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.Д. Коротких

(подпись)

Преподаватель А. О. Алексюк

(подпись)

“ ” 2021 г.

Санкт-Петербург

2021

Оглавление

[1. Техническое задание 3](#_Toc67442640)

[2. Метод решения 3](#_Toc67442641)

[3. Программа Initial Orders 1 3](#_Toc67442642)

[4. Работа программы Программа Initial Orders 1 8](#_Toc67442643)

[5. Программа Initial Orders 1 9](#_Toc67442644)

[6. Работа программы Программа Initial Orders 2 16](#_Toc67442645)

[7. Работа Адреса и правила кодирования исходных данных и результатов. Руководство программиста 17](#_Toc67442646)

# Техническое задание

Определение K-ой порядковой статистики in-place.

# Метод решения

Поиск k-ой порядковой статистики разбит на два этапа:

1) Сортировка исходного массива с использованием алгоритма «пузырька»

2) Выделение в отсортированном массиве элемента на позиции k

# Программа Initial Orders 1

[Техническое задание - Определение K-ой порядковой статистики in-place]

[В ячейках 1 и 2 храним длину оставшейся части массива для обработки (для каждого цикла)]

[Ячейка 0 отводится для перестановки двух соседних элементов (пузырек)]

[Результат хранится в ячейке 0]

[k задаётся в ячейке 114]

[k=3]

[массив находится в ячейках 118-123]

[31]T 124[<N+1>] S [Инструкции загружаются в ячейки 31..N]

[32]Z 0 S

[33]X 0 S

[34]X 0 S

[35]X 0 S

[36]X 0 S

[37]X 0 S

[38]X 0 S

[39]X 0 S

[Здесь предполагается, что аккумулятор обнулен]

[40]A 112[<len>] S [Загрузка в аккумулятор длины обрабатываемого массива]

[41]S 110[<c1>] S [Уменьшаем значение аккумулятора на 1 (сортировка перебором количества пар элементов. Пар меньше длины на 1)]

[42]T 1 S [Запись этого значения в ячейку 1, обнуление аккумулятора]

[loop; Пузырек. Проходим по вложенному циклу (N-1) раз]

[Счетчик цикла храним в ячейке 1]

[43] T 10 S [Очистка аккумулятора]

[44] A 1 S [Загружаем счетчик необработанных элементов массива]

[45] S 110[<c1>] S [Уменьшаем на 1]

[46] G 97[<end sort>] S [Если результат меньше 0, выходим из цикла]

[47] T 1 S [Обновляем значение счетчика и обнуляем аккумулятор]

[Устанавливаем счетчик вложенного цикла в (N-1)]

[48] A 112[<len>] S [Загрузка в аккумулятор длины обрабатываемого массива]

[49] S 110[<c1>] S [Уменьшаем значение аккумулятора на 1 (сортировка перебором количества пар элементов. Пар меньше длины на 1)]

[50] T 2 S [Запись этого значения в ячейку 2, обнуление аккумулятора]

[Установка адресов в инструкциях на 0 и 1 элементы массива (изменяем адреса во вложенном цикле)]

[Инициализируем инструкцию r1 и r2 по шаблонам]

[51] A 113[<addr>] S [Загрузка в аккумулятор адреса 0-го элемента массива]

[52] L 0 L [Сдвиг аккумулятора на 1 разряд влево]

[53] A 115[шаблон <r1>] S [Прибавляем код инструкции с нулевым полем адреса]

[54] U 71[<r1>] S [Запись сформированной инструкции без обнуления аккумулятора]

[55] A 111[<c2>] S [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес 1-го элемента массива)]

[56] T 76[<r2>] S [Запись сформированной инструкции, обнуление аккумулятора]

[Инициализируем инструкции w1 и w2]

[57] A 113[<addr>] S [Загрузка в аккумулятор адреса 0-го элемента массива]

[58] L 0 L [Сдвиг аккумулятора на 1 разряд влево]

[59] A 117[шаблон <w1>] S [Прибавляем код инструкции с нулевым полем адреса]

[60] U 77[<w1>] S [Запись сформированной инструкции без обнуления аккумулятора]

[61] A 111[<c2>] S [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес 1-го элемента массива)]

[62] T 79[<w2>] S [Запись сформированной инструкции, обнуление аккумулятора]

[Инициализируем инструкцию z2]

[63] A 113[<addr>] S [Загрузка в аккумулятор адреса 0-го элемента массива]

[64] L 0 L [Сдвиг аккумулятора на 1 разряд влево]

[65] A 116[шаблон <z2>] S [Прибавляем код инструкции с нулевым полем адреса]

[66] T 73[<z2>] S [Запись сформированной инструкции, обнуление аккумулятора]

[loop 2; Проход по массиву с перестановками (еще один вложенный цикл)]

[Счетчик цикла храним в ячейке 2]

[Здесь предполагается, что аккумулятор обнулен]

[67] A 2 S [Загружаем счетчик необработанных элементов массива]

[68] S 110[<c1>] S [Уменьшаем на 1]

[69] G 43[<loop>] S [Если результат меньше 0, завершаем цикл и уходим в основную ветку]

[70] T 2 S [Обновляем значение счетчика и обнуляем аккумулятор]

[Перестановка элементов при необходимости]

[Здесь предполагается, что аккумулятор обнулен]

[71] [r1]A 0[N] S [Загрузка в аккумулятор значения из ячейки N]

[72] U 0 S [Запись этого значения в рабочую ячейку без обнуления аккумулятора]

[Проверяем, надо ли переставлять числа]

[73] [z2]S 0[N+1] S [Вычитание из аккумулятора значения следующей ячейки (N+1). acc = A{N}-A{N+1}]

[74] G 80[<else>] S [Если A{N+1}>A{N} (acc<0), то переставлять элементы местами не нужно. Пропускаем этот шаг]

[75] T 10 S [Очистка аккумулятора]

[76] [r2]A 0[N+1] S [Загрузка в аккумулятор значения из ячейки N+1]

[77] [w1]T 0[N] S [Запись этого значения в ячейку с адресом N, обнуление аккумулятора]

[78] A 0 S [Загрузка в аккумулятор значения из ячейки 0]

[79] [w2]T 0[N+1] S [Запись этого значения в ячейку с адресом N+1, обнуление аккумулятора]

[Аккумулятор обнулен]

[else:]

[80] T 10 S [Очистка аккумулятора]

[Теперь обновляем адреса в инструкциях r1, r2, w1, w2, z2]

[обновляем r1]

[81] A 71[<r1>] S [Загрузка в аккумулятор инструкции r1]

[82] A 111[<c2>] S [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+1-го элемента массива)]

[83] T 71[<r1>] S [Запись сформированной инструкции, обнуление аккумулятора]

[обновляем r2]

[84] A 76[<r2>] S [Загрузка в аккумулятор инструкции r2]

[85] A 111[<c2>] S [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+2-го элемента массива)]

[86] T 76[<r2>] S [Запись сформированной инструкции, обнуление аккумулятора]

[обновляем w1]

[87] A 77[<w1>] S [Загрузка в аккумулятор инструкции w1]

[88] A 111[<c2>] S [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+1-го элемента массива)]

[89] T 77[<w1>] S [Запись сформированной инструкции, обнуление аккумулятора]

[обновляем w2]

[90] A 79[<w2>] S [Загрузка в аккумулятор инструкции w2]

[91] A 111[<c2>] S [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+2-го элемента массива)]

[92] T 79[<w2>] S [Запись сформированной инструкции, обнуление аккумулятора]

[обновляем z2]

[93] A 73[<z2>] S [Загрузка в аккумулятор инструкции z2]

[94] A 111[<c2>] S [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+2-го элемента массива)]

[95] T 73[<z2>] S [Запись сформированной инструкции, обнуление аккумулятора]

[96] E 67[<loop 2>] S [Повторяем все операции; аккумулятор обнулен]

[Здесь конец цикла loop]

[Конец сортировки пузырьком]

[end sort:]

[97]X 0 S [Останов для отладки]

[Поиск k-ой порядковой статистики]

[98] T 10 S [Очистка аккумулятора]

[99] A 114[<k>] S [Загрузка значения k в аккумулятор]

[100] S 110[<c1>] S [Вычитаем константу 1]

[101] A 113[<addr>] S [Получаем адрес k-ой порядковой статистики]

[102]X 0 S [Останов для отладки]

[103] L 0 L [Сдвиг влево на 1 разряд]

[104] A 115[<шаблон r1>] S [Формируем инструкцию чтения k-ой порядковой статистики из ее адреса]

[105] T 106[N+1] S [Записываем инструкцию в следующую ячейку, обнуляем аккумулятор]

[106] A 0 S [Инструкция для перезаписи; Читаем статистику из ячейки по найденному адресу]

[107] T 0 S [Записываем k-ую порядковую статистику в 0 ячейку, обновляем аккумулятор]

[k-ая порядковая статистика хранится в ячейке 0]

[108]X 0 S [Пустая команда]

[109]Z 0 S [Останов]

[с1:]

[110]P 0 L [Константа 1 (0…0|0|0…0|1 = 1)]

[с2:]

[111]P 1 S [Константа 2 (0…0|0|0…1|0 = 2)]

[len:]

[112]P 3 S [ 6 ] [Длина массива]

[addr:]

[113]P 59 S [ 118 ] [Адрес 0-го элемента массива]

[k] [Номер искомой поряковой статистики]

[114]P 1 L [ 3 ]

[Шаблоны инструкций для загрузки в loop]

[шаблон r1:]

[115] A 0[N] S

[шаблон z2:]

[116] S 1[N+1] S

[шаблон w1:]

[117] T 0[N] S

[array:]

[118]P 2 L [ 5 ]

[119]P 4 S [ 8 ]

[120]P 3 S [ 6 ]

[121]P 0 L [ 1 ]

