок**

У процесі виконання лабораторної роботи я навчився імплементовувати класи, методи та принципи ООП в об’єктно-орієнтованій мові програмування Java.