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 №9**

Выполнил:

студент гр. КТбо1-8

Гриценко Д. А.

«\_\_\_» 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 выводит на экран только простые числа. 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. Проходим по всем элементам массива и проверяем простые ли они.
9. Если элемент массива – простое число, выводим его на экран.
10. Если простых чисел нет – выводим об этом сообщение.
11. Задерживаем консоль.

# Программный код

#include <stdio.h>

#include <stdbool.h>

#define MAX\_LENGTH 100

#define MIN\_LENGTH 3

int main()

{

int i;

int j;

bool are\_simple=0;

int array[MAX\_LENGTH];

int length;

int check;

printf("Enter the length of the array: ");

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

{

printf("Input error. The array length cannot be less than 3 and more than 100, and should not be letters. \n Try to enter: ");

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

}

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

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

{

printf("Enter the value array[%d]:", check);

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

{

printf("Input error. \n Try to enter: array[%d]: ", check);

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

}

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

}

for (i = 0; i < array[MAX\_LENGTH]; i++) {

if ((array[i]) <= 3 && (array[i]) > 1) {

printf("%d\n", array[i]);

are\_simple=1;

}

else {

for (j = 2; j < (array[i]); j++) {

if (array[i] % j == 0) {

break;

}

if (j == (array[i]) - 1) {

printf("%d\n", array[i]);

are\_simple=1; }

}

}

}

if (are\_simple==0) {

printf("No prime numbers."); }

\_getch();

return 0;

}

# Тестирование программы с разными значениями на входе

|  |  |  |  |
| --- | --- | --- | --- |
| Входные данные длинны массива | Входные данные самого массива | Выходные данные (простые числа) | Номер рисунка |
| 3 | 1;2;3 | 2;3 | 1 |
| 4 | 6;8;12;14 | No primes numbers | 2 |
| 5 | 5;4;3;2;1 | 5;3;2 | 3 |