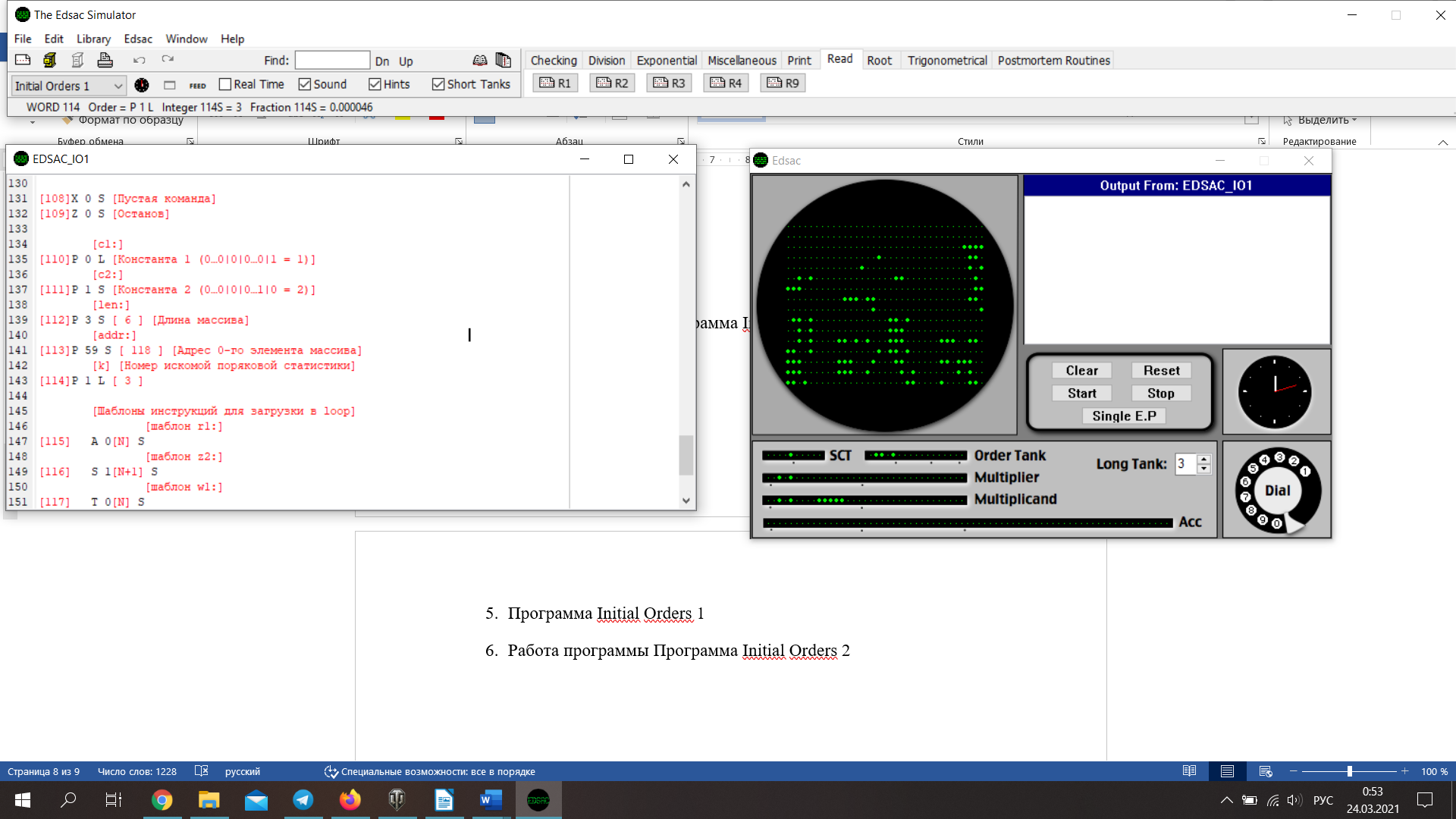
[122]P 7 L [ 15 ]

[123]P 0 S [ 0 ]

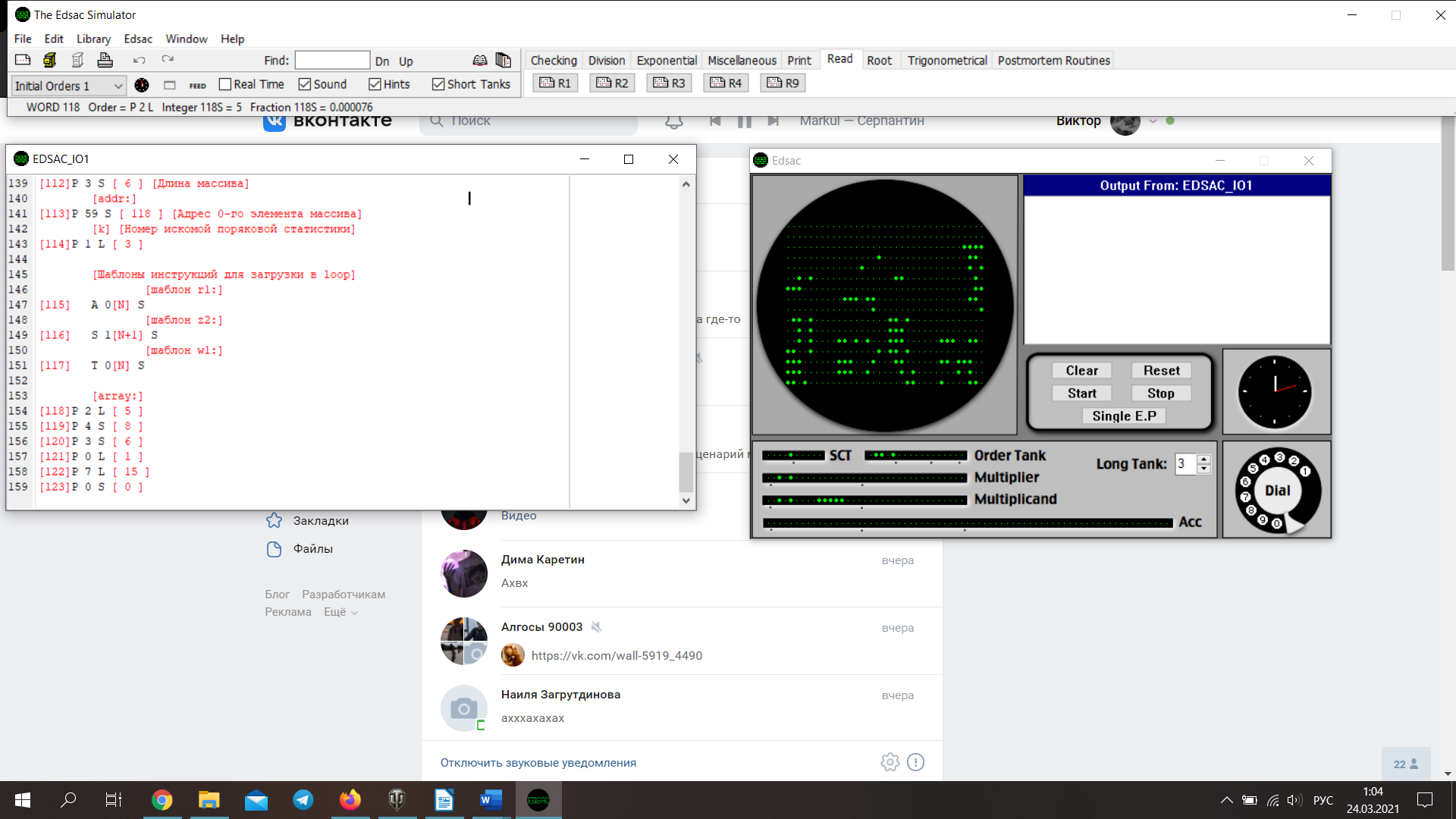
# Работа программы Программа Initial Orders 1

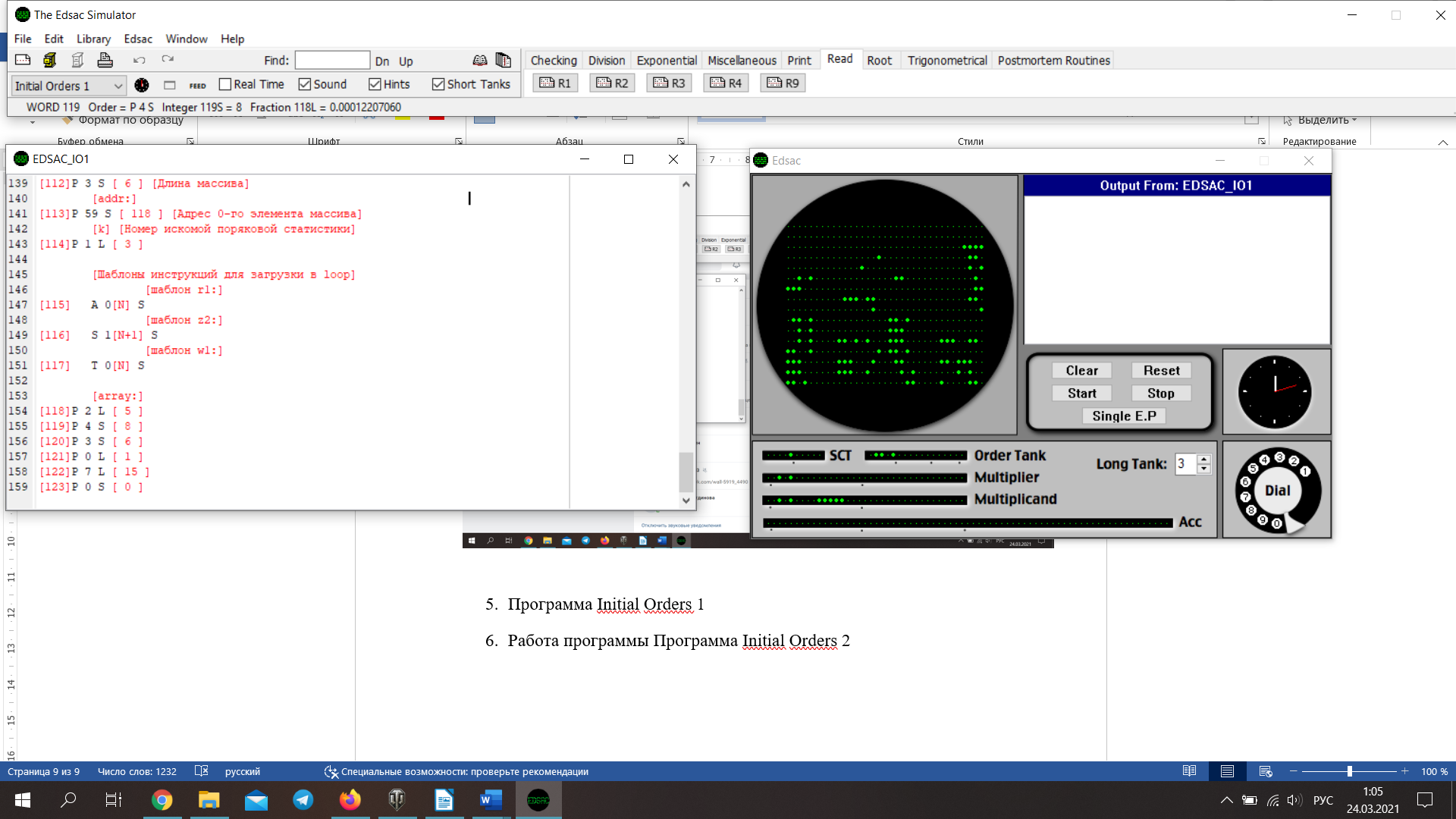
Исходные данные

к:

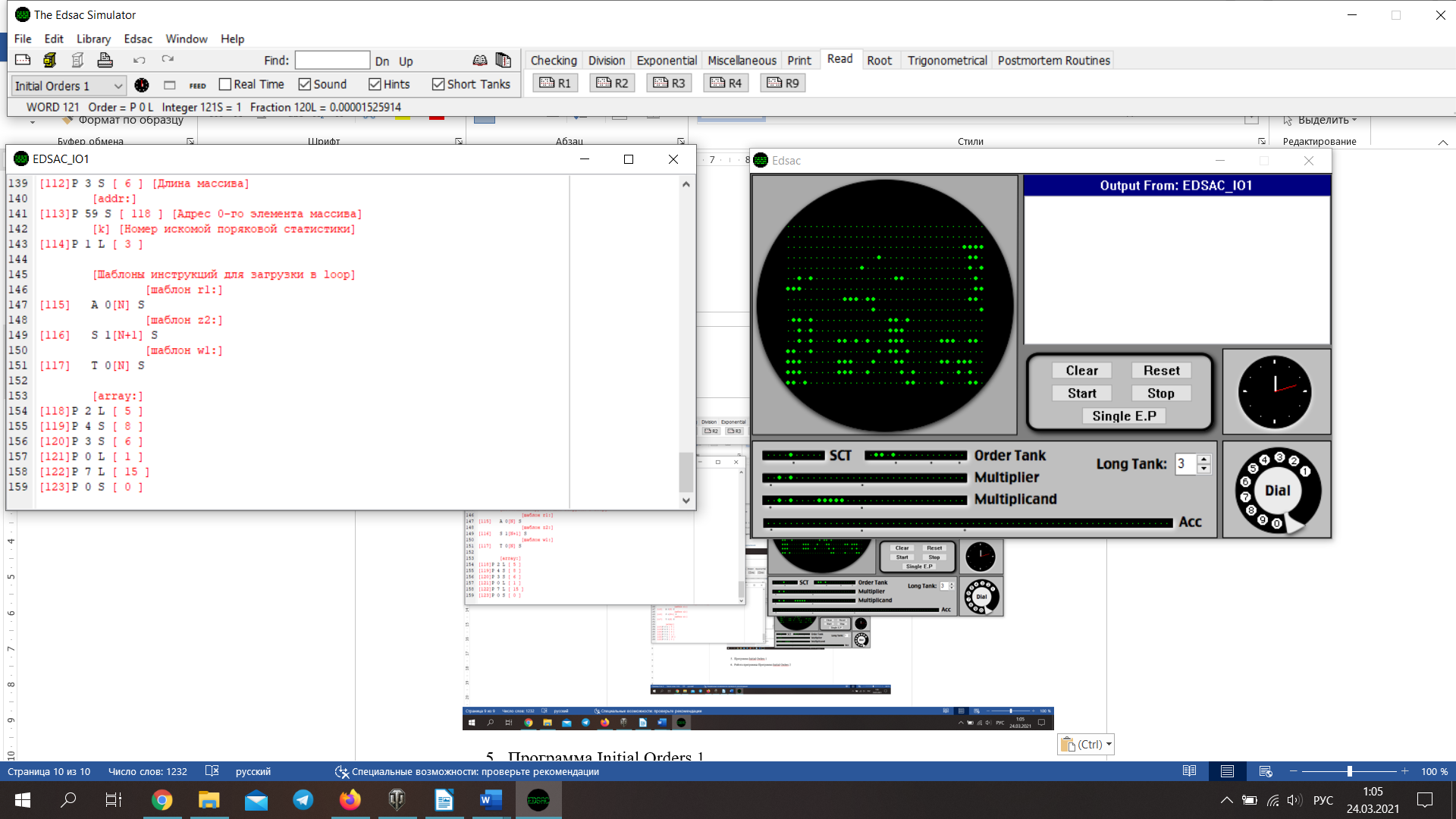


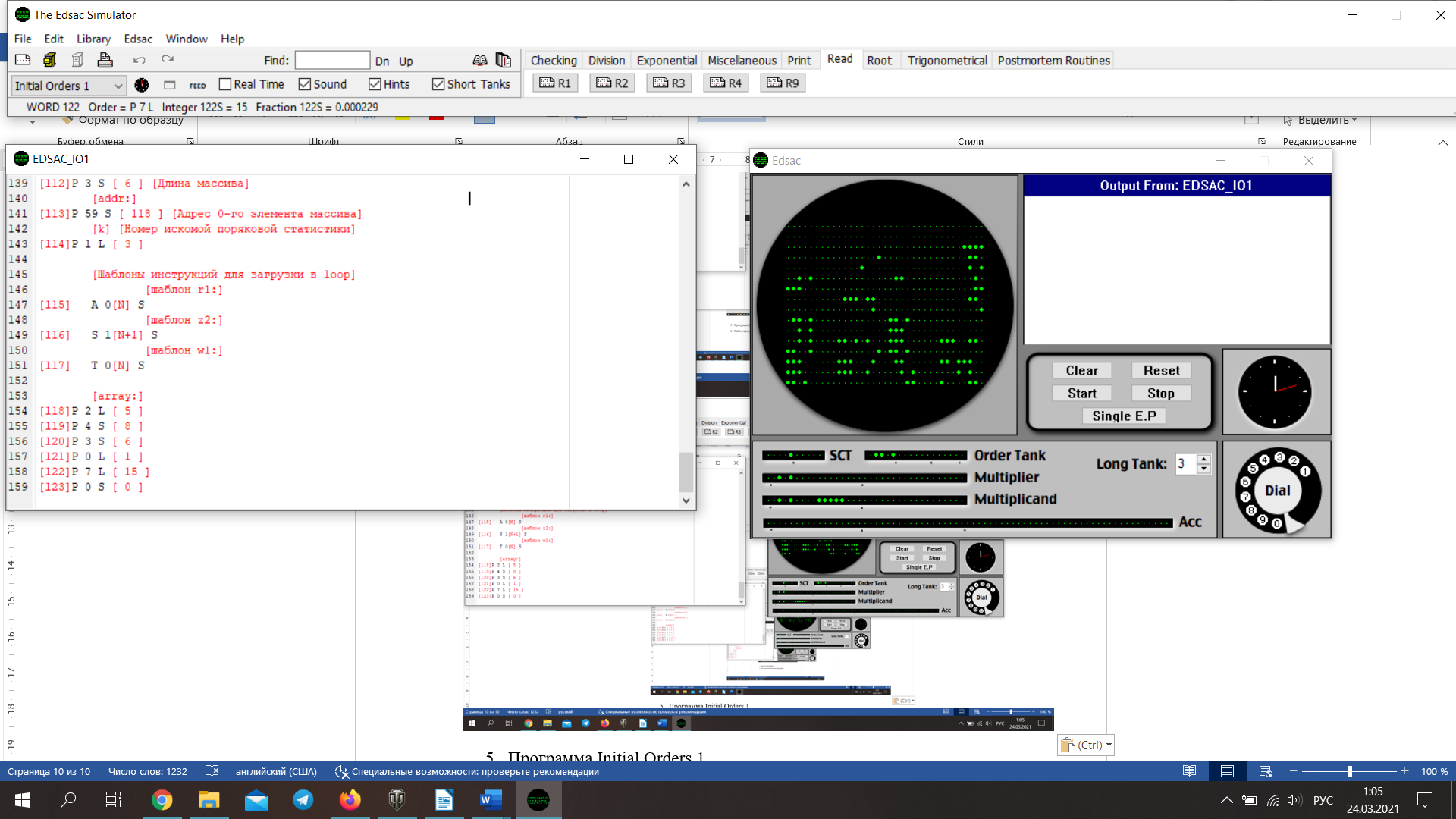
массив:

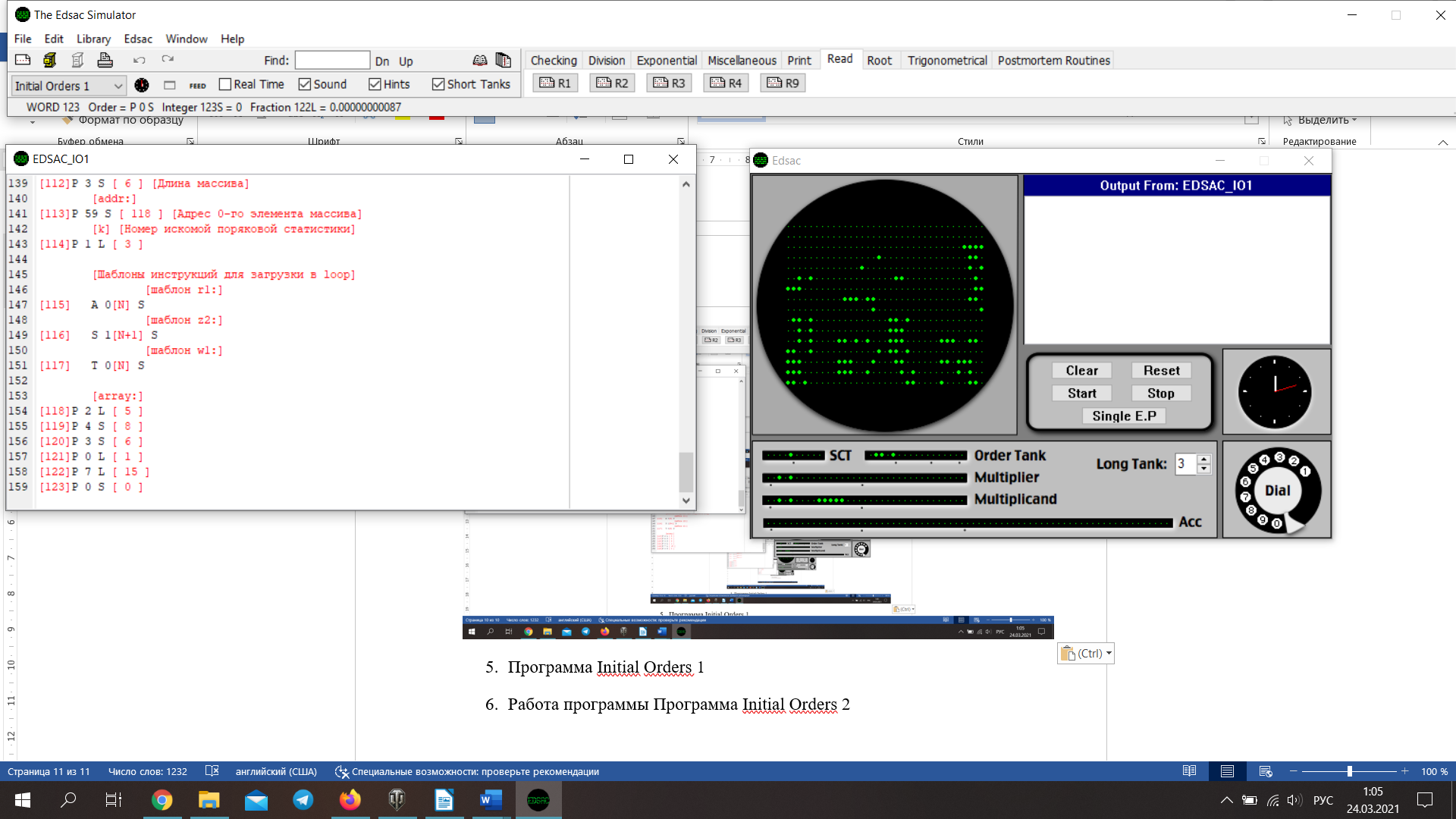




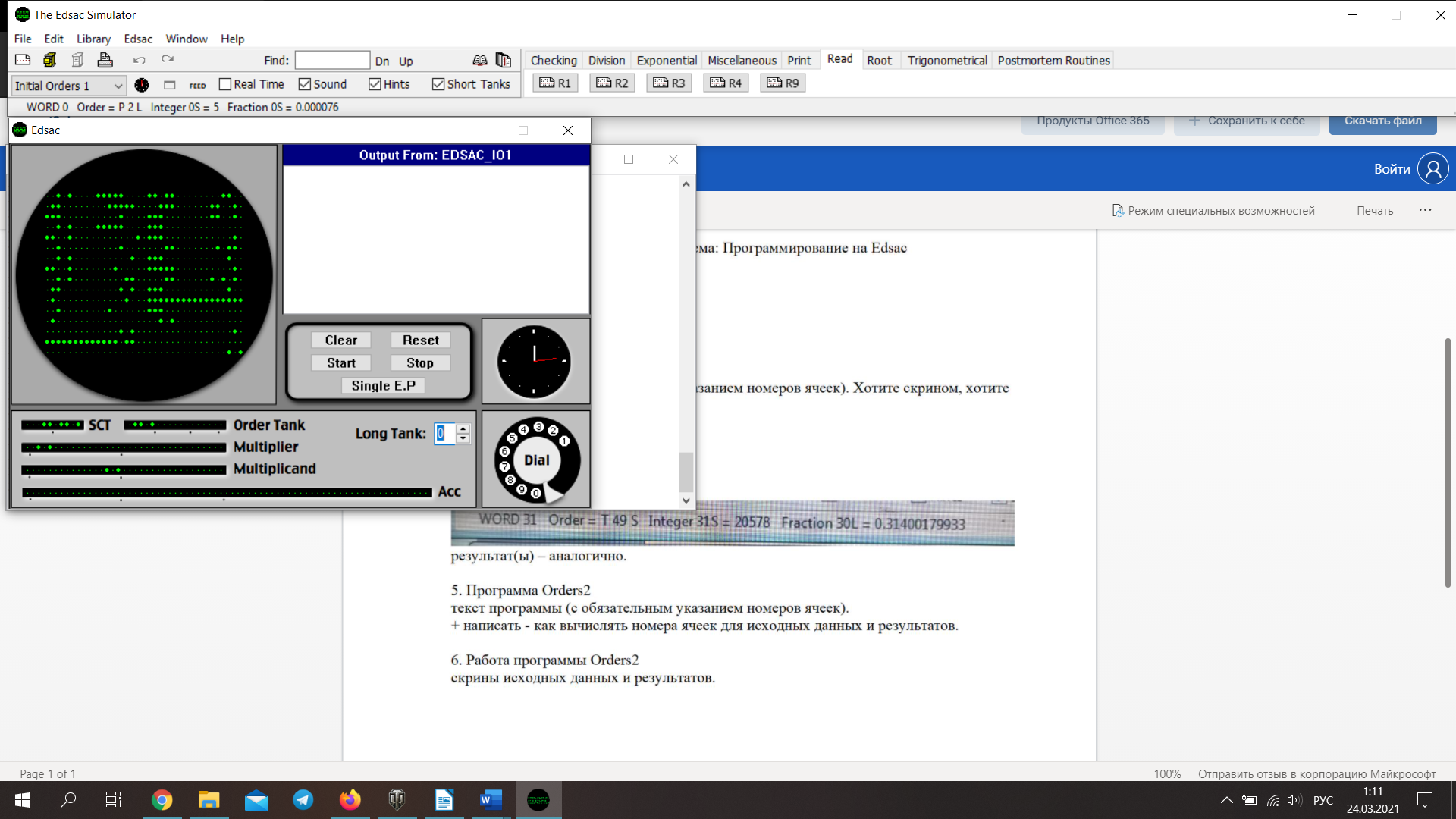








Результат работы:



# Программа Initial Orders 1

[Техническое задание - Определение K-ой порядковой статистики in-place]

[Результат хранится в ячейке 0]

[k задаётся в ячейке 52]

[k=8]

[массив находится в ячейках 55-63]

[Входные данные]

T 52[n] K [Размещаем, начиная с ячейки n = 52]

GK [@ = n]

[@+0] [k] P 4 F [Номер искомой порядковой статистики]

[@+1] [len] P 4 D [Длина массива 9]

[@+2] [addr]P 27 D [Адрес нулевого элемента массива]

[@+3] P 4 F [8]

[@+4] P 3 F [6]

[@+5] P 0 D [1]

[@+6] P 1 D [3]

[@+7] P 1 F [2]

[@+8] P 2 D [5]

[@+8] P 4 D [9]

[@+9]P 2 F [4]

[@+10]P 3 D [7]

[Сортировка]

[Args&Out: ]

[ len <- 53]

[ addr <- 54]

[Uses: ]

[ temp -> 0]

[ bin -> 1]

[Local: ]

[ counter 0 -> @0]

[ counter 1 -> @1]

[ c1 ->@67]

[ c2 ->@68]

[ шаблон r1 ->@69]

[ шаблон s1 ->@70]

[ шаблон w1 ->@71]

T 150[n] K [Размещаем, начиная с ячейки n]

GK [@ = n]

[@+0] [counter 0] A 3 F [Инструкция возврата (формируется в аккумуляторе)]

[@+1] [counter 1] T 66[endSort] @ [Записываем инструкцию возврата в конец]

[@+2] X 0 F

[@+3] X 0 F

[@+4] X 0 F

[@+5] X 0 F

[Здесь аккумулятор обнулен]

[@+6] A 53[len] F [Загрузка в аккумулятор длины обрабатываемого массива]

[@+7] S 67[c1] @ [Уменьшаем значение аккумулятора на 1 (сортировка перебором количества пар элементов. Пар меньше длины на 1)]

[@+8] T 0 @ [Запись этого значения в counter 0, обнуление аккумулятора]

[loop 0]

[Проходим по циклу (N-1) раз]

[@+9] [loop 0] T 1[bin] F [Очистка аккумулятора]

[@+10] A 0 @ [Загружаем счетчик необработанных элементов массива]

[@+11] S 67[c1] @ [Уменьшаем на 1]

[@+12] G 65[exit loop 0] @ [Если результат меньше 0, завершаем цикл]

[@+13] T 0[counter 0] @ [Обновляем значение счетчика и обнуляем аккумулятор]

[Устанавливаем счетчик вложенного цикла в (N-1)]

[@+14] A 53[len] F [Загрузка в аккумулятор длины обрабатываемого массива]

[@+15] S 67[c1] @ [Уменьшаем значение аккумулятора на 1 (сортировка перебором количества пар элементов. Пар меньше длины на 1)]

[@+16] T 1[counter 1] @ [Запись этого значения в counter 1, обнуление аккумулятора]

[Установка адресов в инструкциях на 0 и 1 элементы массива (изменяем адреса во вложенном цикле)]

[Инициализируем инструкции r1 и r2 по шаблонам]

[@+17] A 54[addr] F [Загрузка в аккумулятор адреса 0-го элемента массива]

[@+18] L 0 D [Сдвиг аккумулятора на 1 разряд влево]

[@+19] A 69[шаблон r1] @ [Прибавляем код инструкции с нулевым полем адреса]

[@+20] U 38[r1] @ [Запись сформированной инструкции без обнуления аккумулятора]

[@+21] A 68[c2] @ [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес 1-го элемента массива)]

[@+22] T 43[r2] @ [Запись сформированной инструкции, обнуление аккумулятора]

[Инициализируем инструкции w1 и w2]

[@+23] A 54[addr] F [Загрузка в аккумулятор адреса 0-го элемента массива]

[@+24] L 0 D [Сдвиг аккумулятора на 1 разряд влево]

[@+25] A 71[шаблон w1] @ [Прибавляем код инструкции с нулевым полем адреса]

[@+26] U 44[w1] @ [Запись сформированной инструкции без обнуления аккумулятора]

[@+27] A 68[c2] @ [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес 1-го элемента массива)]

[@+28] T 46[w2] @ [Запись сформированной инструкции, обнуление аккумулятора]

[Инициализируем инструкцию s1]

[@+29] A 54[addr] F [Загрузка в аккумулятор адреса 0-го элемента массива]

[@+30] L 0 D [Сдвиг аккумулятора на 1 разряд влево]

[@+31] A 70[шаблон s1] @ [Прибавляем код инструкции с нулевым полем адреса]

[@+32] T 40[s1] @ [Запись сформированной инструкции, обнуление аккумулятора]

[loop 1: Проход по массиву с перестановками (еще один вложенный цикл)]

[Счетчик цикла храним в ячейке 2]

[Здесь предполагается, что аккумулятор обнулен]

[@+33] X 0 F [Останов для отладки]

[@+34] A 1[counter 1] @ [Загружаем счетчик необработанных элементов массива]

[@+35] S 67[c1] @ [Уменьшаем на 1]

[@+36] G 9[loop 0] @ [Если результат меньше 0, завершаем цикл и уходим в основную ветку]

[@+37] T 1[counter 1] @ [Обновляем значение счетчика и обнуляем аккумулятор]

[Перестановка элементов при необходимости]

[Здесь предполагается, что аккумулятор обнулен]

[@+38] [r1] A 0[N] F [Загрузка в аккумулятор значения из ячейки N]

[@+39] U 0[temp] F [Запись этого значения в рабочую ячейку без обнуления аккумулятора]

[Проверяем, надо ли переставлять числа]

[@+40] [s1] S 0[N+1] F [Вычитание из аккумулятора значения следующей ячейки (N+1). Acc = A(N)-A(N+1)]

[@+41] G 47[else] @ [Если A(N+1)>A(N) (acc<0), то переставлять элементы местами не нужно. Пропускаем этот шаг]

[@+42] T 1[bin] F [Очистка аккумулятора]

[Перестановка чисел в смежных ячейках]

[@+43] [r2] A 0[N+1] F [Загрузка в аккумулятор значения из ячейки N+1]

[@+44] [w1] T 0[N] F [Запись этого значения в ячейку с адресом N, обнуление аккумулятора]

[@+45] A 0[temp] F [Загрузка в аккумулятор значения из temp]

[@+46] [w2] T 0[N+1] F [Запись этого значения в ячейку с адресом N+1, обнуление аккумулятора]

[Аккумулятор обнулен]

[else]

[@+47] T 1[bin] F [Очистка аккумулятора]

[@+48] X 0 F [Останов для отладки]

[Теперь обновляем адреса в инструкциях r1, r2, w1, w2, s1:]

[Обновляем r1]

[@+49] A 38[r1] @ [Загрузка в аккумулятор инструкции]

[@+50] A 68[c2] @ [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+1-го элемента массива)]

[@+51] T 38[r1] @ [Запись сформированной инструкции, обнуление аккумулятора]

[Обновляем r2]

[@+52] A 43[r2] @ [Загрузка в аккумулятор инструкции]

[@+53] A 68[c2] @ [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+2-го элемента массива)]

[@+54] T 43[r2] @ [Запись сформированной инструкции, обнуление аккумулятора]

[Обновляем w1]

[@+55] A 44[w1] @ [Загрузка в аккумулятор инструкции]

[@+56] A 68[c2] @ [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+1-го элемента массива)]

[@+57] T 44[w1] @ [Запись сформированной инструкции, обнуление аккумулятора]

[Обновляем w2]

[@+58] A 46[w2] @ [Загрузка в аккумулятор инструкции]

[@+59] A 68[c2] @ [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+2-го элемента массива)]

[@+60] T 46[w2] @ [Запись сформированной инструкции, обнуление аккумулятора]

[Обновляем s1]

[@+61] A 40[s1] @ [Загрузка в аккумулятор инструкции]

[@+62] A 68[c2] @ [Добавление константы (2 shiftR) = 1 к адресному полю сформированной инструкции (теперь там адрес N+2-го элемента массива)]

[@+63] T 40[s1] @ [Запись сформированной инструкции, обнуление аккумулятора]

[@+64] E 33[loop 1] @ [Повторяем все операции; аккумулятор обнулен]

[exit loop 0]

[@+65] T 1[bin] F [Обнуляем аккумулятор]

[@+66] [endSort] E 0 F [<-Сюда записывается инструкция возврата]

[@+67] [c1] P 0 D [Константа 1]

[@+68] [c2] P 1 F [Константа 2]

[Шаблоны инструкций для загрузки в loop]

[@+69] [шаблон r1] A 0 F [Шаблон чтения (сложения)]

[@+70] [шаблон s1] S 1 F [Шаблон вычитания (адрес+1)]

[@+71] [шаблон w1] T 0 F [Шаблон записи]

[Поиск k-ой порядковой статистики]

[Args: ]

[ k <- 52]

[ len <- 53]

[ addr <- 54]

[Uses: ]

[ sort -> 0]

[ sort -> 1]

[Local: ]

