

**Отчёт по лабораторной работе №2**

Дисциплина: Низкоуровневое программирование

Тема: Программирование на Edsac

Выполнил студент гр. 3530901/10005 \_\_\_\_\_ Кашин В.А.  
(подпись)

Преподаватель \_\_\_\_\_ Коренев Д. А.  
(подпись)

“ \_\_\_\_ ” \_\_\_\_\_ 2022 г.

Санкт-Петербург  
2022

### Вариант задания – 3: Сортировка обменом массива чисел in-place.

#### Initial Orders 1

[Сортировка обменом массива чисел in-place. 110-113]

[31] T 114 S [последняя строка + 1]

[32] T 0 S [acc = 0]

[33] A 108 S [len]

[34] S 107 S [len --]

[35] T 108 S [re len]

[36] T 0 S [acc = 0]

[37] A 109 S [addr 1]

[38] U 4 S [w 4(для первой переменной)]

[39] T 5 S [w 5(для второй переменной)]

[40] A 106 S [0]

[41] T 6 S [I = 0]

[42] A 106 S [0]

[43] S 108 S [-len]

[44] E 103 S [>=0 прошли весь массив, выход из цикла]

[45] T 0 S [acc = 0]

[46] A 4 S [addr 1]

[47] L 0 L [>>]

[48] A 105 S [A 0 S]

[49] T 50 S [w 50]

[50] A 0 S [считываем первый элемент массива]

[51] T 1 S [w 1]

[52] T 0 S [acc = 0]

[53] A 6 S [I]

[54] A 107 S [I ++]

[55] T 6 S [re I]

[56] A 5 S [считываем адрес первого сравниваемого числа]

[57] A 107 S [+1]

[58] U 5 S [re адрес второго числа и оставляем в acc]

[59] L 0 L [>>]

[60] A 105 S [A 0 S]

[61] T 62 S [w 62]

[62] A 0 S [считываем второй элемент массива]

[63] T 2 S [w 2]

[64] A 6 S [I]

[65] S 108 S [I - len]

[66] E 79 S [>=0 сравнение с последним элементом массива, переходим к SWAP,  
или же продолжаем]

[67] T 0 S [acc = 0]

[68] A 1 S [считываем 1-ую переменную]

[69] S 2 S [вычитаем 2-ую переменную]  
[70] E 52 S [ $\geq 0$  то не последний элемент массива и он меньше предыдущего,  
менять не надо]

[71] T 0 S [acc = 0]  
[72] A 2 S [считываем 2 переменную]  
[73] T 1 S [записываем в ячейку для первой переменной]  
[74] A 5 S [считываем адрес второго элемента]  
[75] T 4 S [записываем в ячейку для адреса 1]  
[76] A 2 S [считываем вторую переменную]  
[77] S 2 S [очистили ячейку для 2-ой переменной]  
[78] E 52 S [возврат в цикл]

[79] T 0 S [acc = 0]  
[80] A 1 S [считываем первую переменную]  
[81] S 2 S [вычитаем 2-ую переменную]  
[82] G 96 S [ $< 0$  если 2-ая переменная  $> 1$ -ой, то переставлять не надо  
и продолжаем цикл]

[83] T 0 S [acc = 0]  
[84] A 4 S [считываем адрес 1-го числа]  
[85] L 0 L [ $>>$ ]  
[86] A 104 S [T 0 S]  
[87] T 89 S [w 89]  
[88] A 2 S [считываем второе число]  
[89] T 0 S [записываем на его место(первого числа)]  
[90] A 5 S [считываем адрес 2-го числа]  
[91] L 0 L [ $>>$ ]  
[92] A 104 S [T 0 S]  
[93] T 95 S [w 95]  
[94] A 1 S [считываем 1-ое число]  
[95] T 0 S [записываем на его место(второго числа)]

[96] T 0 S [acc = 0]  
[97] A 108 S [len]  
[98] S 107 S [len --]  
[99] T 108 S [re len]

[100] A 106 S [0]  
[101] S 108 S [- len]  
[102] G 36 S [ $< 0$  проверка на количество пройденных элементов]  
[103] Z 0 S [конец]

[104] T 0 S [записывает значения в ячейку по адресу]  
[105] A 0 S [читает значение ячейки]

[106] P 0 S [0]  
[107] P 0 L [1]  
[108] P 2 S [len = 4]  
[109] P 55 S [addr 1]

[110] P 22 L [45]

[111] P 9 S [18]  
[112] P 45 S [90]  
[113] P 37 S [74]

Размер массива — [108]

Адрес массива – задаётся на 109 адресе, указывается адрес нулевой ячейки массива:

Массив – начинается с 110 адреса, можно задавать разные числа, добавлять новые или убирать данные (но также изменять и размер массива в соответствии с правилами).

### Результаты программы:

Массив:

[110] P 22 L [45]  
[111] P 9 S [18]  
[112] P 45 S [90]  
[113] P 37 S [74]

Результат:

WORD 110 Order = P 9 S Integer 110S = 18 Fraction 110S = 0.000275

WORD 111 Order = P 22 L Integer 111S = 45 Fraction 110L = 0.00068664656

WORD 112 Order = P 37 S Integer 112S = 74 Fraction 112S = 0.001129

WORD 113 Order = P 45 S Integer 113S = 90 Fraction 112L = 0.00137329532

Полученный массив – {18, 45, 74, 90}. Адреса значений отсортированного массива совпадают с исходными. Результат успешен.

