Санкт-Петербургский политехнический университет Петра Великого

Институт компьютерных наук и технологий

Кафедра компьютерных систем и программных технологий

**Отчёт по лабораторной работе**

**Дисциплина**: Низкоуровневое программирование

**Тема**: EDSAC

Выполнила студентка гр. 3530901/90004

Р.И. Галиева

(подпись)

Преподаватель А.О.Алексюк

(подпись)

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 г.

Санкт-Петербург

2021

1. **Формулировка задачи**
   1. Разработать программу для EDSAC, реализующую определение медианы in-place (Вариант 6), 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 определение медианы in-place 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. **Описание работы.**
   1. **Используемый алгоритм**

Counter – счетчик

i – индексация во внешнем цикле

j – индексация во внутреннем цикле

**Псевдокод алгоритма:**

int[] a = new int[];

int counter ;

int res = 0;

for(int i =0; i < a.lenght; i++){

counter = 0;

for(int j=0;j<a.length;j++){

if(a[j] – a[i] < 0)counter += 1;

if(a[j] – a[i] > 0) counter -= 1;

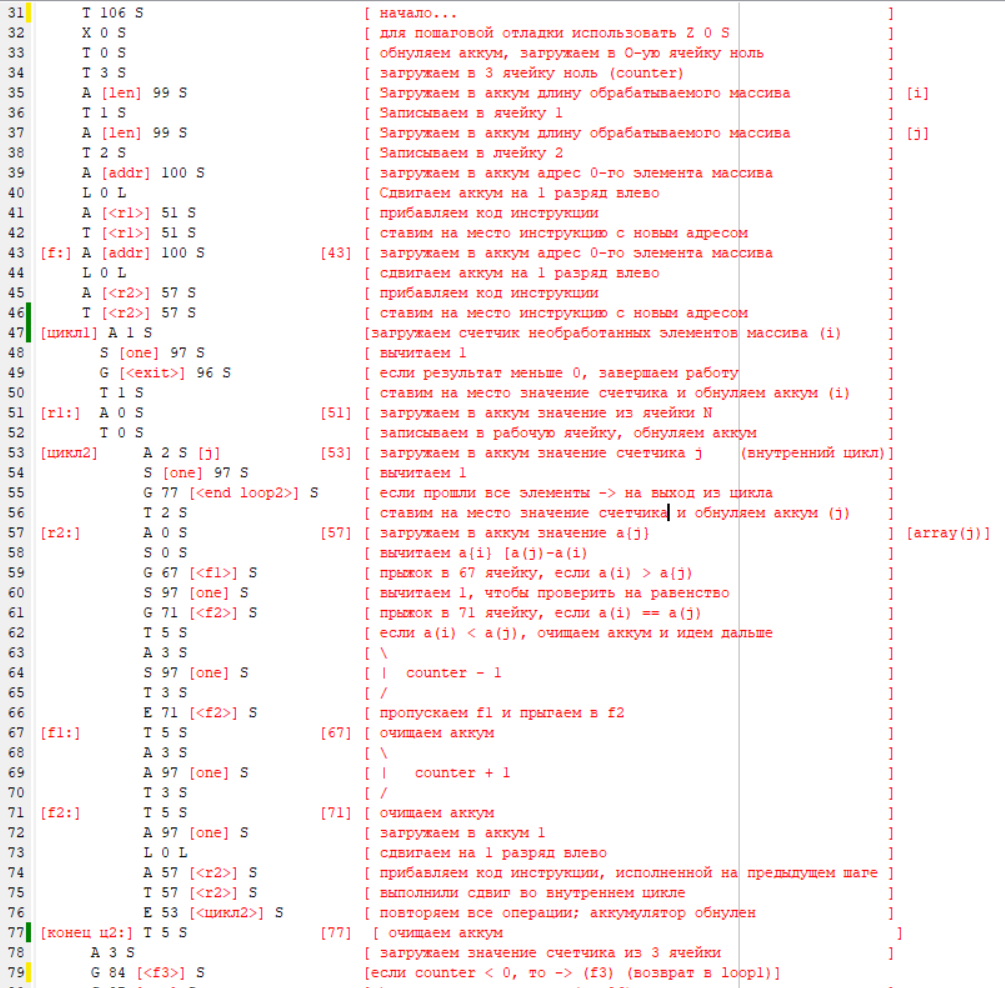
}

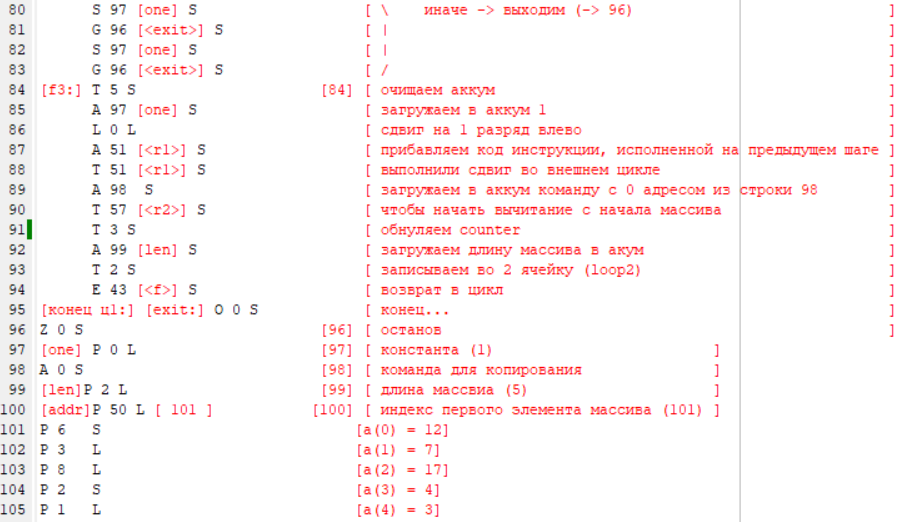
if(counter == 0) res = a[i]

}

* 1. **Код программы Initial Orders 1**

В IO1 инструкции загружаются в ячейки 0-30, программа начинает работу с 31 ячейки. Первая инструкция программы обязательно должна иметь вид T <N+1> S, где N - адрес последней инструкции программы, S - код длины слова. S обозначает "короткое слово" (17 бит), L - "длинное" слово (35 бит).





|  |  |
| --- | --- |
| Нечетное кол-во элементов.  Array = [12,7,17,4,3]  Result = 7 |  |
| Четное кол-во элементов  Array = [12,7,17,4,]  Result = 12 |  |

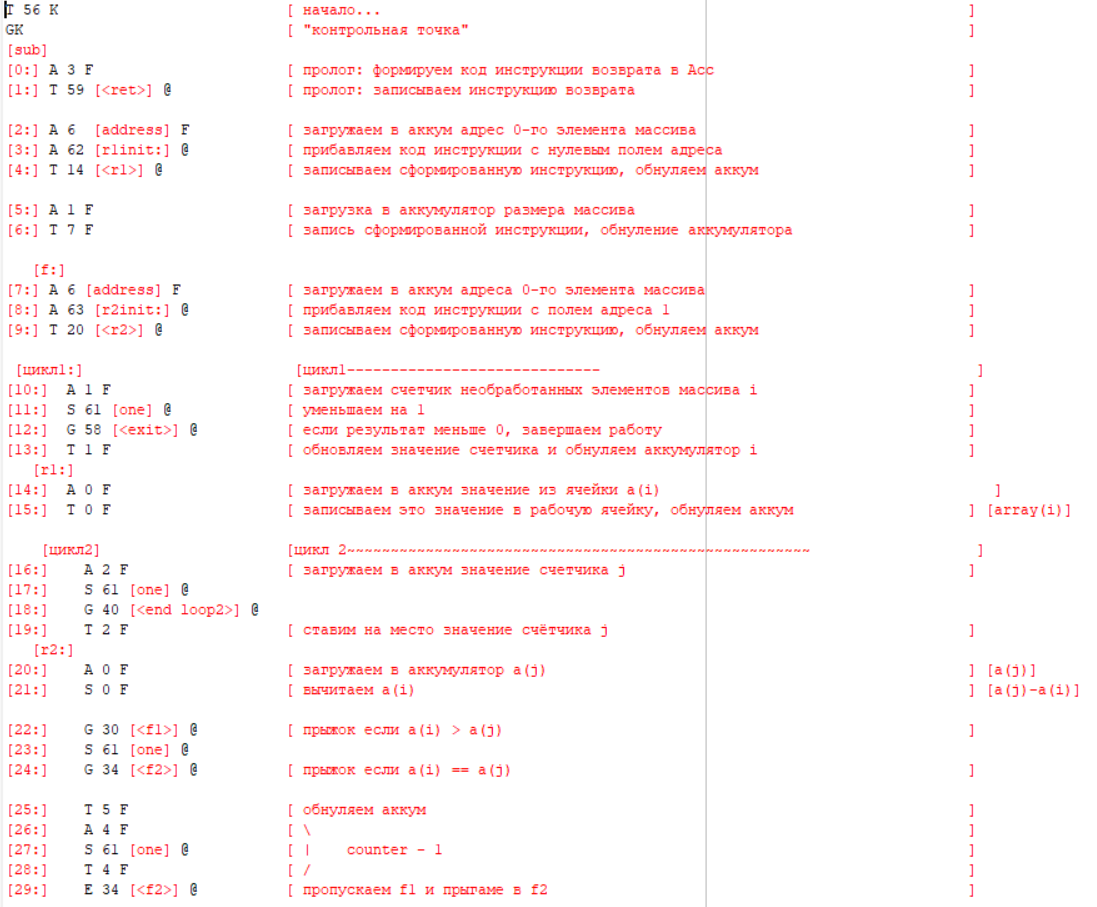
Таким образом, в IO1 используется абсолютная адресация, при которой идёт указание на конкретную ячейку, адрес которой не изменяется, что сильно усложняет написание кода. Так как при сдвиге или добавление одной строки, необходимо переписывать практически весь код.

