

Санкт-Петербургский политехнический университет Петра Великого  
Институт компьютерных наук и технологий  
Высшая школа интеллектуальных систем и суперкомпьютерных технологий

## **Отчёт по лабораторной работе № 2**

Дисциплина: Низкоуровневое программирование

Тема: Программирование EDSAC

Вариант 7

Выполнил студент гр. 3530901/90002 \_\_\_\_\_ Д.С. Ковалевский  
(подпись)

Принял старший преподаватель \_\_\_\_\_ Д.С. Степанов  
(подпись)

“ \_\_\_\_ ” \_\_\_\_\_ 2021 г.

Санкт-Петербург

2021

### **Цель работы:**

1. Разработать программу для EDSAC, реализующую определенную вариант задания функциональность, 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 определенную вариант задания функциональность 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.

## Вариант 7: Определение k-й порядковой статистики in-place.

### 1. Постановка задачи

Необходимо смоделировать программу для EDSAC, которая определит такой элемент неупорядоченного массива, если бы он был k-ым в упорядоченном.

Для реализации будем использовать сортировку и вывод k-ого элемента уже отсортированного массива.

### 2. Initial Orders 1.

Программа для EDSAC, реализующий поиск k-й порядковой статистики, использующий загрузчик Initial Orders 1.

```
1 T 116[end] S
2 [w4]T 113[booling] S [Сброс bool в 0]
3 T 114[iter] S [ Установка в 0 счетчика итераций]
4 [w1]A 106[N1] S [i - ое число массива]
5 S 107[N2] S [i + 1 - число массива]
6 G 50[to w2] S [if i < i+1]
7 S 106[N1] S [i2 - проверка, ноль ли во втором числе ]
8 S 110[i] S [null]
9 E 76[to next loop w3] S [если второе число ноль - переход на следующий круг ]
10 T 1 S [clear acc]
11 A 111[i] S [\]
12 T 113[booling] S [/ установка bool в 1]
13 T 1 S [clear acc]
14 A 106[N1] S [copy i]
15 T 112[temp] S [paste to temp]
16 A 107[N2] S [copy i+1]
17 T 106[N1] S [paste to N1]
18 A 112[temp] S [copy temp]
19 T 107[N2] S [Paste to N2]
20 [w2]T 1 S [clear acc]
21 A 34[i] S [\]
22 A 111[i] S [i (i) += 1]
23 T 34[i] S [/]
24 A 35[i + 1] S [\]
25 A 111[i] S [i (i+1) +=1]
26 T 35[i + 1] S [/]
27 A 44[copy i] S [\]
28 A 111[i] S [i (copy i) +=1]
29 T 44[copy i] S [/]
30 A 46[copy i + 1] S [\]
31 A 111[i] S [i (copy i + 1) +=1]
32 T 46[copy i + 1] S [/]
33 A 47[paste to N1] S [\]
34 A 111[i] S [i (paste to N1) +=1]
35 T 47[paste to N1] S [/]
36 A 49[paste to N2] S [\]
37 A 111[i] S [i (paste to N2) +=1]
38 T 49[paste to N2] S [/]
39 A 37[i2] S [\]
40 A 111[i] S [i (i2) +=1]
41 T 37[i2] S [/]
42 A 114[iteration] S [\]
43 A 111[i] S [i iter += 1]
44 T 114[iteration] S [/]
45 E 34[to w1] S [переход к следующей итерации]
46 [w3]A 34[i] S [\]
47 S 114[iter] S [i]
48 T 34[i] S [i]
49 A 35[i + 1] S [i]
50 S 114[iter] S [i]
51 T 35[i + 1] S [i]
52 A 37[i2] S [i]
53 S 114[iter] S [i]
54 T 37[i2] S [i]
55 A 44[copy i] S [i сброс всех индексов для следующего круга]
56 S 114[iter] S [i]
57 T 44[copy i] S [i]
58 A 46[copy i+1] S [i]
59 S 114[iter] S [i]
60 T 46[copy i+1] S [i]
61 A 47[paste to N1] S [i]
62 S 114[iter] S [i]
63 T 47[paste to N1] S [i]
64 A 49[paste to N2] S [i]
65 S 114[iter] S [i]
66 T 49[paste to N2] S [/]
67 A 113[bool] S [\]
68 S 111[i] S [i если перестановок не было, то переходим к выводу]
69 E 32[w4] S [/]
70 T 1 S [clear acc]
71 A 115[k] S [\]
72 A 104[exitpath] S [i считаем индекс k-ого число]
73 T 104[exitpath] S [/]
74 A 106[N1] S [exitpath]
75 T 115[k] S [выводим в 115 ячейку результат]
76 [N1]P 100 S [\]
77 [N2]P 10 S [i]
78 [N3]P 35 S [i массив данных]
79 [N4]P 22 S [/]
80 [null]P 0 S [i]
81 [i]P 1 S [\]
82 [temp]P 0 S [i константы]
83 [booling]P 0 S [i показывает были ли изменения в итерации]
84 [iteration]P 0 S [/]
85 [k]P 2 S [ число k, и результат после работы программы]
```

Рис.1. Программа на IO 1.

Входной массив данных храниться в ячейках 106 – 109(строки 76-79), а также в ячейке 110(строка 80) хранится 0, означающий конец массива, таким образом в массив нельзя добавлять нули, массив должен состоять из положительных десятичных ненулевых чисел.

Число k вводится в ячейку 115(85 строка), в конце работы в ней же будет выведен результат, таким образом для повторного запуска программы необходим сброс EDSAC.

Для увеличения массива, необходимо сместить все строки программы, начиная с 80 вниз на необходимое число, а также поменять значения адресов констант в ячейках, где они используются (номер строки + 30).

