Тема: Алгоритмічна декомпозиція. Прості алгоритми обробки даних

Мета: Розробка простих консольних програм для платформи Java SE.

**1 ВИМОГИ**

**1.1 Розробник**

Інформація про розробника:

- Гряник Георгій Володимирович

- КІТ-119Д;

- 6 варіант.

**1.2 Загальне завдання**

* 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.
* Застосувати функціональну (процедурну) декомпозицію і забезпечити рішення прикладної задачі за допомогою відповідних методів.
* Забороняється використання даних типу String та масивів 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 числа.

2 ОПИС ПРОГРАМИ

2.1 Засоби ООПВ цій роботі не використовується особливі засоби ООП .  
2.2 Ієрархія та структура класівКлас “ Hrianyk\_Heorhii\_02 ” виконує головне завдання з вирішення завдань лабораторної роботи відповідно номеру варіанту.

2.3 Важливі фрагменти програми **public** **static** **void** TheSumOfThree()///6.!Перевірити чи дорівнює сума перших трьох цифр сумі останніх трьох цифр в десятковому запису 6-значного

{

System.***out***.print("\n\nTask 06");

System.***out***.print("\nВведене випадкове число у якому 6 знакі:");

System.***out***.println("\n+---------+-----------+");

System.***out***.println("|Число |порівняння |");

System.***out***.println("+---------+-----------+");

**for** (**int** i=0;i<10;i++)

{

**int** num = (**int**)(Math.*random*()\*1000000);

**int** a=num;

**int** sum1 = 0;

**int** sum2 =0;

sum1= *sumThree*(num,sum1);

num/=1000;

sum2= *sumThree*(num,sum2);

System.***out***.println("|"+a+" |" + ( sum1 == sum2 ? "дорівнює" : "не дорівнює")+"|");

System.***out***.println("+---------+-----------+");

}

}

**public** **static** **int** sumThree (**int** num,**int** sum)

{

**for** (**int** i = 0; i < 3; i++) {

sum += num % 10;

num /= 10;

}

**return** sum;

}

3 ВАРІАНТИ ВИКОРИСТАННЯ

Функція ” TheSumOfThree ” в циклі перевіряє в циклі 10 разів (для точності перевірки правильності роботи) підраховує перші три та останні цифри числа за допомогою функції «sumThree» та порівнює і виводе результат в таблиці.

Прямого використання немає через те що число генерується випадковим чином

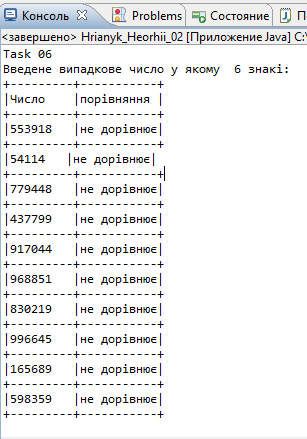


Рисунок 1 – результат роботи перевірки чи дорівнює сума перших трьох цифр сумі останніх трьох цифр в десятковому запису 6-значного цілого числа

ВИСНОВКИ

Під час виконання лабораторної роботи було набуто навички розробка простих консольних програм для платформи Java SE та роботи з консольними функціями.