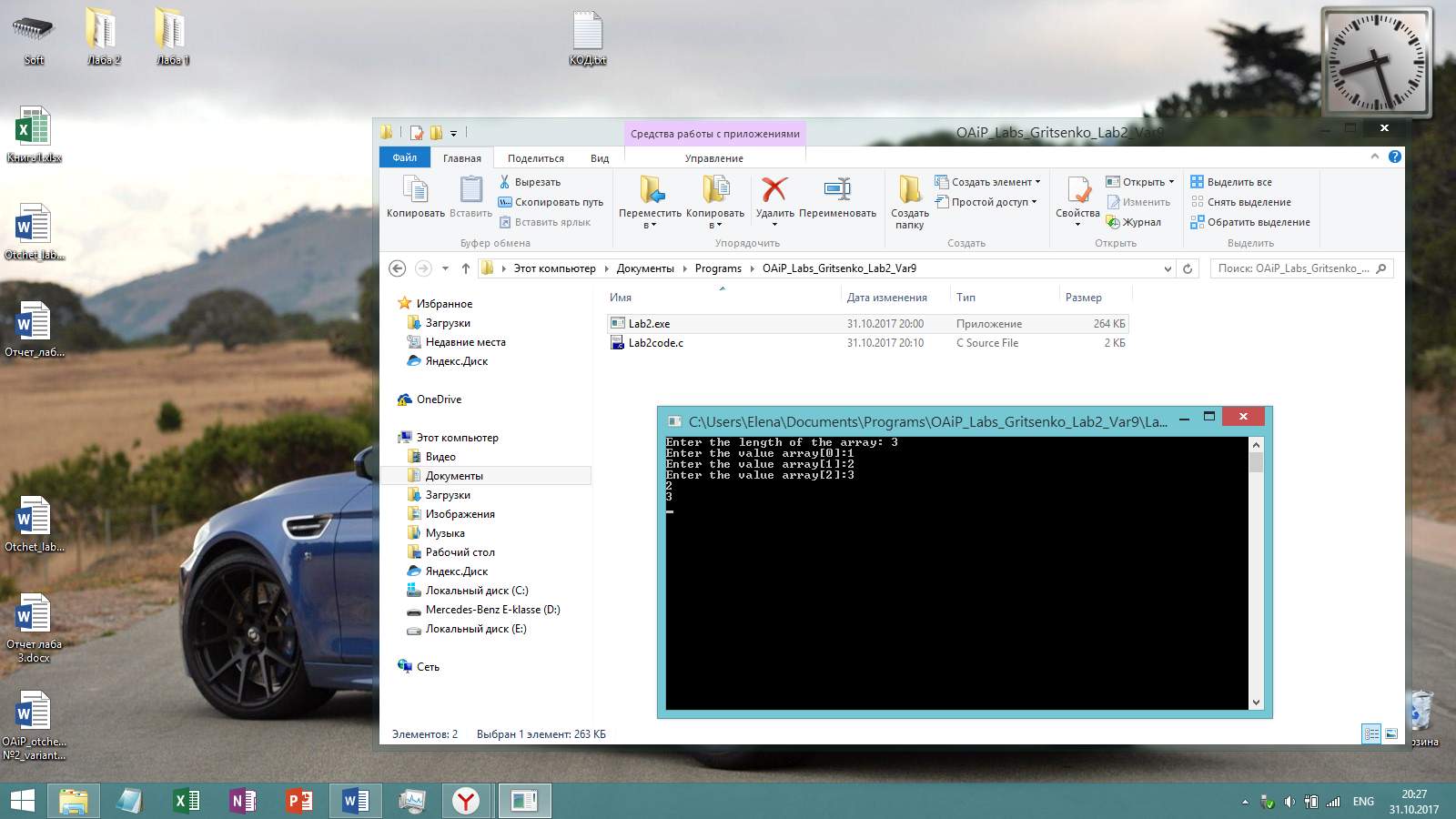


Рис. 1

Ввод первого набора данных и вывод результата

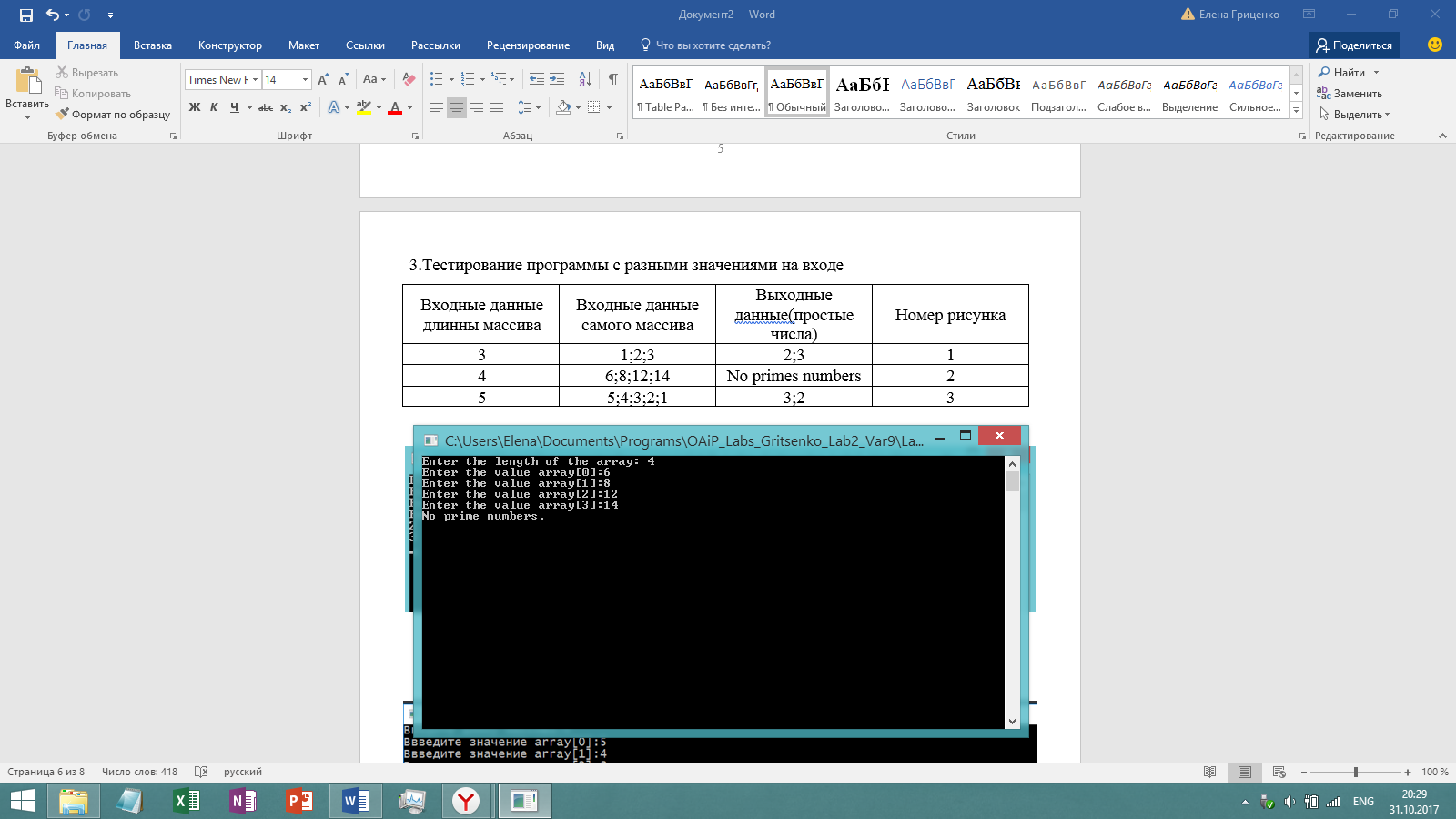


Рис. 2

