Министерство науки и высшего образования Российской Федерации

ФГАОУ ВО «УрФУ имени первого Президента России Б.Н.Ельцина»

ИРИТ-РТФ

Центр ускоренного обучения

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1**

по дисциплине «Введение в специальность»

**Тема:** Схема программы по ГОСТ 19.701-90

Студент группы РИВ-210938у: В.В. Андреев

Преподаватель: С.И. Тимошенко,

доц., к.т.н.

Екатеринбург 2022

**1. Постановка задачи**

1.1 Выбрать свой вариант задания.

**Вариант 1. Ряд Фибоначчи**

1.2 Если в описании алгоритма есть несколько его реализаций на Java, то выбрать один из них, который обеспечивает наилучшее решение по скорости работы.

1.3 Проверить работоспособность выбранной реализации алгоритма в одной из интегрированных сред (Eclipse, NetBeans или IntelliJ IDEA).

1.4 Путем пошаговой отладки в интегрированной среде разобраться в особенностях выбранной реализации алгоритма.

1.5 Используя JUnit, создать тесты для проверки правильности работы основных методов выбранной реализации алгоритма.

1.6 Составить схему программы по ГОСТ 19.701-90 для выбранной реализации алгоритма.

**2 Результаты проделанной работы**

2.1 Описать принцип работы выбранной реализации алгоритма.

Ряд Фибоначчи — это такая последовательность чисел, в которой любое число, кроме первого и второго, является суммой двух предыдущих:

0, 1, 1, 2, 3, 5, 8, 13, 21...

Первое число в последовательности Фибоначчи — 0, четвертое — 2. Отсюда следует, что для получения значения любого числа n в последовательности Фибоначчи можно использовать формулу:

fib(n) = fib(n — 1) + fib(n — 2).

2.2 Привести текст на Java для выбранной реализации алгоритма.

public class Main {

public static void main(String[] args) {

System.out.println(Fib4.fib4(5));

}

}

public class Fib4 {

static int fib4(int n) {

int last = 0, next = 1; // fib(0), fib(1)

for (int i = 0; i < n; i++) {

int oldLast = last;

last = next;

next = oldLast + next;

}

return last;

}

}

2.3 Привести текст тестов JUnit для выбранной реализации алгоритма.

import static org.junit.jupiter.api.Assertions.\*;

class Fib4Test {

@org.junit.jupiter.api.Test

void fib4() {

assertEquals(6765,Fib4.fib4(20));

assertEquals(5,Fib4.fib4(5));

assertEquals(2,Fib4.fib4(3));

}

}

2.4 Привести схему программы по ГОСТ 19.701-90 для выбранной реализации алгоритма и дополнить ее комментариями.

Начало блок-схемы программы представляет собой функцию static int fib4(int n). Затем инициализируются переменные две переменные: last и next с присвоенными значениями. После создаются начало цикла с его название for, тело цикла и условие окончания цикла. Затем программа завершается, возвращая значение last.

Начало

int last = 0;

int next = 1;

for (int i = 0; i < n; i++)

int oldLast = last;

last = next;

next= oldLast+ next;

i < 0

Начало

**3 Выводы**

В данном отчете было изученное построение алгоритма по вычислению ряда Фибоначчи, построение тестов с помощью Junit для проверки корректного вывода алгоритма, а также было изучено построение блок-схем по ГОСТ 19.701-90 для построения схемы алгоритма ряда Фибоначчи.