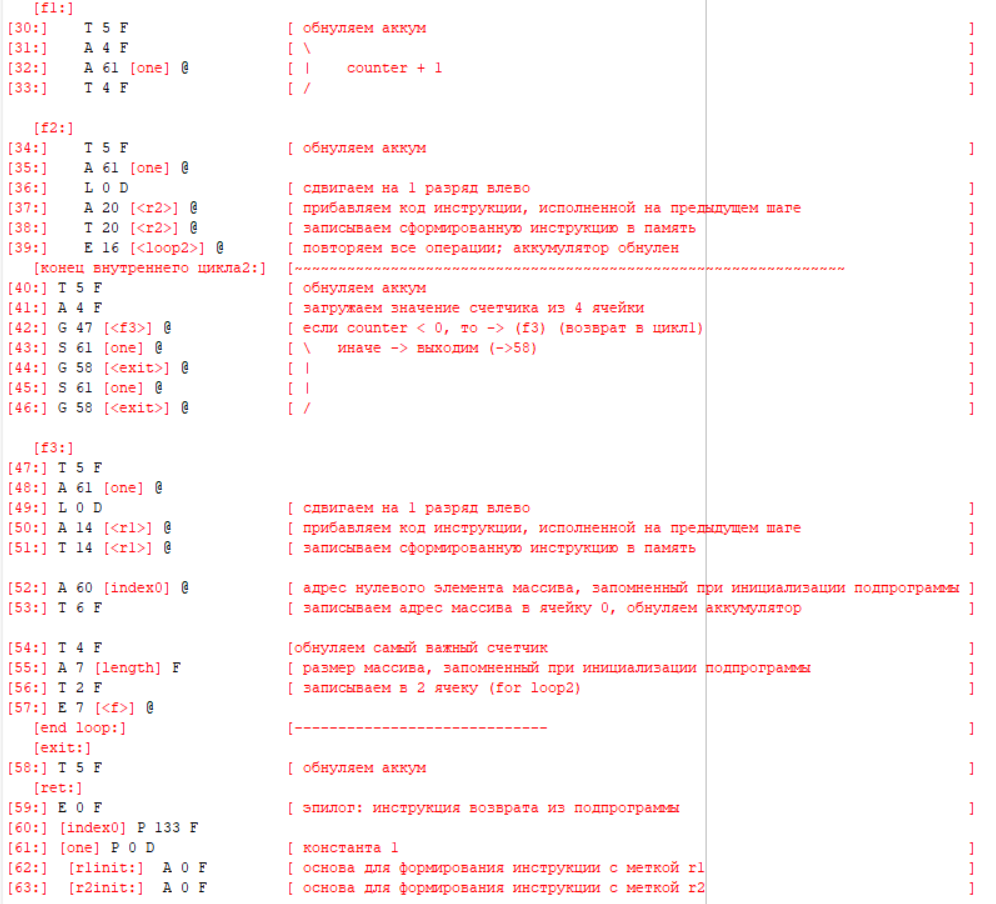
* 1. **Код программы Initial Orders 2.**

