Санкт-Петербургский политехнический университет Петра Великого

Высшая школа интеллектуальных систем и суперкомпьютерных технологий

**Отчёт по лабораторной работе № 2**

**Дисциплина**: Низкоуровневое программирование

**Тема**: Программирование EDSAC

Выполнил студент гр. 3530901/90003 В.С. Андрианов

(подпись)

Преподаватель А.О. Алексюк

(подпись)

“ ” 2021 г.

Санкт-Петербург

2021

**Содержание**

**1. Техническое задание………………………………………………………... 3**

**2. Метод решения……...………………………………………………………. 3**

**3. Программа Initial Orders 1****………………………………………………… 4**

**4. Результат работы Initial Orders 1………………………………………… 5**

**5. Программа Initial Orders 2…………………………………………………. 6**

**6. Результат работы Initial Orders 2………………………………………….. 8**

**7. Адреса и правила кодирования исходных данных и результатов. Руководство программиста…………………………………………………… 9**

**1. Техническое задание**

1. Разработать программу для EDSAC, реализующую расчет заданного члена последовательности Фибоначчи, и предполагающую загрузчик Initial Orders 1. Массив (массивы) данных и другие параметры (преобразуемое число, длина массива, параметр статистики и пр.) располагаются в памяти по фиксированным адресам.

2. Выделить функциональность (расчет заданного члена последовательности Фибоначчи) в замкнутую (closed) подпрограмму, разработать вызывающую ее тестовую программу. Использовать возможности загрузчика Initial Orders 2. Адрес обрабатываемого массива данных и другие параметры передавать через ячейки памяти с фиксированными адресами.

**2. Метод решения**

Алгоритм расчета заданного члена последовательности Фибоначчи приведен на ЯП Kotlin:

*fun* fib() {  
 *var n* = 5  
 *var f* = 1  
 *var s* = 0  
 *var t* = 0  
 *while* (*n* > 0) {  
 *t* = *f* + *s  
 f* = *s  
 s* = *t  
 n*--  
 }  
 print(*t*)  
}

Алгоритм, реализованный на EDSAC

1. Берем заданное для расчета число (n) из ячейки, в которое оно записано.
2. Вычитаем из него значение шага (h = 1) из ячейки, в которое оно записано.
3. Проверили не отрицательное ли значение имеет n, если отрицательное, то переходим к ячейке выхода, если положительное, то продолжаем выполнение программы.
4. Обновили значение n, то есть записали в ячейку значение, полученное после вычитания единицы.
5. Берем значение первой переменной f (first = 1) из ячейки, в которое оно записано.
6. Берем значение второй переменной s (second = 0) из ячейки, в которое оно записано. Складываем с переменной f, которая уже находится в acc.
7. Записываем результат сложение в ячейку третьей переменной t (third = first + second).
8. Берем значение s.
9. Обновляем ячейку f (f = s)
10. Берем значение t.
11. Обновляем ячейку s (s = t).
12. Продолжаем проход по циклу (начиная с п.1) пока значение n – положительное.
13. Выход из программы.

**3. Программа Initial Orders 1**

На рисунке 1 представлен код программы Initial Orders 1

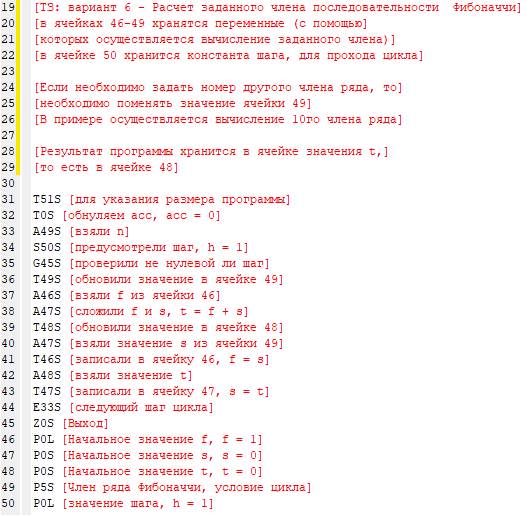


Рис. 1 Код программы Initial Orders 1

В ячейках 46, 47, 48 записаны переменные f = 1, s = 0, t = 0. В ячейку 49 записан заданный для расчета член последовательности Фибоначчи, также эта ячейка является условием цикла. В ячейке 50 записан шаг h = 1.

Результат работы программы будет записан в ячейку переменной t, то есть в ячейку 48.

**4. Результат работы Initial Orders 1**

На рисунке 2 представлен результат вычисления десятого члена последовательности.

