МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное

образовательное учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт компьютерных технологий и информационной безопасности

Кафедра информационно-аналитических систем безопасности

имени Л.С. Берштейна

**ОТЧЕТ**

**Лабораторная работа №2**

**Основы алгоритмизации и программирования**

**«Обработка массивов»**

**Вариант №4**

Выполнил:

студент гр. КТбо1-8

Костюков М.Д.

«\_\_\_» 2017 г.

Проверил:

Профессор кафедры ИАСБ

Беляков Станислав Леонидович

«\_\_\_» 2017 г.

Таганрог – 2017

# Оглавление

[Оглавление 2](#_Toc498587987)

[Цели работы. 3](#_Toc498587988)

[Выполнение работы 4](#_Toc498587989)

[Программный код 5](#_Toc498587990)

[Тестирование программы с разными значениями на входе 6](#_Toc498587991)

[Вывод 8](#_Toc498587992)

# Цели работы.

1. Изучить интегрированную среду разработки и научиться её использовать.

2. Написать программу, которая вычисляет сумму всех положительных элементов массива. Массив и его длина вводятся пользователем.

# Выполнение работы

1. Подключаем необходимые библиотеки.
2. Объявляем переменные: размер массива.
3. Просим ввести количество элементов в массиве.
4. Проверяем корректность ввода. Если введены некорректные данные, то сообщаем об ошибке и просим ввести число заново.
5. Выделяем память для динамического массива.
6. Просим пользователя по очереди ввести все элементы массива.
7. Проверяем вводимые данные на корректность. Если введены некорректные данные, то сообщаем об ошибке и просим повторить ввод.
8. Суммируем значения введенные пользователем.
9. Если значения отрицательные – выводим сообщение об ошибке.
10. Задерживаем консоль.

# Программный код

#include <stdio.h>

#include <stdbool.h>

#include <locale.h>

#define MAX\_LENGTH 50

#define MIN\_LENGTH 2

int main()

{

int i;

int sum = 0;;

int array[MAX\_LENGTH];

int length;

int check;

setlocale( LC\_ALL,"rus");

printf("Введите размер массива: ");

while (!scanf("%d", &length) || (length > MAX\_LENGTH) || (length < MIN\_LENGTH))

{

printf("Вы ввели не допустимое значение. \n Введите значение в диапазоне от 2 до 50: ");

while (getchar() != '\n');

}

while (getchar() != '\n');

for (check = 0; check < length; check++)

{

printf("Значение [%d]:", check);

while (!scanf("%d", &array[check]))

{

printf("Ошибка ввода. \n Введите: array [%d]: ", check);

while (getchar() != '\n');

}

while (getchar() != '\n');

}

for (check=0;check<length;check++)

{

if(check % 2==0)

sum+=array[check];

}

printf("%d", check);

\_getch();

return 0;

}

# Тестирование программы с разными значениями на входе

|  |  |  |  |
| --- | --- | --- | --- |
| Входные данные длинны массива | Входные данные самого массива | Выходные данные (сумма) | Номер рисунка |
| 3 | 1;2;3 | 6 | 1 |
| 4 | 6;8;12;14 | 40 | 2 |
| 5 | 5;4;3;2;1 | 15 | 3 |

