

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
ИНСТИТУТ КОМПЬЮТЕРНЫХ НАУК И ТЕХНОЛОГИЙ  
ВЫСШАЯ ШКОЛА ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ И СУПЕРКОМПЬЮТЕРНЫХ  
ТЕХНОЛОГИЙ

**Отчет по лабораторной работе №2**

**Дисциплина:** Низкоуровневое программирование

**Тема:** Машина Тьюринга

Работу выполнил: Чевычелов А. А.

Группа: 3530901/10003

Преподаватель: Коренев Д. А.

Санкт-Петербург  
2022

#### Оглавление

- 1. ТЗ.....3 стр.
- 2. Метод решения.....3 стр.
- 3. Описание состояний.....3 стр.
- 4. Работа программы.....3-5 стр.

# 1. ТЗ

Вариант 9. Реверс массива чисел.

## 2. Метод решения

Работа программы осуществляется с помощью «буфера». Числа будем менять попарно, начиная с концов массива и доходя до середины. При этом сначала первое число кладётся в «буфер», затем на его место в массиве ставится второе число. Далее на место второго числа в массиве ставится число из буфера (первое число).

Чтобы достичь нужного количества итераций цикла будем прибавлять +1 к размеру вне цикла, а затем вычитать -2 каждую итерацию.

Для работы программы необходимо знать адрес последнего элемента массива до входа в цикл. Для этого необходимо из длины массива вычесть 1 и прибавить адрес первого (например, адрес первого 91, а длина массива 6:  $(6-1) + 91 = 96$ ). Введем длину массива в виде  $P <len> S(F)$ . Чтобы получить  $P <len-1> S(F)$  нужно из  $P <len> S(F)$  вычесть 2. Затем прибавим адрес первого элемента ( $P <1st> S(F)$ ) и получим адрес второго в виде  $P <2nd> S(F)$ .

## 3. Программа Initial orders 1

Исходные данные находятся в ячейках 91-96, туда же записывается ответ (Этот же массив в реверсе). Длина массива находится в ячейке 81. В ячейке 86 находится длина массива в необходимом для вычислений виде (т.е.  $P <len> S$ ). Она нужна для изначального поиска адреса противоположного (последнего) элемента массива.

[Вариант 9. Реверс массива чисел.]

[Ответы находятся в 91-96 ячейках]

[31] T 97 S

[32] X 0 S

[33] A 81 S [Запись длины массива в аккумуля.]

[34] A 84 S [Увеличение длины массива на 1]

[35] T 82 S [Запись длины в ячейку редактирования]

[36] A 86 S [Запись в аккумуля. адреса противоположного эл-та]

[37] S 83 S [Запись в аккумуля. адреса противоположного эл-та]

[38] A 85 S [Запись в аккумуля. адреса противоположного эл-та]

[39] T 90 S [Запись в ячейку противоположного(последнего) эл-та]

[40] [loop:] T 0 S [Обнуление аккумуля.]

[41] A 88 S [Запись в аккумуля. значения T 0 S]

[42] T 68 S [Возвращение значения]

[43] A 88 S [Запись в аккумуля. значения T 0 S]

[44] T 71 S [Возвращение значения]

[45] A 87 S [Запись в аккумуля значения A 0 S]

[46] T 62 S [Возвращение значения A 0 S]

[47] A 87 S [Запись в аккумуля. значения A 0 S]

[48] T 67 S [Возвращение значения A 0 S]

[49] A 82 S [Запись оставшейся длины массива в аккумуля.]

[50] S 83 S [Уменьшение счётчика массива на 2]

[51] G 79 S [Условие выхода из цикла]

[52] T 82 S [Обновление длины]

[53] A 85 S [Запись в аккумуля. адреса 1го операнда]