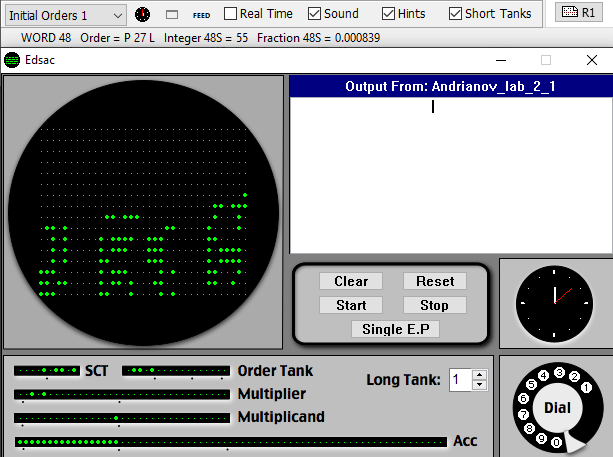


Рис. 2 Результат расчета 10го члена, Initial Orders 1

**5. Программа Initial Orders 2**

На рисунке 3 представлена подпрограмма, реализующая расчет заданного члена последовательности Фибоначчи.

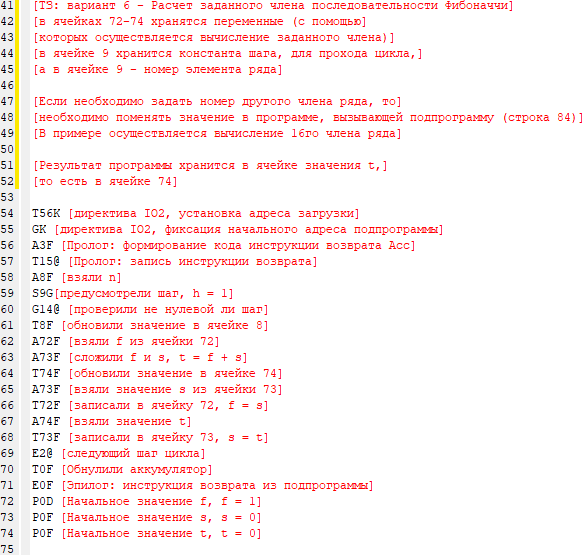


Рис. 3 Подпрограмма (closed)

На рисунке 4 представлена программа, которая вызывает подпрограмму.

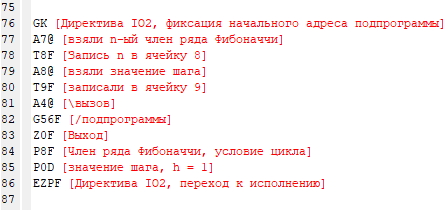


Рис. 4 Программа, вызывающая подпрограмму

В ячейках 72, 73, 74 записаны переменные f = 1, s = 0, t = 0. В ячейку 84 записан заданный для расчета член последовательности Фибоначчи, который, по ходу выполнения программы, будет переписан в ячейку 8. В ячейке 85 записан шаг, который будет переписан в ячейку 9.

Результат работы программы будет записан в ячейку переменной t, то есть в ячейку 74.

**6. Результат работы Initial Orders 2**

На рисунке 5 представлен результат вычисления шестнадцатого члена последовательности.

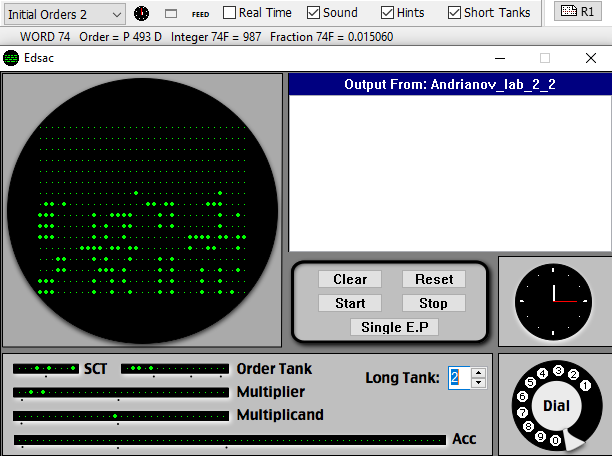


Рис. 5 Результат расчета 16го члена, Initial Orders 2

**7.** **Адреса и правила кодирования исходных данных и результатов. Руководство программиста**

**1. Initial Orders 1**

1) Чтобы воспользоваться программой, необходимо в строку 49 записать член последовательности, который нужно рассчитать.

2) Результат выполнения программы будет записан в переменную t, то есть в ячейку 48.

**2. Initial Orders 2**

1) Чтобы воспользоваться программой, необходимо в строку 84 записать член

последовательности, который нужно рассчитать.

2) Результат выполнения программы будет записан в переменную t, то есть в

ячейку 74.