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

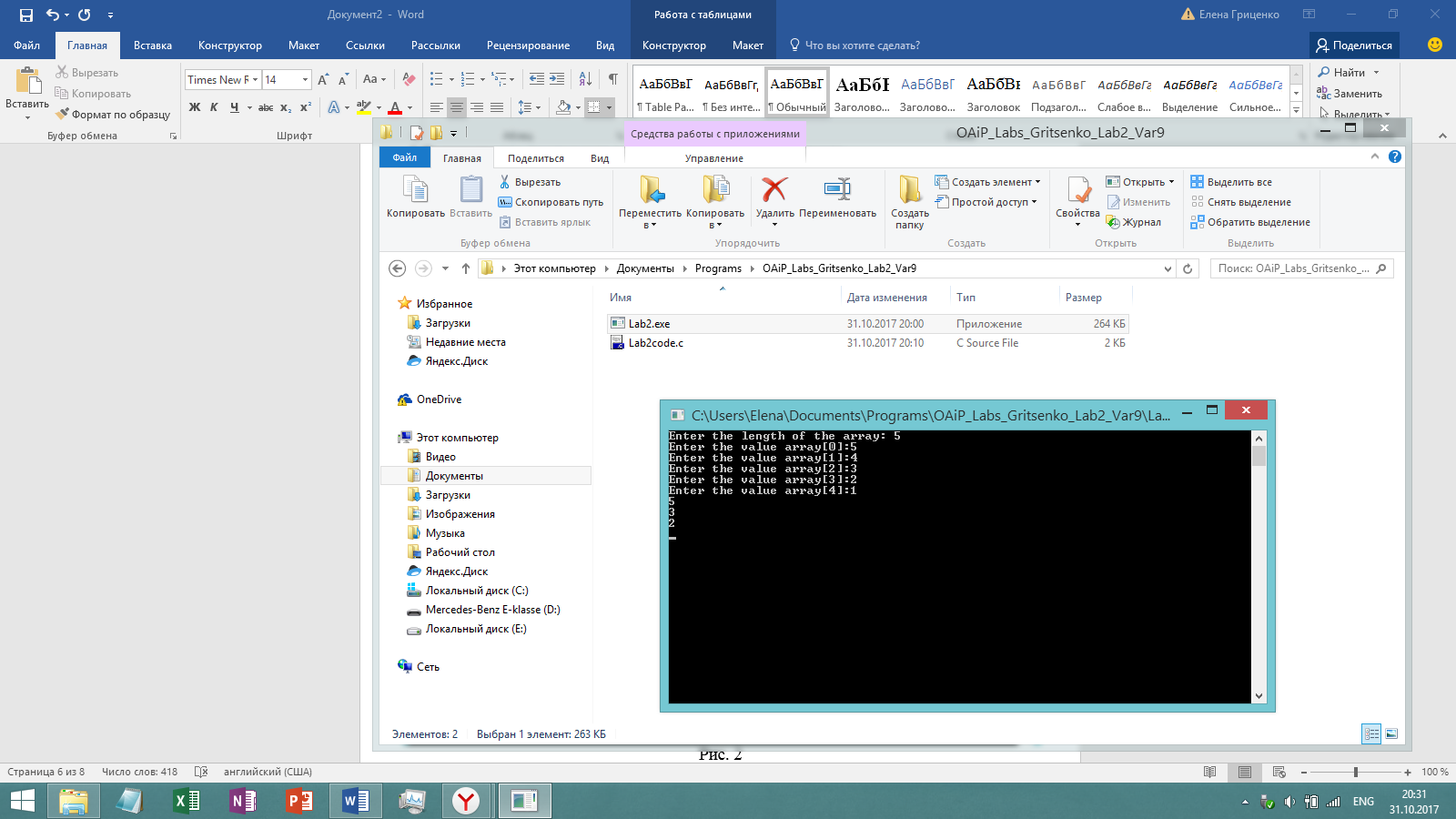


Рис. 3

Ввод третьего набора данных и вывод результата

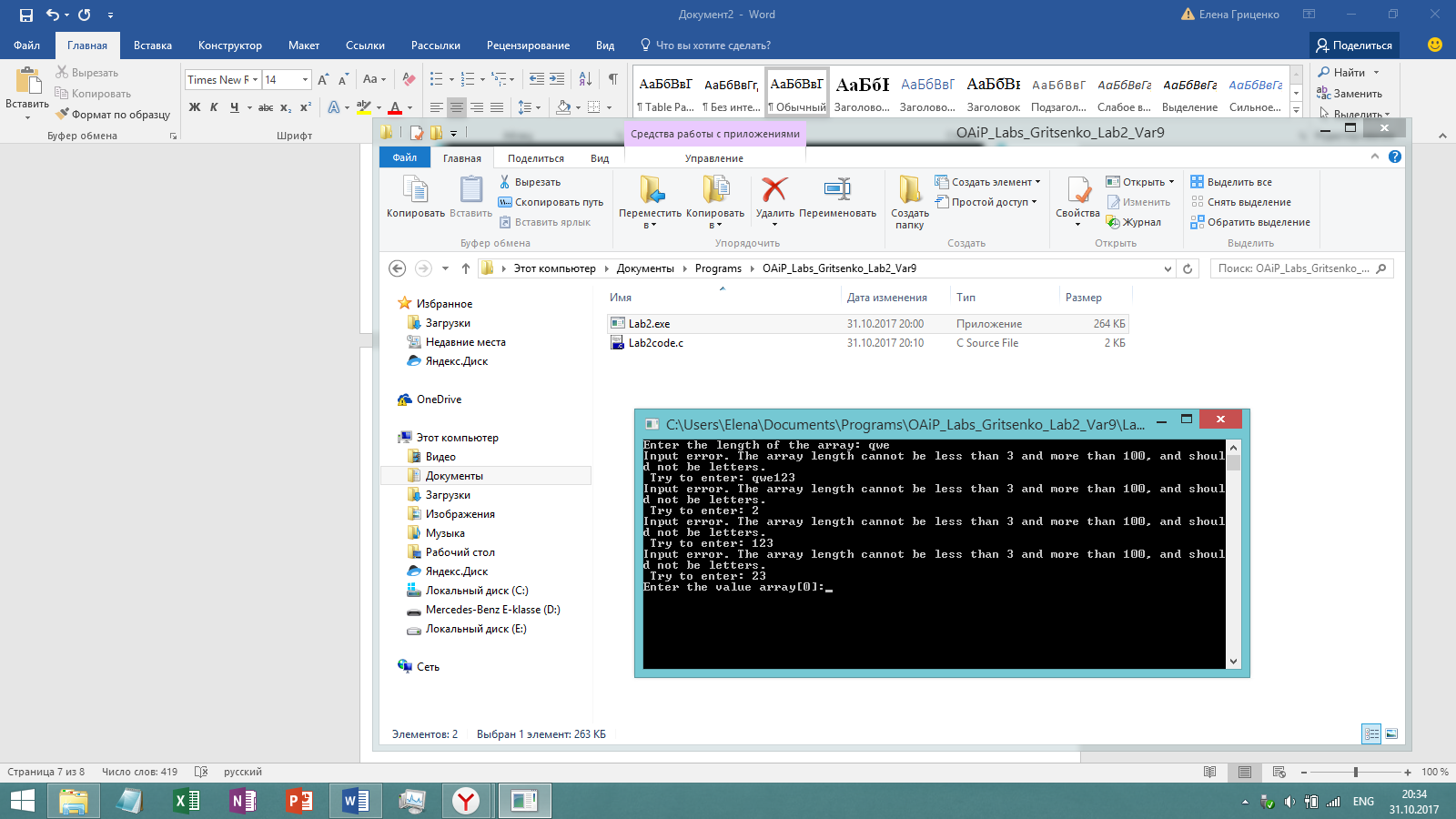