Программа использует сортировку пузырьком, со строки 2 по 69. В строках 14 -19 ячейки меняются местами, если необходимо, все остальные строки нужны для смены в них индексов или сброса к первоначальному.

67-75 выводит результат, прибавляя к индексу первого числа массива, число k.

### 3.Initial Orders 2.

Та же программа, используется как замкнутая подпрограмма с тестовой программой, вызывающей её.

```

1  T 43[start] K
2  G K [фиксируем адрес]
3  [0]A 3 F [инструкция возврата]
4  [1]T 76 @ [запись инструкции возврата]
5  [w4]T 84[booling] @ [Сброс bool в 0]
6  [3]T 85[iter] @ [ Установка в 0 счетчика итераций]
7  [w1]A 77[N1] @ [1 - ое число массива]
8  S 78[N2] @ [i + 1 - число массива]
9  G 20[to w2] @ [if i < i+1]
10 S 77[N1] @ [i2 - проверка, ноль ли во втором числе ]
11 S 81[i1] @ [null]
12 E 46[to next loop w3] @ [если второе число ноль - переход на следующий круг ]
13 T 1 F [cl acc]
14 A 82[i1] @ [\]
15 T 84[booling] @ [/ установка bool в 1]
16 T 1 F [clear acc]
17 A 77[N1] @ [copy i]
18 T 83[temp] @ [paste to temp]
19 A 78[N2] @ [copy i +1]
20 T 77[N1] @ [paste to N1]
21 A 83[temp] @ [copy temp]
22 T 78[N2] @ [Paste to N2]
23 [w2]T 1 F [clear acc]
24 A 4[i] @ [\]
25 A 82[i1] @ [i i += 1]
26 T 4[i] @ [/]
27 A 5[i + 1] @ [\]
28 A 82[i1] @ [i i+1 +=1]
29 T 5[i + 1] @ [/]
30 A 14[copy i] @ [\]
31 A 82[i1] @ [i copy i +=1]
32 T 14[copy i] @ [/]
33 A 16[copy i + 1] @ [\]
34 A 82[i1] @ [i copy i + 1 +=1]
35 T 16[copy i + 1] @ [/]
36 A 17[paste to N1] @ [\]
37 A 82[i1] @ [i paste to N1+=1]
38 T 17[paste to N1] @ [/]
39 A 19[paste to N2] @ [\]
40 A 82[i1] @ [i paste to N2+=1]
41 T 19[paste to N2] @ [/]
42 A 7[i2] @ [\]
43 A 82[i1] @ [i i2 +=1]
44 T 7[i2] @ [/]
45 A 85[iteration] @ [\]
46 A 82[i1] @ [i iter += 1]
47 T 85[iter] @ [/]
48 E 4[to w1] @ [переход к следующей итерации]
49 [w3]A 4[i] @ [\]
50 S 85[iter] @ [i]
51 T 4[i] @ [i]
52 A 5[i + 1] @ [i]
53 S 85[iter] @ [i]
54 T 5[i + 1] @ [i]
55 A 7[i2] @ [i]
56 S 85[iter] @ [i]
57 T 7[i2] @ [i сброс всех индексов для следующего круга]
58 A 14[copy i] @ [i]
59 S 85[iter] @ [i]
60 T 14[copy i] @ [i]
61 A 16[copy i+1] @ [i]
62 S 85[iter] @ [i]
63 T 16[copy i+1] @ [i]
64 A 17[paste to N1] @ [i]
65 S 85[iter] @ [i]
66 T 17[paste to N1] @ [i]
67 A 19[paste to N2] @ [i]
68 S 85[iter] @ [i]
69 T 19[paste to N2] @ [i]
70 A 84[bool] @ [\]
71 S 82[i1] @ [i если перестановок не было, то переходим к выводу]
72 E 2[w4] @ [/]
73 [w5]T 1 F [clear acc]
74 A 86[k] @ [\]
75 A 74[exitpath] @ [i считаем индекс k-ого число]
76 T 74[exitpath] @ [/]
77 A 77[N1] @ [exitpath]
78 T 134[k] F [выводим в 134 ячейку результат]
79 E 0 F [возврат из подпрограммы]
80 [N1]P 100 F [\]
81 [N2]P 43 F [i]
82 [N3]P 35 F [i массив данных]
83 [N4]P 50 F [/]
84 [null]P 0 F
85 [1]P 1 F [\]
86 [temp]P 0 F [i константы]
87 [booling]P 0 F [i показывает были ли изменения в итерации]
88 [iteration]P 0 F [/]
89 [k]P 2 F [ число k]
90 G K [фиксируем адрес]
91 A 0 @ [\вызов подпрограммы]
92 G 43 F [/]
93 Z 0 F [останов]
94 EZ PF [завершение]

```

Рис.2. Программа на Ю 2.

Входной массив данных храниться в ячейках 120 – 123(строки 80-83), а также в ячейке 128(строка 88) хранится 0, означающий конец массива, таким

образов в массив нельзя добавлять нули, массив должен состоять из положительных десятичных ненулевых чисел.

Число  $k$  вводится в ячейку 119(89 строка), в конце работы в ней же будет выведен результат, таким образом для повторного запуска программы необходим сброс EDSAC.

Для увеличения массива, необходимо сместить все строки программы, начиная с 84 вниз на необходимое число, а также поменять значения адресов констант в ячейках, где они используются (с учетом использования теты, число строки – 3 для модификации программы).

Результат выводится в 134 ячейку для удобства нахождения.

#### **4.Вывод**

Составленные программы для EDSAC с использованием IO 1 и IO 2 удовлетворяют варианту и находят  $k$ -ую порядковую статистику in-place.