

Лабораторна робота №2.

ПРОСТІ АЛГОРИТМИ. КОНСОЛЬНІ ПРОГРАМИ

Мета роботи:

Розробка простих консольних програм мовою Java.

1. Завдання:

- 1) Знайти найбільший спільний дільник двох цілих позитивних чисел.
- 2) Знайти суму цифр заданого цілого числа.
- 3) Знайти найбільшу цифру в десятковому запису цілочисельного значення.
- 4) Знайти позиції всіх найменших цифр в десятковому запису цілочисельного значення.
- 5) Перевірити, чи є задане число простим (тобто не ділиться без залишку на жодні числа, крім себе і 1).
- 6) Перевірити чи дорівнює сума перших трьох цифр сумі останніх трьох цифр в десятковому запису 6-значного цілого числа.
- 7) Перевірити чи рівні значення першої та останньої, а також другої та передостанньої цифри в вісімковому запису 4-значного цілого числа.
- 8) Знайти суму всіх парних і суму всіх непарних цифр в десятковому запису 8-значного цілого числа.
- 9) Підрахувати кількість цифр, що відображаються за допомогою латинських букв в шістнадцятковому запису 10-значного цілого числа.
- 10) Знайти кількість комбінацій 01 та 10 в двійковому запису цілого числа.

1.1 Індивідуальне завдання:

Перевірити, чи є задане число простим (тобто не ділиться без залишку на жодні числа, крім себе і 1).

1.2 Розробник:

студент Литвин Ігнатій Ігорович; КІТ-26А; Варіант №7

1.3 Рекомендації / вимоги до лабораторної роботи:

Розробити та продемонструвати програму мовою Java в середовищі Eclipse для вирішення прикладної задачі за номером, що відповідає збільшеному на одиницю залишку від ділення на 10 зменшеного на одиницю номера студента в журналі групи.

Для визначення вхідних даних використовувати генератор псевдовипадкових чисел (`java.util.Random`) та забезпечити циклічне (принаймні десять ітерацій) знаходження результату рішення прикладної задачі.

Забезпечити виведення до консолі відповідних значень вхідних даних та результатів обчислень у вигляді таблиці.

Забороняється використання даних типу `String` та масивів при знаходженні рішення прикладної задачі.

Застосувати функціональну (процедурну) декомпозицію і забезпечити рішення прикладної задачі за допомогою відповідних статичних методів.

2. Розробка програми

2.1 Опис програми

Дана програма перевіряє чи є задане число простим (тобто не ділиться без залишку на жодні числа, крім себе і 1);

виконує виведення до консолі відповідних значень вхідних даних та результатів обчислень у вигляді таблиці.

У розробці програми застосовано функціональну (процедурну) декомпозицію і забезпечити рішення прикладної задачі за допомогою відповідних статичних методів.

2.2 Важливі фрагменти програми

```
public class Main {  
  
    /**  
     * Виконує поставлене завдання  
     */  
    public static void main(String[] args) {  
        System.out.format("    Число          Чи просте?\n\n");  
        for (int i = 0; i < 21; i++) {  
            findPrimeNumber(randInt());  
        }  
    }  
  
    /**  
     * Повертає псевдо-рандомне число.  
     *  
     * @return Число типу Integer.  
     * @see java.util.Random#nextInt(int)  
     */  
    public static int randInt() {  
        Random rand = new Random();  
        int randomNum = rand.nextInt(Integer.MAX_VALUE);  
        return randomNum;  
    }  
  
    /**  
     * Перевіряє чи є число простим.  
     *  
     * @param number  
     *          Число для перевірки  
     * @return true - число просте | false - ні.  
     */  
    public static boolean isPrimeNumber(int number) {  
        if (number <= 1) {  
            return false;  
        }  
        if (number == 2) {  
            return true;  
        }  
        if (number % 2 != 0) {  
            return true;  
        }  
    }  
}
```

```

    }
    final double ceiledNumberSqrt = Math.ceil(Math.sqrt(number));
    for (long i = 3; i <= ceiledNumberSqrt; i += 2) {
        if (number % i == 0) {
            return false;
        }
    }
    return false;
}

/**
 * Шукає прості числа.
 *
 * @param number
 *         Число для перевірки.
 */
public static void findPrimeNumber(int number) {
    if (isPrimeNumber(number)) {
        System.out.format("%d    Так\n", number);
    } else {
        System.out.format("%d    Hi\n", number);
    }
}
}

```

3. РЕЗУЛЬТАТ РОБОТИ

Для налагодження роботи програми було успішно проведено її тестування.