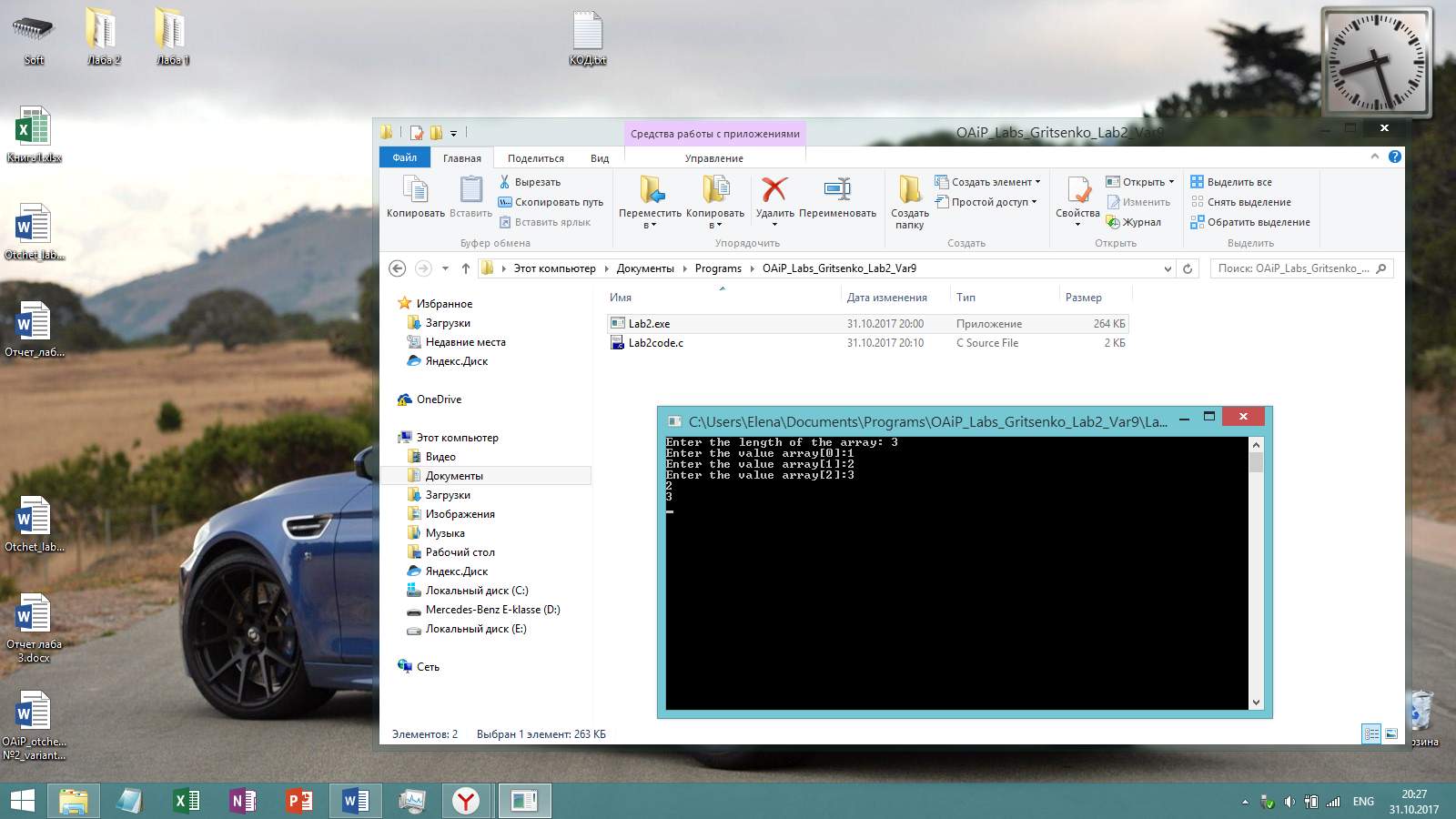


Рис. 1

Ввод первого набора данных и вывод результата

