МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

КАФЕДРА ПРОГРАМНОЇ ІНЖЕНЕРІЇ ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГИЙ УПРАВЛІННЯ

Лабораторна робота №1

З курсу «Якість і тестування програмного забезпечення»

«Модульне тестування. Частина 1. Модульне тестування в Java»

Виконав:

Студент групи КН 36-А

Кулик В.В.

Перевірила:

Бабкова Н. В.

ХАРКІВ 2018

**Цель работы:**

Знакомство с методами и техниками модульного тестирования на реальных примерах программного кода, выполненного Java.

**Код на Java**:

package com.company;

import static java.lang.Math.abs;

public class Main {

public int temp = 0;

public boolean result = false;

public boolean soversh(int num) {

if (num <= 3000 && num >= 0) {

for(int i=1;i<=num/2;i++){

if(num%i == 0){

temp += i;

}

}

if (temp == num) {

//System.out.println(num + " - простое число");

result = true;

return result;

} else {

//System.out.println(num + " - совершенное число");

result = false;

return result;

}

} else if (num > 3000) {

result = false;

return result;

} else {

result = false;

return result;

}

}

public static void main(String [] args){

Main main = new Main();

System.out.println(main.soversh(5));

}

}

**Тестовый класс:**

public class MainTest {

@Test

public void f() {

Main main = new Main();

boolean expected = true;

assertEquals(false, main.soversh( 7 ));

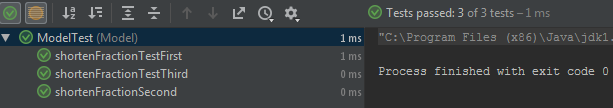
assertEquals(false, main.soversh( 5 ));

assertFalse(expected==main.soversh(6));

}

}

**Результаты тестирования:**



**Выводы:**

Выполняя лабораторную работу было совершено знакомство с методами и техниками модульного тестирования на реальных примерах программного кода, выполненного на языке программирования Java.

По результатам тестирования в данной программе не было обнаружено ошибок, все тесты были выполнены удачно.

По моему мнению, и JUnit и использование механизма исключений хорошо себя проявляют в модульном тестировании, но их можно использовать в разных сценариях, Try-Catch использовать как механизм защиты от случайных ошибок, а JUnit для тестирования конкретных модулей и дальнейшего анализа и устранения найденных ошибок.