Рис. 4

Ввод не корректных данных

Контрольный пример

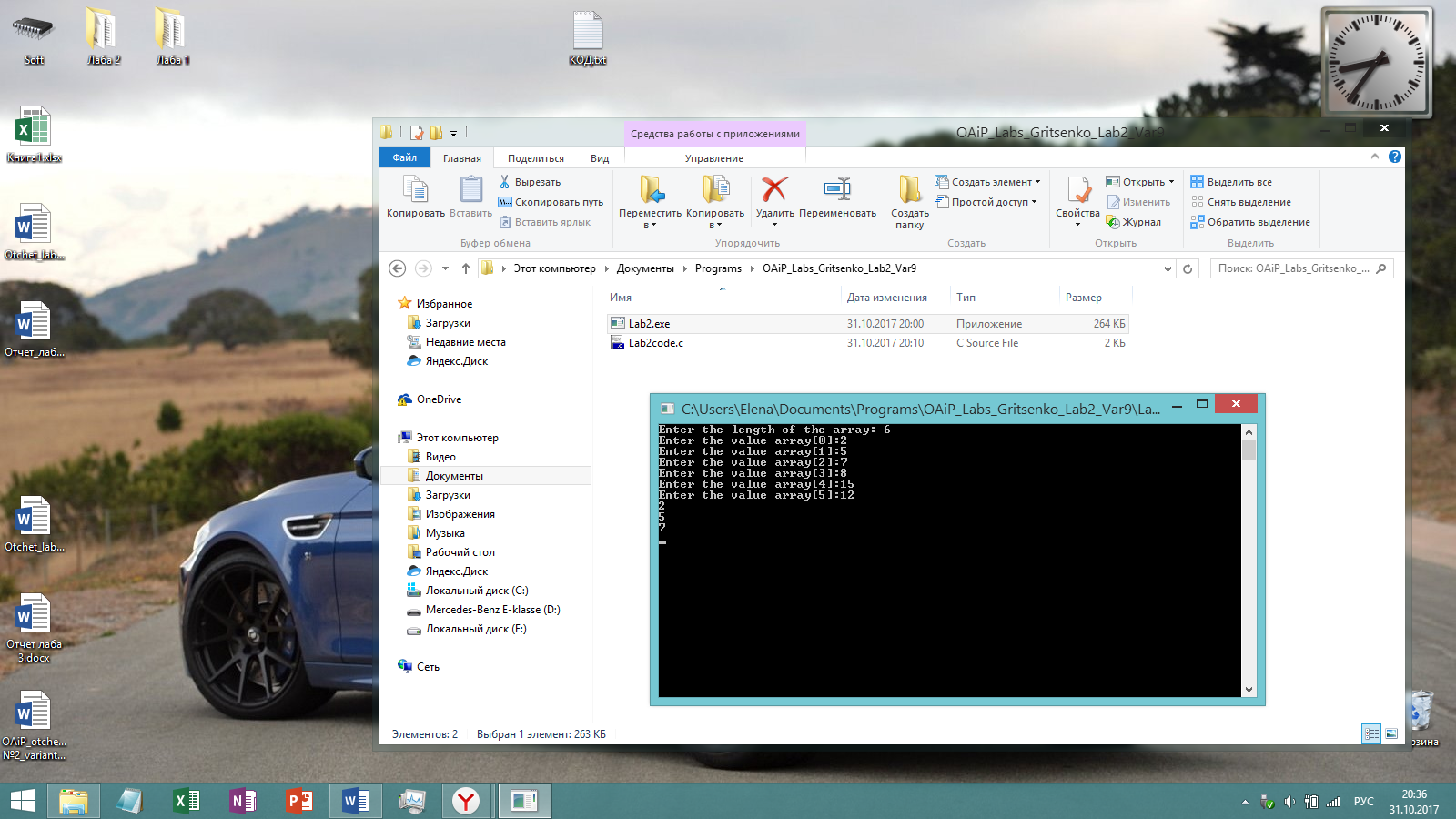


Рис. 5

Контрольный пример

# Вывод

В результате выполнения лабораторной №2 (Вариант 9) 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, а программа выдает только простые числа. В случае их отсутствия программа выведет сообщение «No primes numbers».