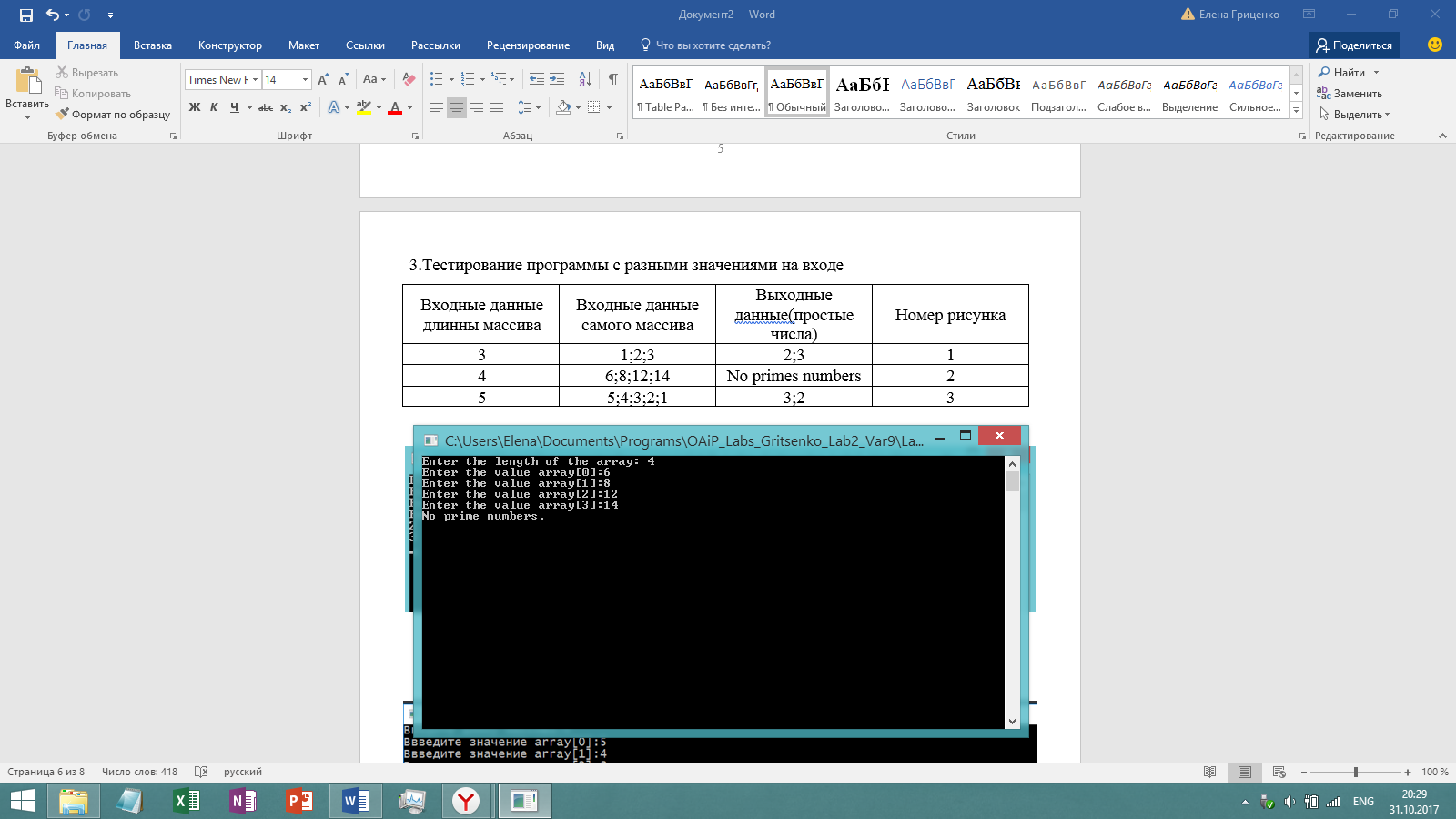


Рис. 2

Ввод второго набора данных и вывод результата (Отсутствие простых чисел)