### Initial Orders 2

[Сортировка обменом массива чисел in-place. 110-113]

T 130 K

GK

[0]A 3 F

[1]T 72 @

[2]A 108 F [len]

[3]S 107 F [len--]

[4]T 108 F [re len]

[5]T 0 F [acc = 0]

[6]A 109 F [addr 1]

[7]U 4 F [записываем в 4 ячейку для первой переменной]

[8]T 5 F [то что выше, но для 2 переменной]

[9]A 106 F [0]

[10]T 6 F [I = 0]

[11]A 106 F [0]

[12]S 108 F [-len]

[13]E 72 @ [>=0 поросмотрели весь масив=(конец)]

[14]T 0 F [acc = 0]

[15]A 4 F [addr 1]

[16]L 0 D [>>]

[17]A 105 F [A 0 S]

[18]T 19 @ [w в 19 ячейку]  
 [19]A 0 F [смотрим addr1]  
 [20]T 1 F [записываем в первую ячейку]  
  
 [21]T 0 F [acc = 0]  
 [22]A 6 F [I]  
 [23]A 107 F [I++]  
 [24]T 6 F [re I]  
  
 [25]A 5 F [считываем адрес первого сравниваемого числа(addr2)]  
 [26]A 107 F [I++]  
 [27]U 5 F [re addr2 + перенос числа из аккумулятора без обнуления аккумулятора ]  
 [28]L 0 D [>>]  
 [29]A 105 F [A 0 S]  
 [30]T 31 @ [w 31]  
 [31]A 0 F [считываем второй элемент массива]  
 [32]T 2 F [записываем во вторую ячейку]  
  
 [33]A 6 F [I]  
 [34]S 108 F [I - len]  
 [35]E 52 @ [>=0 сравнение с последним элементом массива, переходим к SWAP,  
 или же продолжаем]  
  
 [36]T 0 F [acc = 0]  
 [37]A 1 F [считываем 1-ую переменную]  
 [38]S 2 F [вычитаем 2-ую переменную]  
 [39]E 21 @ [>=0 это не последний элемент массива и он меньше предыдущего,  
 менять не надо]  
  
 [40]T 0 F [acc = 0]  
 [41]A 2 F [считываем 2 переменную]  
 [42]T 1 F [записываем в ячейку для первой переменной]  
 [43]A 5 F [записываем в ячейку для адреса 1]  
 [44]T 4 F [записываем в ячейку для адреса 2]  
 [45]A 2 F [еще раз считываем 2-ую переменную]  
 [46]S 2 F [очистили ячейку для 2-ой переменной]  
 [47]E 21 @ [возврат в цикл]  
  
 [48]T 0 F [acc = 0]  
 [49]A 1 F [считываем 1-ую переменную]  
 [50]S 2 F [вычитаем 2-ую переменную]  
 [51]G 65 @ [<0 если 2-ая переменная > 1-ой, то переставлять не надо  
 и продолжаем цикл]  
  
 [52]T 0 F [acc = 0]  
 [53]A 4 F [считываем адрес 1-го числа]  
 [54]L 0 D [>>]  
 [55]A 104 F [T 0 S]  
 [56]T 58 @ [w 58]  
 [57]A 2 F [считываем второе число]  
 [58]T 0 F [записываем на его место(первого числа)]  
 [59]A 5 F [считываем адрес 2-го числа]

```

[60]L 0 D [>>]
[61]A 104 F [T 0 S]
[62]T 64 @ [w 64]
[63]A 1 F [считываем 1-ое число]
[64]T 0 F [записываем на его место(второго числа)]

[65]T 0 F [acc = 0]
[66]A 108 F [len]
[67]S 107 F [len--]
[68]T 108 F [re len]

[69]A 106 F [0]
[70]S 108 F [-len]
[71]G 5 @ [<0 проверка на количество пройденных элементов]
[72]Z 0 F [конец]

```

```

T 101 K
GK
[0:]T 0 F
[1:]A 2 @
[2:]G 130 F

```

```

[3:]T 0 F [записывает значения в ячейку по адресу]
[4:]A 0 F [читает значение ячейки]

```

```

[5:]P 0 F [0]
[6:]P 0 D [1]
[7:]P 3 F [len = 6]
[8:]P 55 F [!!addr 1!!]

```

```

[10:]P 6 D [13]
[11:]P 4 F [8]
[12:]P 3 D [7]
[13:]P 1 F [0]
[14:]P 0 F [2]
[15:]P 2 F [4]

```

```

[13:]EZ PF

```

Результаты программы:

**Массив:**

```

[9:] P 6 D [13]
[10:]P 4 F [8]
[11:]P 0 F [0]
[12:]P 1 F [2]

```

Результат:

WORD 110 Order = P 0 F Integer 110F = 0 Fraction 110F = 0.000000

WORD 111 Order = P 1 F Integer 111F = 2 Fraction 110D = 0.00003051758

WORD 112 Order = P 4 F Integer 112F = 8 Fraction 112F = 0.000122

WORD 113 Order = P 6 D Integer 113F = 13 Fraction 112D = 0.00019836472

Полученный массив – {0, 2, 8, 13}. Адреса значений отсортированного массива совпадают с исходными. Результат успешен.