[54] A 88 S [Формирование функции с помощью ячейки 88]  
 [55] T 68 S [Вставка функции в ячейку]  
 [56] A 90 S [Запись в аккумулятор адреса 2го операнда]  
 [57] A 88 S [Формирование функции с помощью ячейки 88]  
 [58] T 71 S [Вставка функции в ячейку]  
 [59] A 85 S [Запись адреса первого операнда в аккумулятор.]  
 [60] A 87 S [Преобразование адреса в команду]  
 [61] T 62 S [Запись команды в след. ячейку]  
 [62] A 0 S [Выполнение записанной команды]  
 [63] T 89 S [Сохранение первого эл-та перестановки в буфер]  
 [64] A 90 S [Запись адреса второго операнда в аккумулятор.]  
 [65] A 87 S [Преобразование адреса в команду]  
 [66] T 67 S [Запись команды в след. строку]  
 [67] A 0 S [Выполнение команды]  
 [68] T 0 S [Выполнение ранее записанной программы (55), Замена первого операнда]  
 [69] X 0 S [Ничего]  
 [70] A 89 S [Запись значения из буфера в аккумулятор.]  
 [71] T 0 S [Замена второго операнда значением из буфера, ранее вставленная функция]  
 [72] A 85 S [Замена адресов]  
 [73] A 83 S [Увеличение адреса на 2(след. эл-т)]  
 [74] T 85 S [Запись измененного адреса]  
 [75] A 90 S [Замена адресов]  
 [76] S 83 S [Уменьшение адреса на 2(пред. эл-т)]  
 [77] T 90 S [Запись измененного адреса]  
 [78] E 40 S [Возвращение к началу цикла] [/loop]  
 [79] Z 0 S [Завершение функциональной части]  
 [80] X 0 S [Ничего]  
 [81] P 3 S [LENGTH]  
 [82] P 0 S [LEN.MOD, счетчик массива]  
 [83] P 1 S [Константа 2]  
 [84] P 0 L [Константа 1]  
 [85] P 91 S [adr of 1st]  
 [86] P 6 S [Размер массива в необходимом для вычислений виде]  
 [87] A 0 S [Константа для записи в аккумулятор.]  
 [88] T 0 S [Константа для функции вставки]  
 [89] P 0 S [буфер]  
 [90] P 0 S [N + Size: adr of 2nd]  
 [91] P 0 S [ 0 ] [Начало массива]  
 [92] P 0 L [ 1 ]  
 [93] P 1 S [ 2 ]  
 [94] P 1 L [ 3 ]  
 [95] P 2 S [ 4 ]  
 [96] P 2 L [ 5 ]

## 4. Работа программы Initial orders 1

Входные данные:

[91] P 0 S [ 0 ]  
 [92] P 0 L [ 1 ]  
 [93] P 1 S [ 2 ]  
 [94] P 1 L [ 3 ]

[95] P 2 S [ 4 ]

[96] P 2 L [ 5 ]

Результат:

WORD 91 Order = P 2 L Integer 91S = 5 Fraction 90L = 0.00007630477

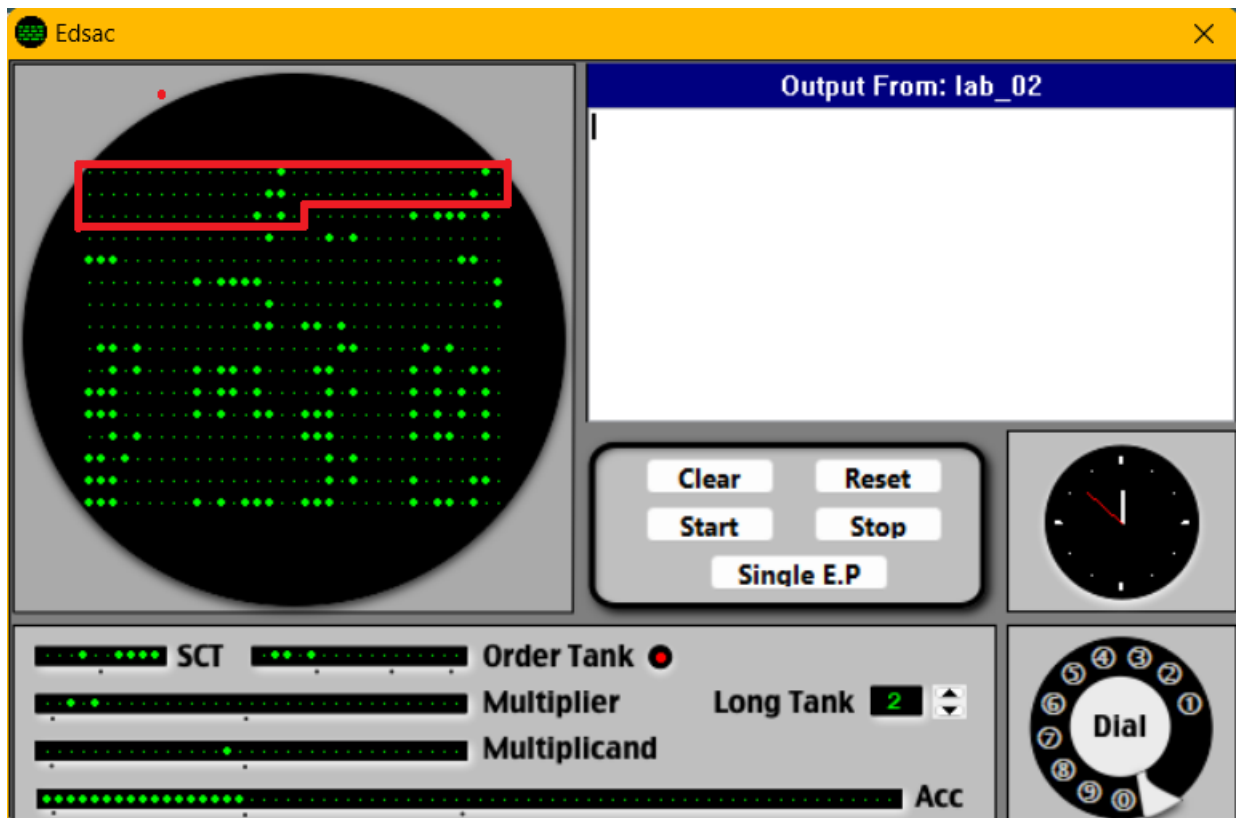
WORD 92 Order = P 2 S Integer 92S = 4 Fraction 92S = 0.000061