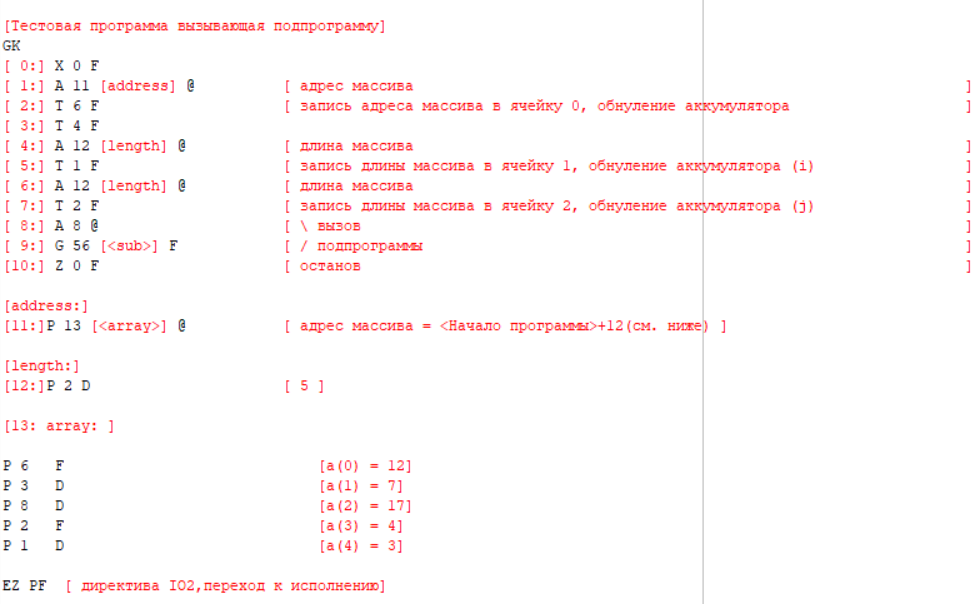
В отличие от Initial Orders 1, при использовании Initial Orders 2 на ленте находятся не только инструкции (или константы) загружаемой программы, но и управляющие последовательности (“control combinations”, современный термин – “директивы”), определяющее поведение самой программы Initial Order 2 в процессе загрузки. Директивы Initial Order 2 записываются в той же форме, что и инструкции загружаемой программы.

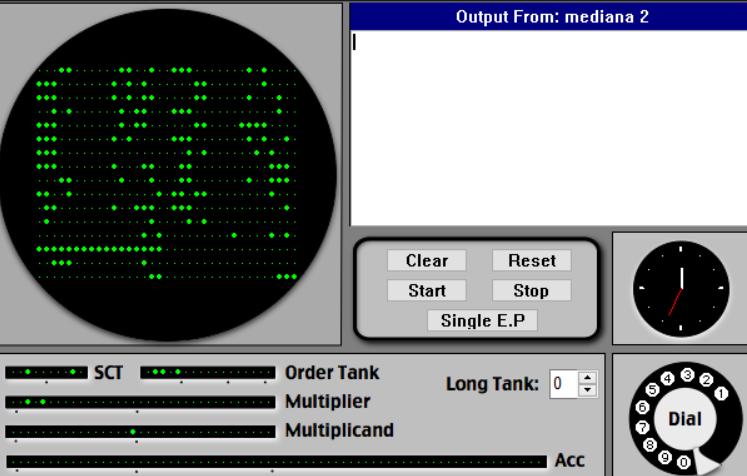
Переход к использованию Initial Orders 2 начинается с замены признака операций с короткими словами “S” на код (code letter) “F”, “L” на “D”.

«Каркас» замкнутой подпрограммы состоит из типовых директив, адресованных загрузчику Initial Orders 2, а также «пролога» (prologue) и «эпилога» (epilogue) – инструкций, исполняемых соответственно сразу после вызова и непосредственно перед возвратом из подпрограммы. Кроме этого, загрузчик Initial Orders 2 позволяет перейти от подсчета адресов инструкций во всей программе к подсчету инструкций только в рамках подпрограммы (@)





****

****



**3. Адреса и правила кодирования данных**

**3.1. Вариант программы (Initial Orders 1)**

Параметрами служат:

* Адреса элементов массива [100], [101] и т. д.;
* Длина массива [98];
* Вспомогательные ячейки: [1] и [2] содержат счетчики и [3] для контроля медианы
* Результат [0].

**3.2. Вариант с подпрограммой (Initial Orders 2)**

Параметрами служат:

1. Подпрограмма определения медианы:

* Индексы для прохода по массиву: [1], [2];
* Адрес в [6];
* Результат в [0].

1. Основная программа (адреса указываются относительно основной программы):

* Адреса элементов массива [13,18];
* Длина элементов массива [12];
* Результат работы подпрограммы [0]

**4.Вывод**

В ходе лабораторной работы были получены навыки программирования EDSAC . Основные проблемы, возникающие при работе с Initial Orders 1 это абсолютная адресация. Чтобы решить ее, мы использовали Initial Orders 2, где были реализованы подпрограммы и относительная адресация.