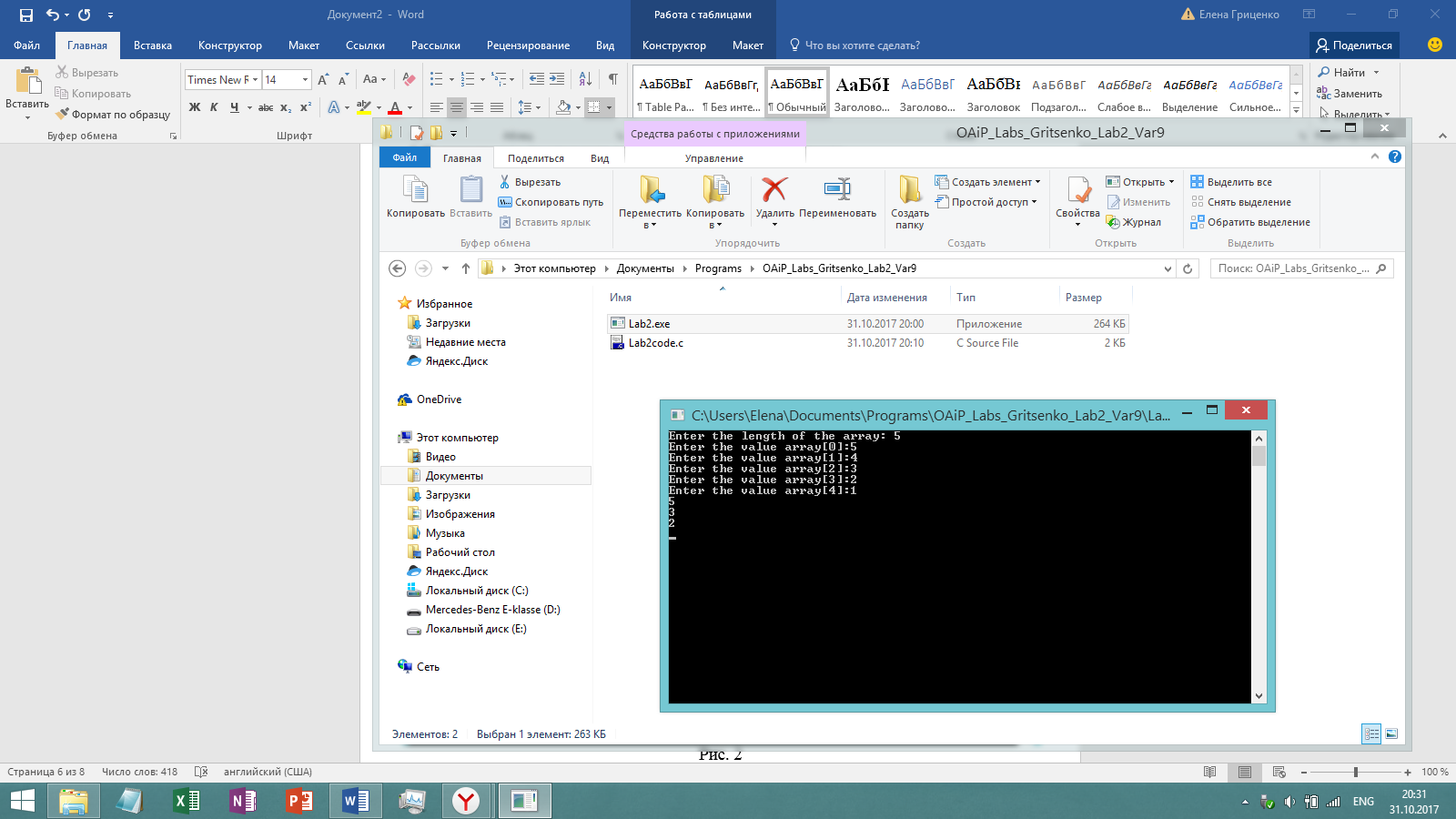


Рис. 3

Ввод третьего набора данных и вывод результата

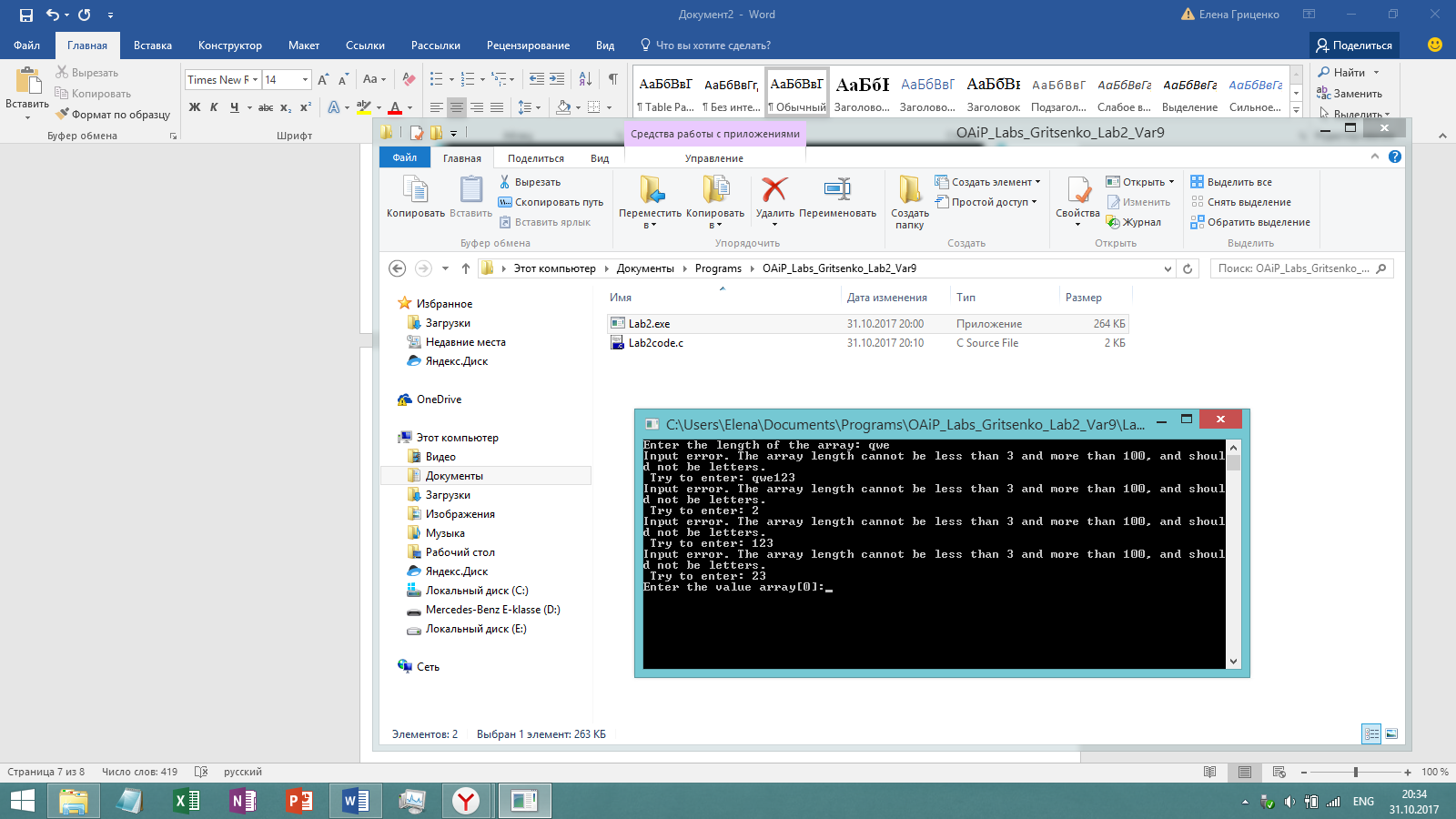


Рис. 4

Ввод не корректных данных

Контрольный пример