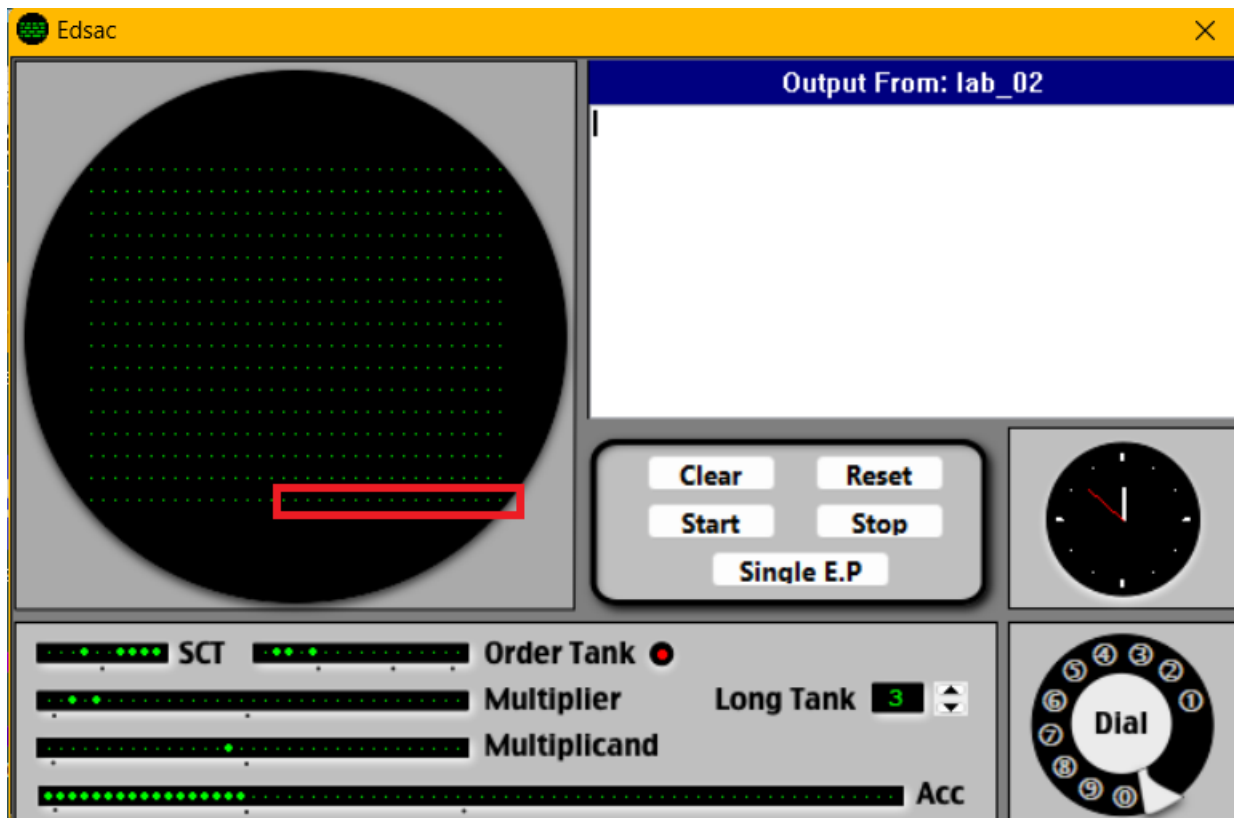
WORD 93 Order = P 1 L Integer 93S = 3 Fraction 92L = 0.00004577660

WORD 94 Order = P 1 S Integer 94S = 2 Fraction 94S = 0.000031

WORD 95 Order = P 0 L Integer 95S = 1 Fraction 94L = 0.00001525891

WORD 96 Order = P 0 S Integer 96S = 0 Fraction 96S = 0.000000





## 5. Программа Initial orders 2

Строчки 54-108 занимает подпрограмма.