[ c1 ->@19]

[ шаблон r1 ->@20]

[Out: ]

[ res -> 0]

T 250[n] K [Размещаем, начиная с ячейки n]

GK [@ = n]

[@+0] A 3 F [Инструкция возврата (формируется в аккумуляторе)]

[@+1] T 18[endSearch] @ [Записываем инструкцию возврата в конец]

[@+2] X 0 F

[@+3] X 0 F

[Аккумулятор обнулен]

[Вызываем сортировку массива]

[@+4] A 4 @

[@+5] G 150[Сортировка] F

[Формирование адреса k-ой порядковой статистики]

[@+6] T 0 F [Обнуление аккумулятора]

[@+7] X 0 F [Останов для отладки]

[@+8] A 52[k] F [Загрузка значения k в аккумулятор]

[@+9] S 19[c1] @ [Вычитаем константу 1]

[@+10] A 54[addr] F [Получаем адрес k-ой порядковой статистики]

[@+11] X 0 F [Останов для отладки]

[Сохраняем статистику в ячейку res]

[@+12] L 0 D [Сдвиг влево на 1 разряд]

[@+13] A 20[шаблон r1] @ [Формируем инструкцию чтения k-ой порядковой статистики из ее адреса]

[@+14] T 15[next] @ [Записываем инструкцию в следующую ячейку, обнуляем аккумулятор]

[@+15] [next] A 0 F [Инструкция для перезаписи; Читаем статистику из ячейки по найденному адресу]

[@+16] T 0[res] F [Записываем k-ую порядковую статистику в res, обновляем аккумулятор]

[k-ая порядковая статистика хранится в ячейке 0]

[@+17] X 0 F [Останов для отладки]

[@+18] [endSearch] E 0 F [<-Сюда записывается инструкция возврата]

[@+19] [c1] P 0 D [Константа 1]

[@+20] [шаблон r1] A 0 F [Шаблон чтения (сложения)]

[Программа]

T 100[n] K [Размещаем, начиная с ячейки n]

GK [@ = n]

[@+0] Z 0 F [Останов для отладки]

[@+1] T 0 F [Acc=0]

[Вызов подпрограммы (поиск статистики)]

[@+2] A 2 @

[@+3] G 250[Поиск k-ой порядковой статистики] F

[k-ая порядковая статистика хранится в ячейке 0]

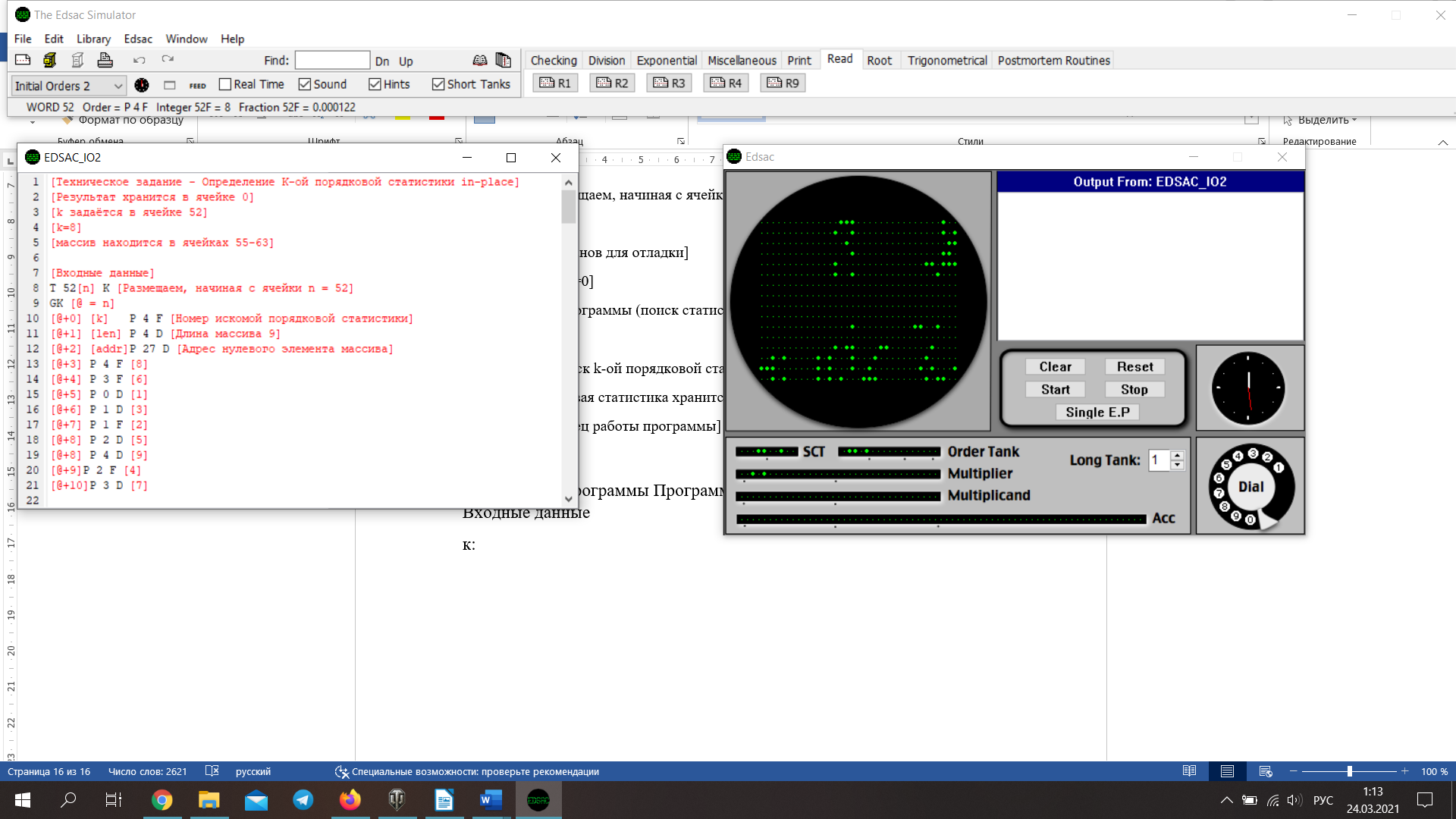
[@+4] Z 0 F [Конец работы программы]

[@+5] EZPF

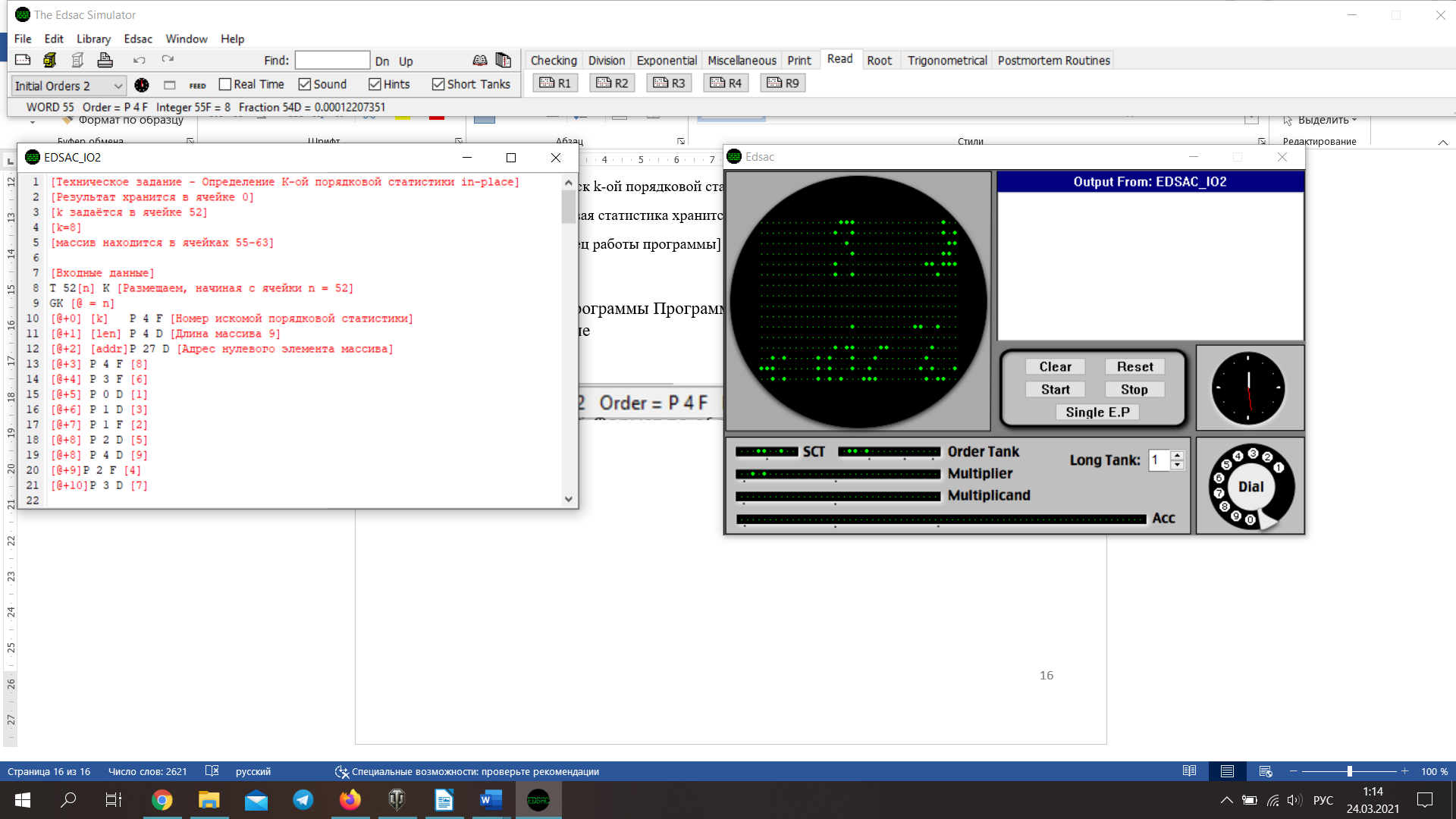
# Работа программы Программа Initial Orders 2