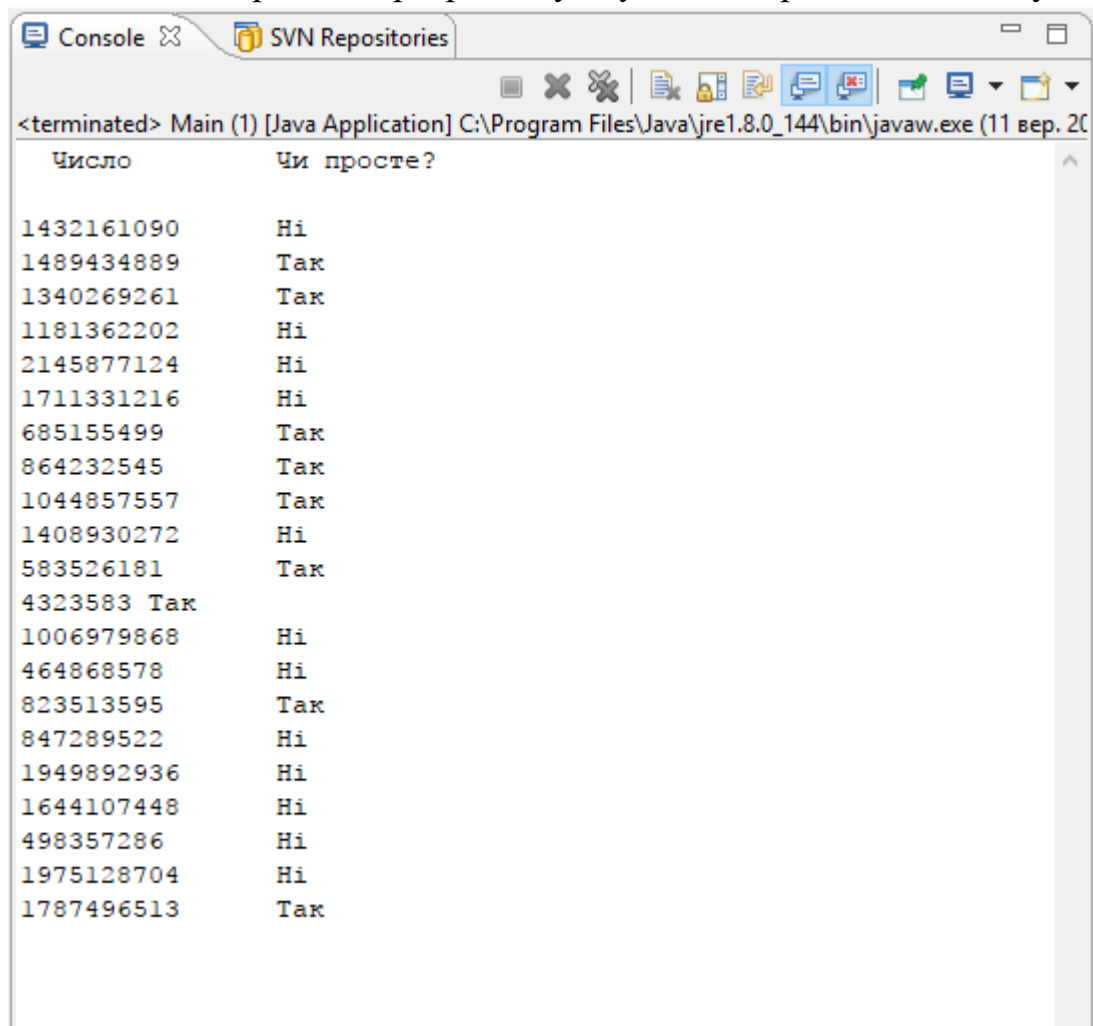


Рисунок 1 "Результат роботи програми"

ВИСНОВКИ

Створено і налагоджено програму, що повністю виконую поставлене індивідуальне завдання та відповідає вимогам.

Було отримано і вдосконалено навички у розробці простих консольних програм мовою Java.