Строчки 109-128 занимает программа.

10 строчка – адрес массива (первого операнда).

11 строчка – адрес противоположного элемента (второго операнда).

12 строчка – длина массива.

13 строчка – длина массива в необходимом для вычислений виде (т.е.  $P < len > F$ ).

Исходные данные находятся в ячейках 122-127.

Ответ записывается в ячейки 122-127.

(В редакторе кода ячейки 122-127 находятся на строчках 123-128, это произошло из-за сдвига, образованного в ячейке 109: G K; так как это директива IO2, она не занимает ячейку памяти)

[Вариант 9. Реверс массива чисел.]

[Ответ находится в ячейках 122-127]

[Исходный массив находится в ячейках 122-127, 13-18 ячейки программы]

[54] T 56 K [Директива IO2, установка адреса загрузки]

[55] G K [Директива IO2, фиксация начального адреса подпрограммы]

[56] [0:] A 3 F [Пролог: формирование кода инструкции возврата в аккумуляторе.]

[57] [1:] T 46 @ [Пролог: запись инструкции возврата]

[58] [2:] X 0 F [Ничего]

[59] [3:] A 12 F [Загрузка длины массива в аккумулятор.]

[60] [4:] A 51 @ [+1 к длине массива]

[61] [5:] T 52 @ [Запись в ячейку для редактирования]

[62] [6:] A 13 F [Загрузка длины массива в нужном виде в аккумулятор.]

[63] [7:] S 50 @ [Вычитание 2]  
[64] [8:] A 10 F [Прибавление адреса первого операнда]  
[65] [9:] T 11 F [Запись адреса второго операнда в ячейку]  
[66] [loop] [10:] T 0 F [Обнуление аккумулята]  
[67] [11:] A 49 @ [Запись в аккумулята значения T 0 F]  
[68] [12:] U 36 @ [Возвращение значения]  
[69] [13:] T 38 @ [Возвращение значения]  
[70] [14:] A 48 @ [Запись в аккумулята значения A 0 F]  
[71] [15:] U 30 @ [Возвращение значения]  
[72] [16:] T 35 @ [Возвращение значения]  
[73] [17:] A 52 @ [Загрузка счетчика массива в аккумулята]  
[74] [18:] S 50 @ [Вычитание 2]  
[75] [19:] G 45 @ [Условие выхода, счетчик пустой]  
[76] [20:] T 52 @ [Запись оставшегося счетчика]  
[77] [21:] A 10 F [Загрузка адреса первого операнда в аккумулята]  
[78] [22:] A 49 @ [Формирование функции в ячейке]  
[79] [23:] T 36 @ [Вставка готовой функции в ячейку 36]  
[80] [24:] A 11 F [Загрузка адреса второго операнда в аккумулята]  
[81] [25:] A 49 @ [Формирование функции в ячейке]  
[82] [26:] T 38 @ [Вставка готовой функции в ячейку 38]  
[83] [27:] A 10 F [Загрузка адреса первого операнда в аккумулята]  
[84] [28:] A 48 @ [Формирование функции в ячейке]  
[85] [29:] T 30 @ [Вставка готовой функции в след. ячейку]  
[86] [30:] A 0 F [Выполнение вставленной функции, загрузка значения первого операнда в аккумулята]  
[87] [31:] T 47 @ [Запись в buffer]  
[88] [32:] A 11 F [Загрузка адреса второго операнда в аккумулята]  
[89] [33:] A 48 @ [Формирование функции в ячейке]  
[90] [34:] T 35 @ [Вставка готовой функции в след. ячейку]  
[91] [35:] A 0 F [Выполнение вставленной функции, загрузка значения второго операнда в аккумулята]  
[92] [36:] T 0 F [Выполнение вставленной функции, запись значения второго на место первого]  
[93] [37:] A 47 @ [Загрузка значения buffer в аккумулята]  
[94] [38:] T 0 F [Выполнение вставленной функции, запись значения первого на место второго]  
[95] [39:] A 10 F [Обновление адресов, загрузка адреса первого операнда в аккумулята]  
[96] [40:] A 50 [+2] @ [Прибавление 2, след. эл-т]  
[97] [41:] T 10 F [Запись нового адреса]  
[98] [42:] A 11 F [Обновление адресов, загрузка адреса второго операнда в аккумулята]  
[99] [43:] S 50 [-2] @ [Вычитание 2, пред. эл-т]  
[100] [44:] T 11 F [Запись нового адреса]  
[101] [45:] E 10 @ [Конец цикла, возвращение к началу]  
[102] [46:] E 0 F [Эпилог: инструкция возврата из подпрограммы]  
[103] [47:] P 0 F [buffer]  
[104] [48:] A 0 F [A 0 F]  
[105] [49:] T 0 F [T 0 F]  
[106] [50:] P 1 F [Константа 2]  
[107] [51:] P 0 D [Константа 1]  
[108] [52:] P 0 F [Ячейка редактирования, счетчик массива]  
[109] G K [Директива IO2, фиксация начального адреса программы]  
[110] [0:] X 0 F [Отладка]