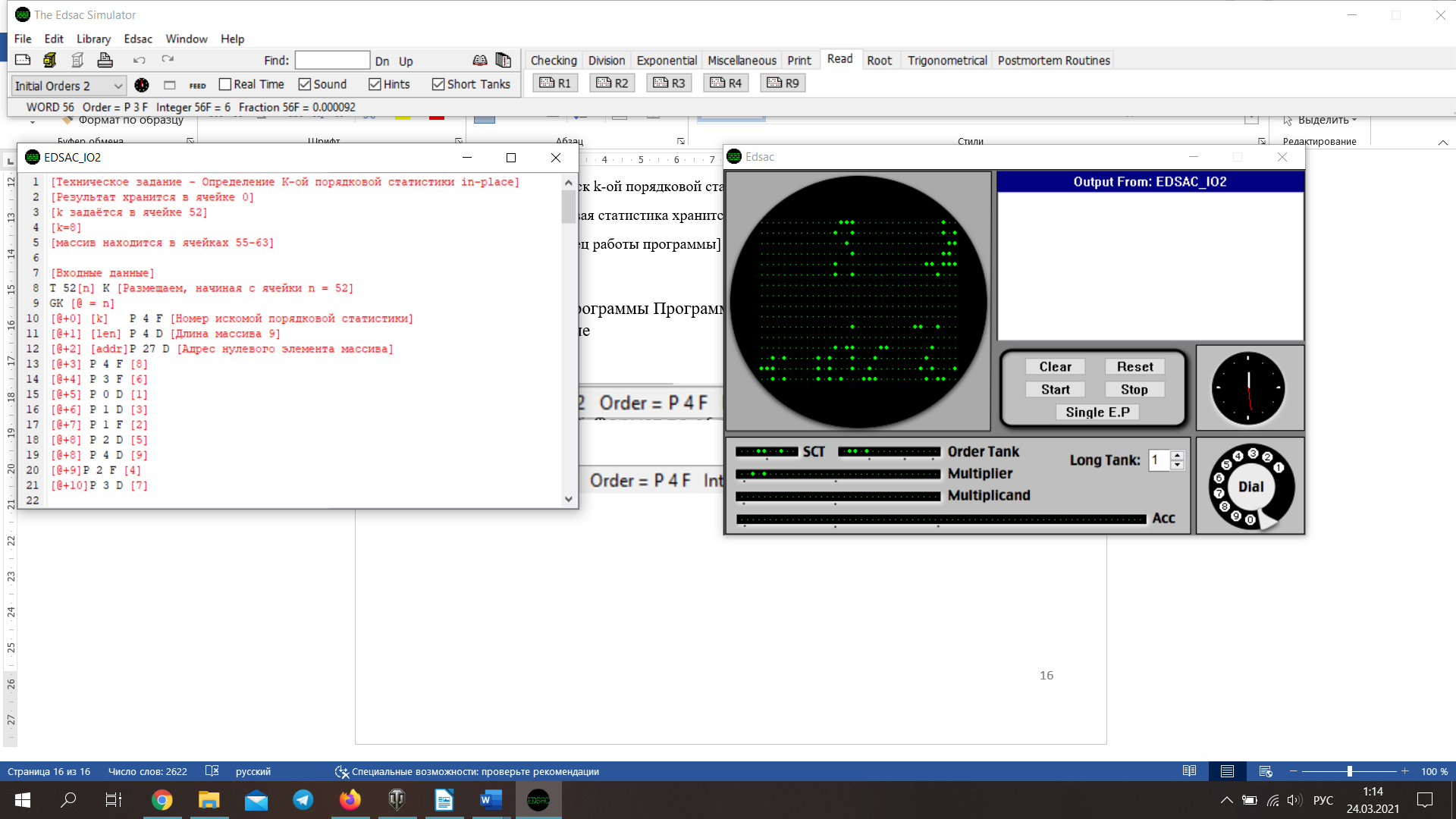
Входные данные

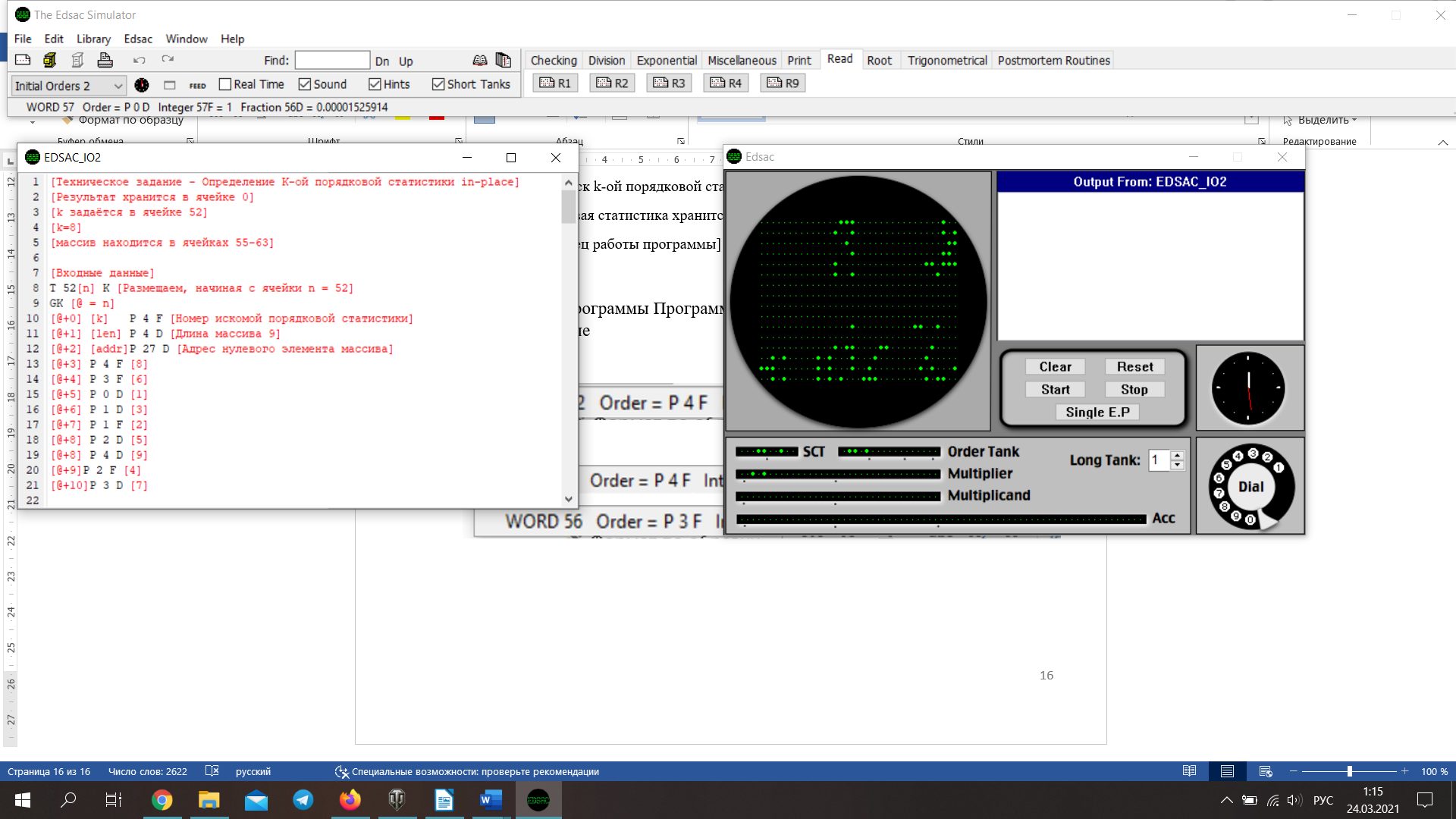
к:

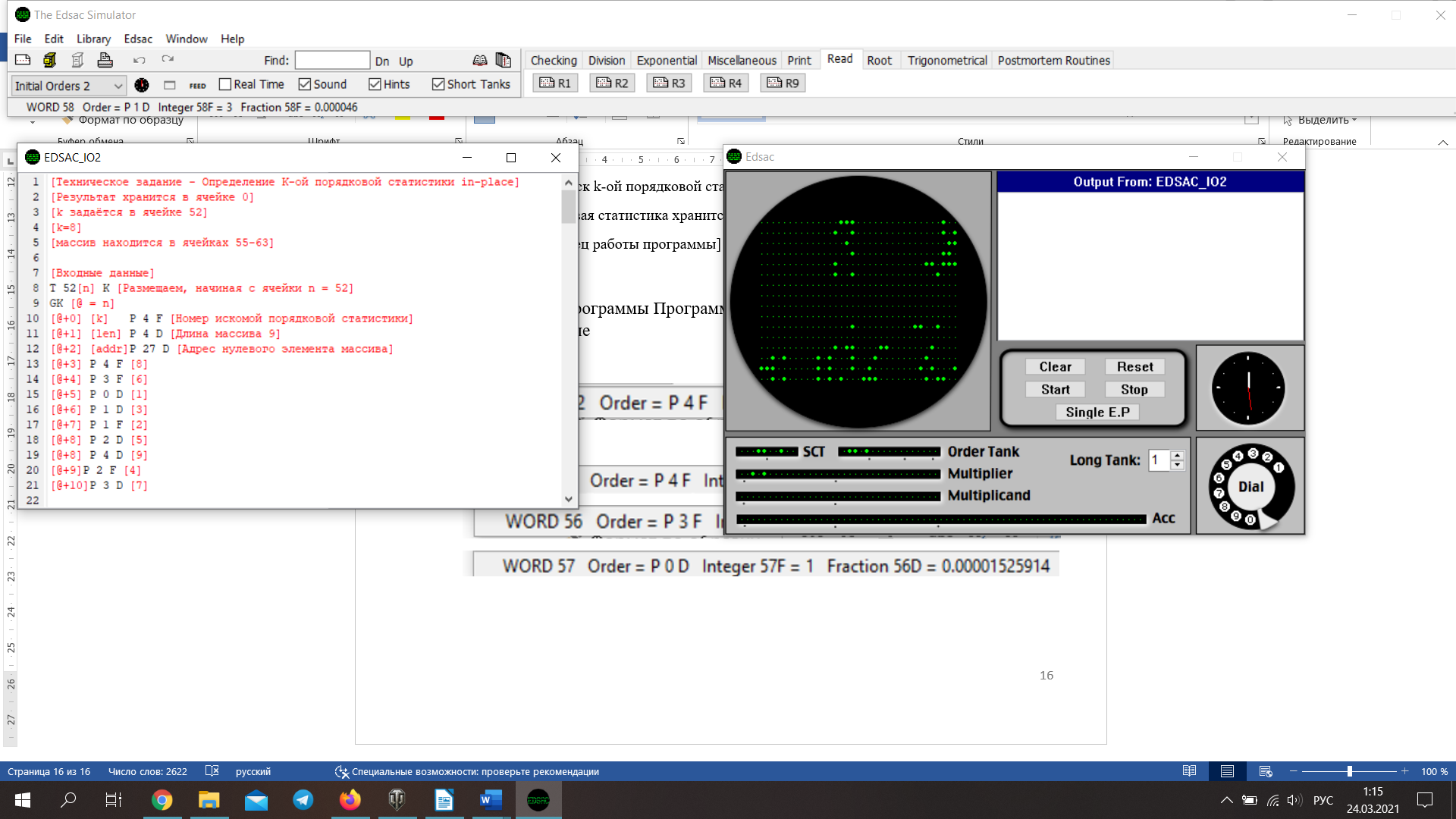


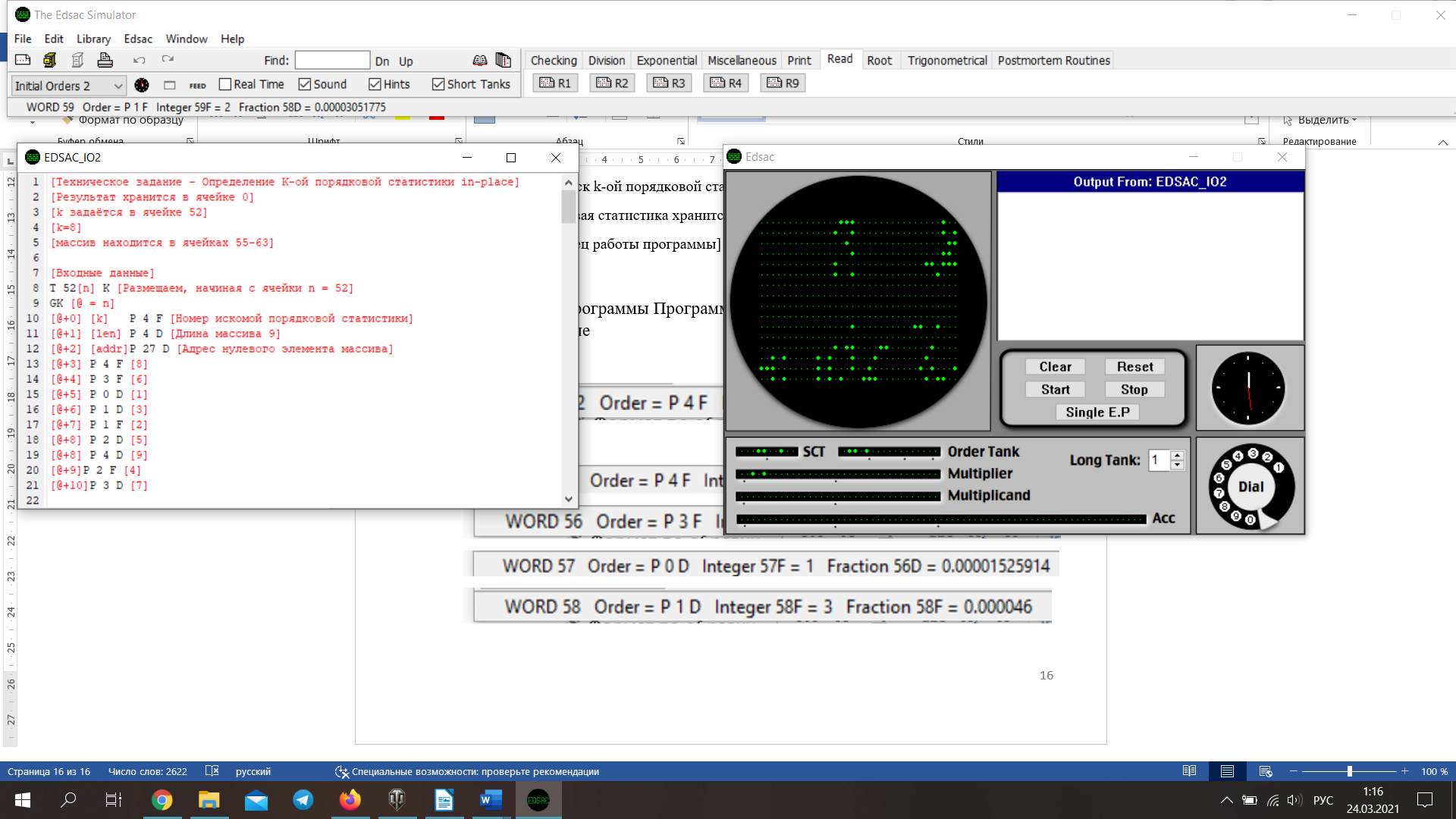
Массив:

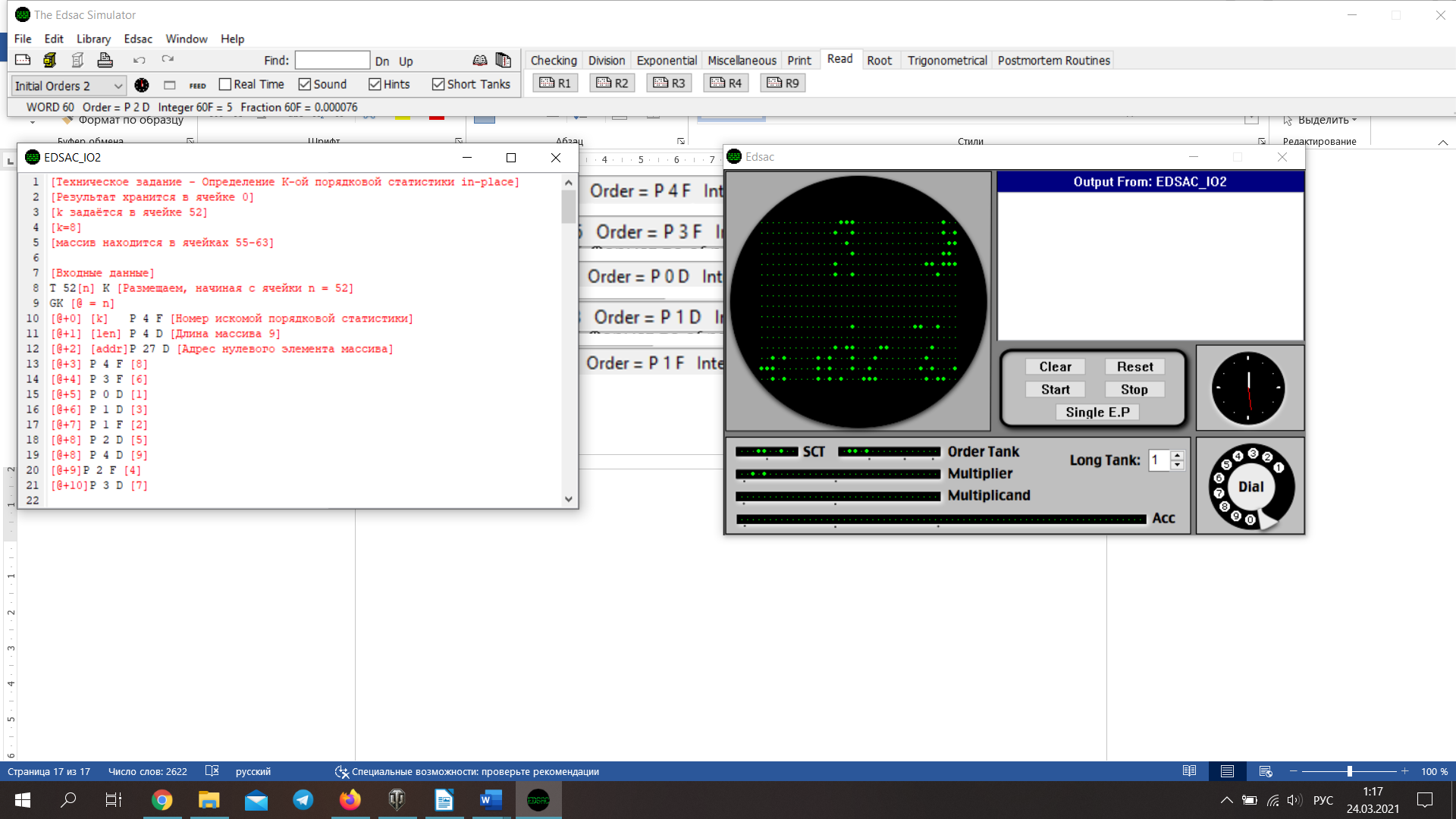


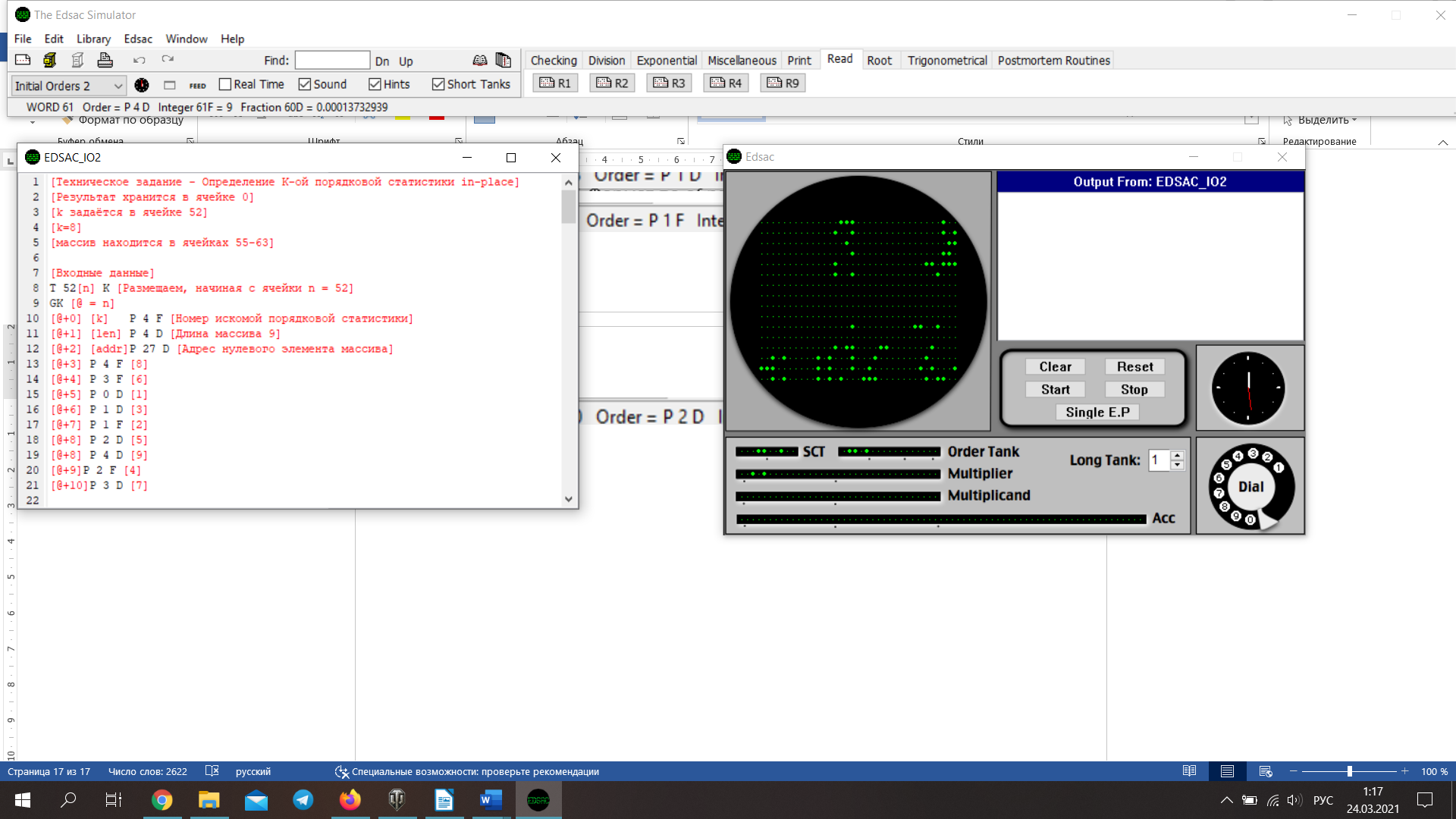


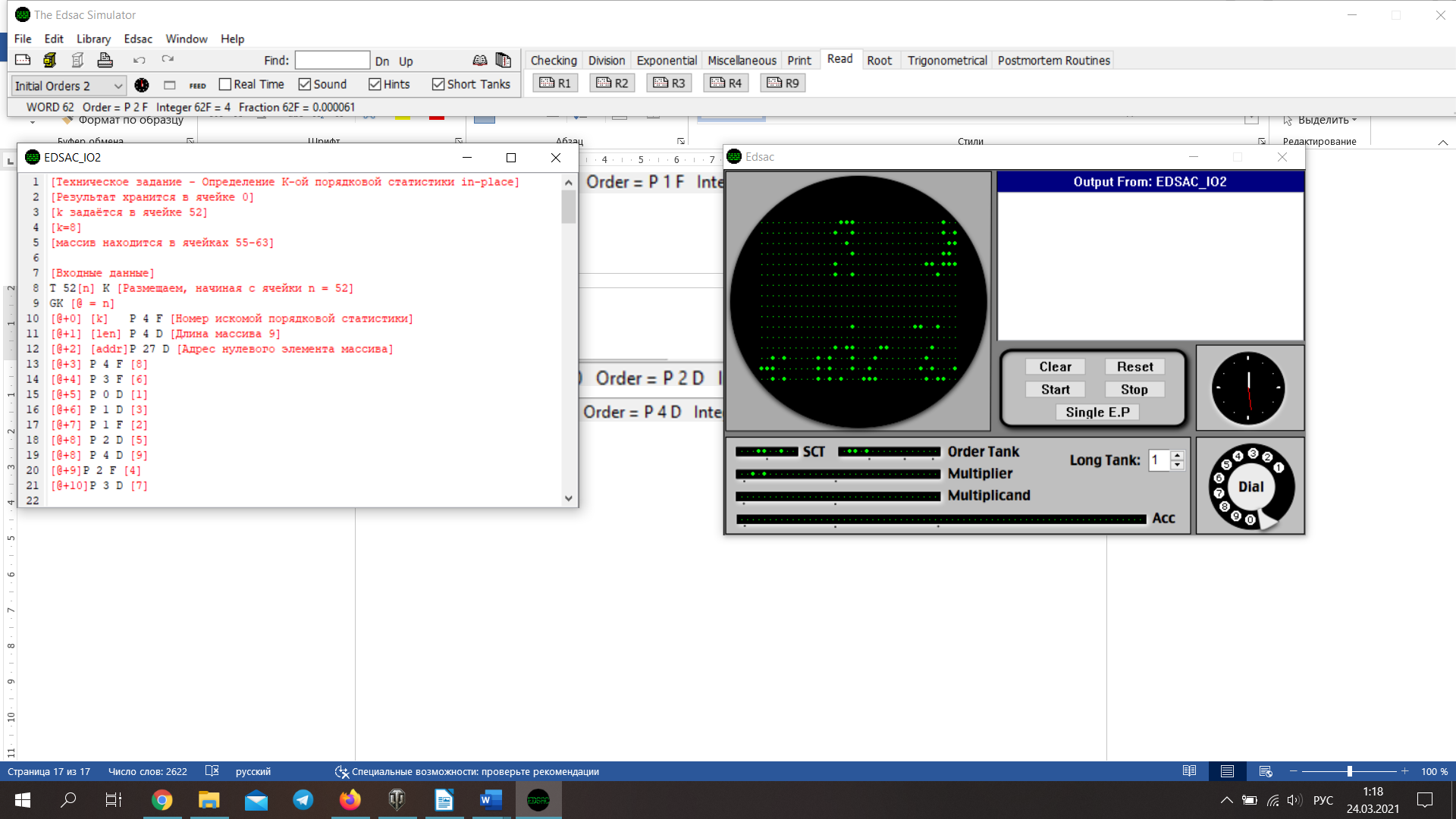


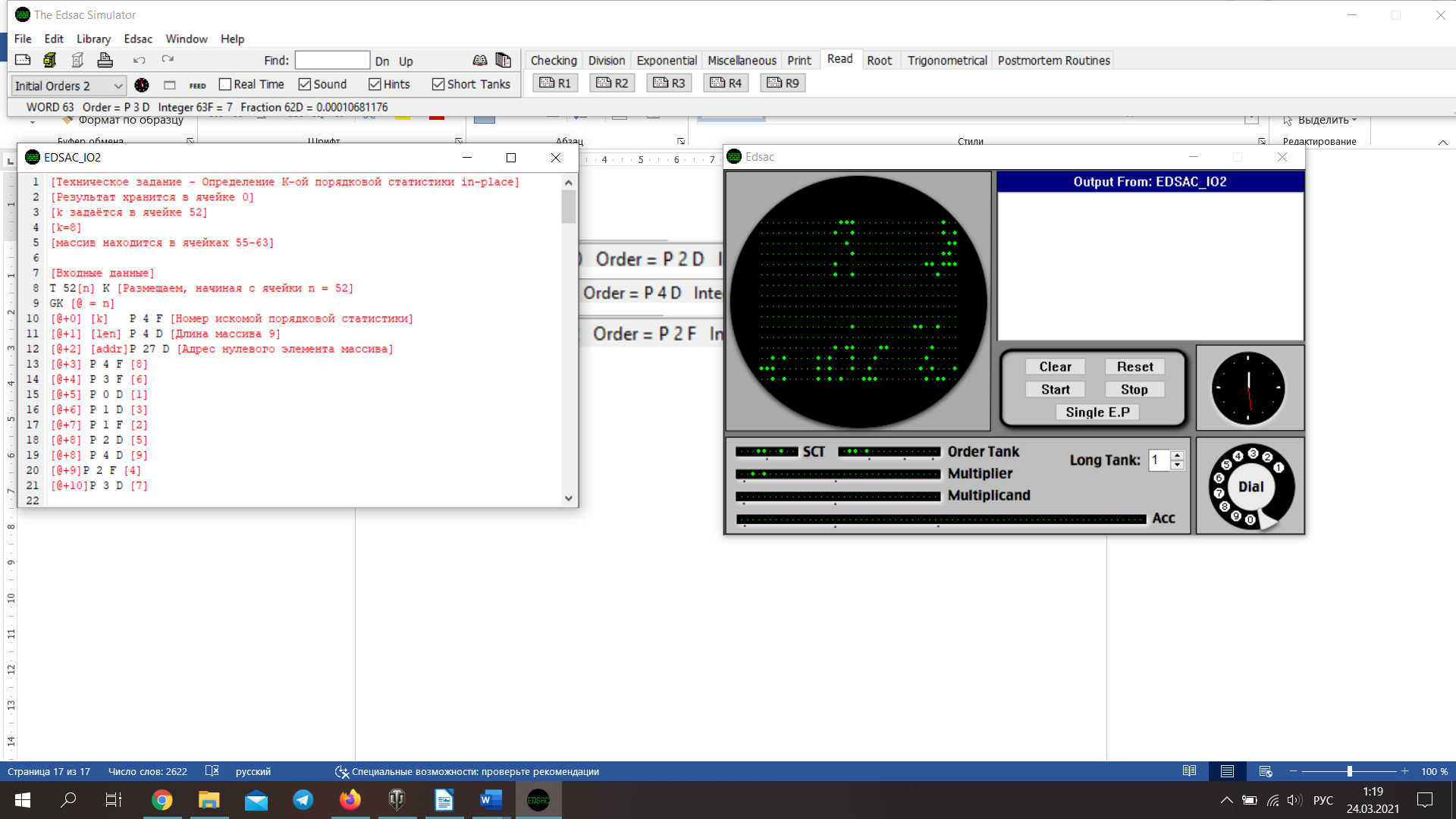




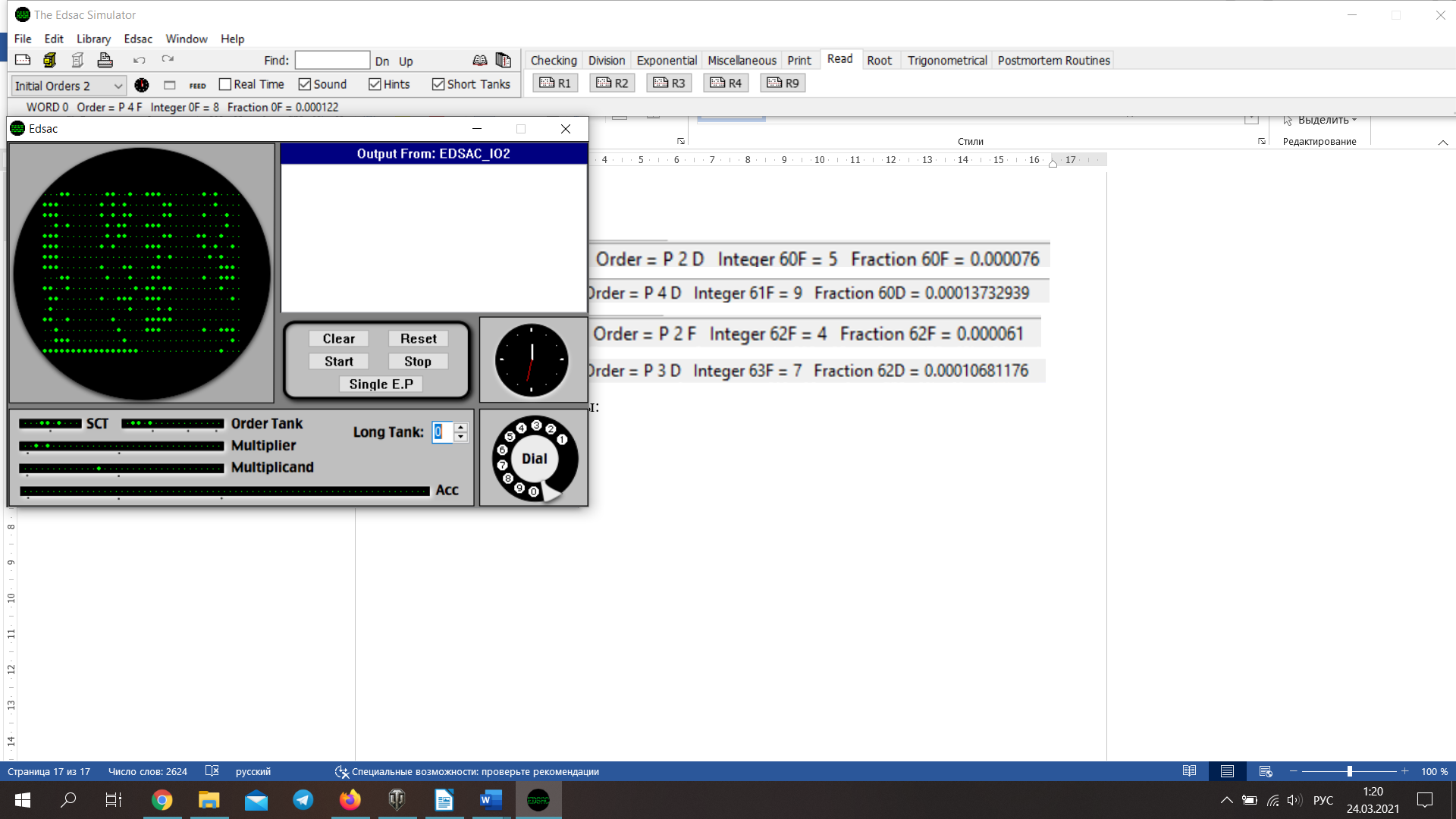








Результат работы:



# Работа Адреса и правила кодирования исходных данных и результатов. Руководство программиста

* 1. Загрузчик Initial Orders 1

Чтобы воспользоваться программой, необходимо:

* Записать исследуемый массив, начиная с 118 строки.
* В 112 строку записать длину массива.
* В строку 114 записать значение k.

Результат выполнения программы будет записан в ячейке 0.

* 1. Загрузчик InitialOrders 2

Чтобы воспользоваться программой, необходимо:

* Записать исследуемый массив, начиная с 13 строки

([@+3] P 4 F [8]

[@+4] P 3 F [6]

…)

Таким образом массив будет лежать с 55 ячейки, так как заданы относительные адреса – [@+3] P 4 F [8]; @=52 => 52+3=55

* В 11 строку записать длину массива

[@+1] [len] P 4 D [Длина массива 9]

То есть длина массива лежит в 53 ячейке.

Аналогично тому, что выше – @=52 => 52+1=53

* В строку 11 записать значение k.

[@+0] [k] P 4 F [Номер искомой порядковой статистики]

Значение к будет находится в ячейке 52, так как @=52 => @+0=52.

Результат выполнения программы будет записан в ячейке 0.