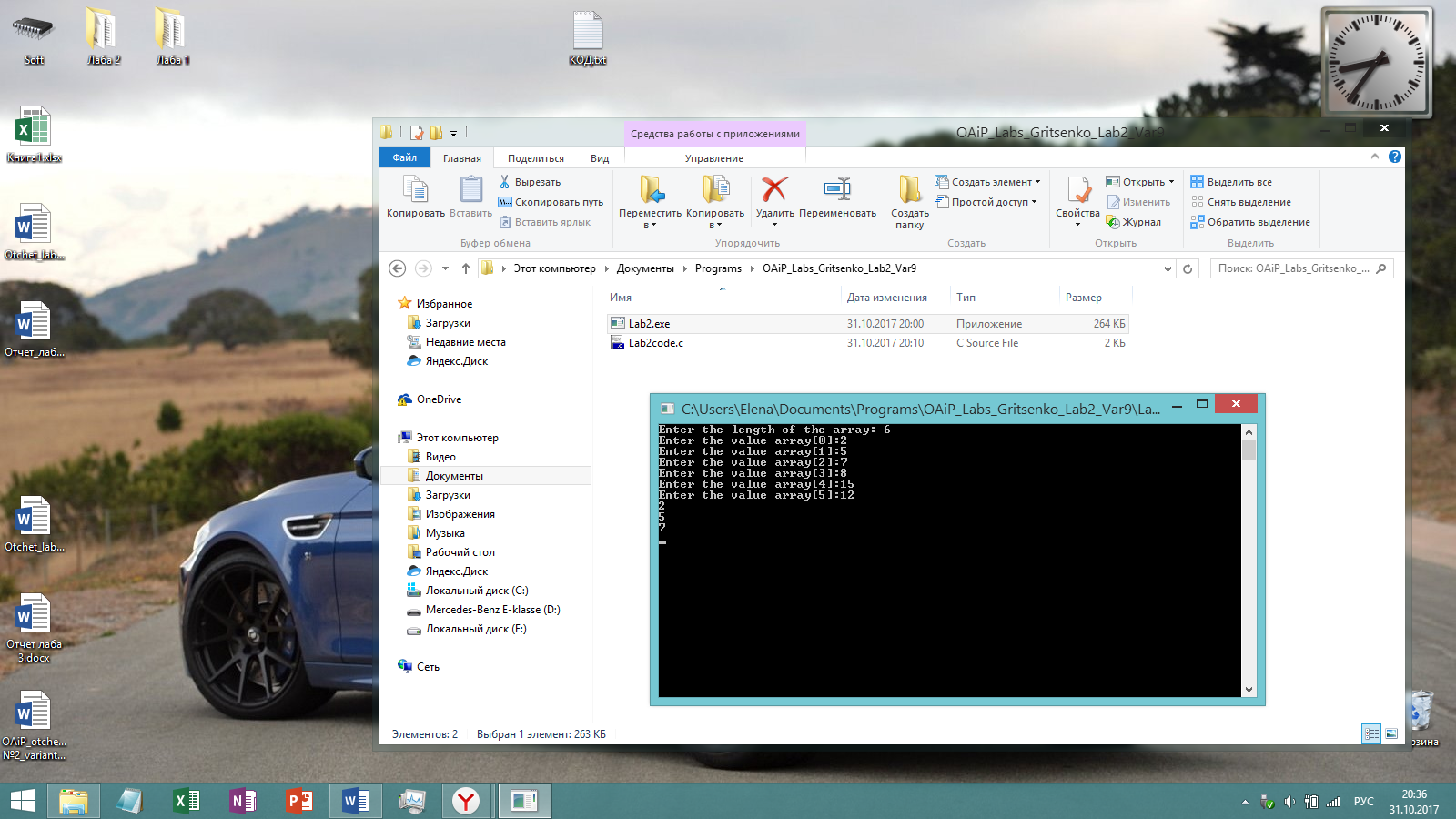


Рис. 5

Контрольный пример

# Вывод

В результате выполнения лабораторной №2 (Вариант 4) были изучены массивы, их обработка и был более глубоко изучен язык C. Так же была написана программа, в ходе выполнения которой пользователь задает длину массива и вводит значения его элементов, программа же выводит их сумму. Если введенные значения отрицательны выводиться «No primes numbers».