[111] [1:] A 11 @ [Загрузка длины массива в аккумулятор]  
 [112] [2:] T 12 F [Запись в ячейку]  
 [113] [3:] A 10 @ [Загрузка адреса массива, первый элемент]  
 [114] [4:] T 10 F [Запись в ячейку]  
 [115] [5:] A 12 @ [Загрузка длины в аккумулятор]  
 [116] [6:] T 13 F [Запись в ячейку]  
 [117] [7:] A 7 @ [\ вызов]  
 [118] [8:] G 56 F [/ подпрограммы]  
 [119] [9:] Z 0 F [Останов]  
 [120] [10:] P 13 @ [Адрес массива, первый элемент]  
 [121] [11:] P 3 F [Длина массива]  
 [122] [12:] P 6 F [Длина массива в удобном для работы программы виде: (P <len> F)]  
 [123] [13:] P 0 F [0]  
 [124] [14:] P 0 D [1]  
 [125] [15:] P 1 F [2]  
 [126] [16:] P 1 D [3]  
 [127] [17:] P 2 F [4]  
 [128] [18:] P 2 D [5]  
 [129] EZ PF [Директива IO2, переход к исполнению]

## 6. Работа программы Initial orders 2

Входные данные:

[123] P 0 F [0]  
 [124] P 0 D [1]  
 [125] P 1 F [2]  
 [126] P 1 D [3]  
 [127] P 2 F [4]  
 [128] P 2 D [5]

При этом занятые ячейки в памяти 122-127

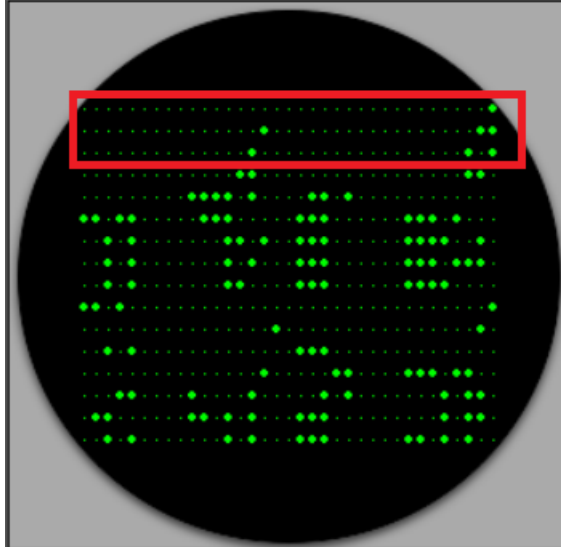
Результат:

```

WORD 122  Order = P 2 D  Integer 122F = 5  Fraction 122F = 0.000076
WORD 123  Order = P 2 F  Integer 123F = 4  Fraction 122D = 0.00006103545
WORD 124  Order = P 1 D  Integer 124F = 3  Fraction 124F = 0.000046
WORD 125  Order = P 1 F  Integer 125F = 2  Fraction 124D = 0.00003051775
WORD 126  Order = P 0 D  Integer 126F = 1  Fraction 126F = 0.000015
WORD 127  Order = P 0 F  Integer 127F = 0  Fraction 126D = 0.000000000006
  
```



Output From: lab\_02\_I02



Clear

Reset

Start

Stop

Single E.P



SCT Order Tank ●

Multiplier

Multiplicand

Long Tank 3

Acc

