ЛАБОРАТОРНА РОБОТА №2

АЛГОРИТМІЧНА ДЕКОМПОЗИЦІЯ. ПРОСТІ АЛГОРИТМИ ОБРОБКИ ДАНИХ

*Мета:* Розробка простих консольних програм для платформи Java SE.

1 ВИМОГИ

* 1. Розробник
* Соколенко Дмитро Григорович
* група КІТ 102.8а
* варіант 6
  1. Загальне завдання

Вимоги:

1. 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.
2. Для визначення вхідних даних використовувати генератор псевдовипадкових чисел (java.util.Random) та забезпечити циклічне (принаймні десять ітерацій) знаходження результату рішення прикладної задачі.
3. Забезпечити виведення до консолі відповідних значень вхідних даних та результатів обчислень у вигляді таблиці.
4. Застосувати функціональну (процедурну) [декомпозицію](https://ru.wikipedia.org/wiki/%D0%94%D0%B5%D0%BA%D0%BE%D0%BC%D0%BF%D0%BE%D0%B7%D0%B8%D1%86%D0%B8%D1%8F) і забезпечити рішення прикладної задачі за допомогою відповідних методів.
5. Забороняється використання даних типу [String](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html) та [масивів](https://docs.oracle.com/javase/specs/jls/se7/html/jls-10.html) при знаходженні рішення прикладної задачі.

1.3 Задача

Перевірити чи дорівнює сума перших трьох цифр сумі останніх трьох цифр в десятковому запису 6-значного цілого числ

2 ОПИС ПРОГРАМИ

**2.1 Опис логічної структури**

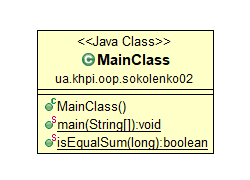


Рис 1 – Діаграма класу

**2.2 Фрагменти коду**

Основна функція:

**public** **static** **boolean** isEqualSum(**long** initialNumber) {

**int** sixthDigit = (**int**)initialNumber % 10;

**int** fifthDigit = ((**int**)initialNumber % 100) / 10;

**int** fourthDigit = ((**int**)initialNumber % 1000) / 100;

**int** thirdDigit = ((**int**)initialNumber % 10000) / 1000;

**int** secondDigit = ((**int**)initialNumber % 100000) / 10000;

**int** firstDigit = ((**int**)initialNumber % 1000000) / 100000;

**int** firstPartSum = firstDigit + secondDigit + thirdDigit;

**int** secondPartSum = fourthDigit + fifthDigit + sixthDigit;

**return** (firstPartSum == secondPartSum);

}

**3 ВАРІАНТИ ВИКОРИСТАННЯ**

**3.1 Ілюстрація роботи програми**

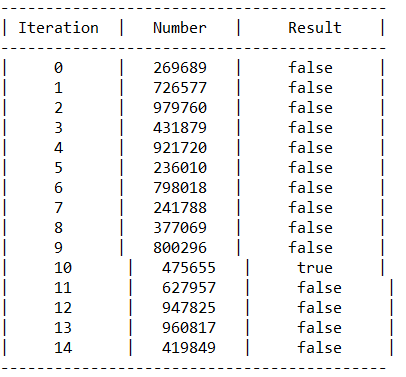


Рис 2 – результат роботи у вигляді таблиці

**ВИСНОВОК**

Отримали навички з розробки простих консольних програм для платформи Java SE.