

Правительство Российской Федерации

**Федеральное государственное автономное образовательное учреждение
высшего профессионального образования
"Национальный исследовательский университет
"Высшая школа экономики"**

Московский институт электроники и математики Национального
исследовательского университета "Высшая школа экономики"

Департамент прикладной математики

ОТЧЕТ

По лабораторной работе №4

ОБРАБОТКА ДАННЫХ В ОДНОМЕРНОМ МАССИВЕ

По курсу «Алгоритмизация и программирование»

| ФИО студента | Номер группы | Дата |
|--------------------------|--------------|------|
| Андреев Евгений Игоревич | БПМ-191 | |

Москва – 2019 г.

ЗАДАНИЕ (вариант №1)

Числовой массив В (тип массива указан в формулировке второго задания) содержит k элементов. Элементы массива и пороговые значения X , Y вводятся с клавиатуры. Написать подпрограммы создания массива и вывода его на экран. В первом задании требуется написать функцию нахождения соответствующего варианту максимального/минимального значения, а во втором – среднего арифметического указанных в условии элементов («между» понимать строго – не включая найденные позиции).

Оба задания реализовать в одной программе.

| Задание 1 | Задание 2 |
|---|--|
| $\max(b_1 , \dots, b_k)$ для $ b_i < Y$ | Среднее арифметическое элементов, расположенных между первым и последним положительными элементами. Массив вещественный. |

РЕШЕНИЕ

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>                                // библиотека для завершения программы
в функции
#define K 100                                     // введение константы

double max(double* array, int size, int Y)         // функция для рассчитывания
max
{
    int i;                                         // объявление переменной типа int
    double dmax = 0;                              // объявление переменной типа double
    for(i = 0; i < size; i++)                     // цикл для переменной i от 1 до size с
шагом
    {
        if(fabs(array[i]) < Y && fabs(array[i]) > dmax) // условие для нахождения max
        dmax = fabs(array[i]);                    // обновление max
    }
    return dmax;                                  // возвращение max
}

double mean_(double* array, int size)             // функция для рассчитывания
среднего
{
    int first_i = 0, second_i = 0, i, q;          // объявление переменной типа int
    double sum = 0, mean;                         // объявление переменных типа double
    for(i = 0; i <= size; i++)                     // цикл для определения first_i
                                                    // (индекса первого положительного элемента)
                                                    // для переменной i от 1 до size с шагом 1
    {
```

```

        if((array[i] > 0) && (first_i == 0))                // условие для нахождения first_i
        {
            first_i = i;
            break;                                          // выход из цикла, когда значение нашлось
        }
    }
    for(i = size - 1; i >= 0; i--)                          // цикл для определения second_i
// (индекса последнего положительного
элемента)
// для переменной i от size до 1 с шагом 1
    {
        if((array[i] > 0) && (second_i == 0))                // условие для нахождения second_i
        {
            second_i = i;
            break;                                          // выход из цикла, когда значение нашлось
        }
    }
    for(i = first_i + 1; i < second_i; i++)                  // цикл для нахождения суммы
элементов массива
    {
        sum += array[i];                                  // суммирование элементов массива
    }
    q = second_i - first_i - 1;                             // нахождение количества искомых
элементов
    if (sum == 0) return mean;
    else if(q != 0)
    {
        mean = sum / q;                                    // рассчитывание среднего
        return mean;                                       // возвращение среднего
    }
    else
    {
        printf("Не обнаружено элементов между первым и последним положительными
элементами.");
        exit(0);                                           // завершение программы при отсутствии
искомых элементов
    }
}

void print_array(double* array, int size)                  // функция для считывания массива
{
    int i;                                                  // объявление переменных типа int
    for(i = 0; i < size; i++)                              // цикл для считывания массива
    {
        printf("Введите %d-й элемент: ", i + 1);          // вывод номера элемента
        // который нужно считать
        if(scanf("%lf", &array[i]) == 0 || getchar( ) != '\n') // считывание элементов массива
        {
            printf("Ошибка! Элементы массива - вещественные числа."); // вывод ошибки
            exit(0);                                         // завершение программы
        }
    }
}

```


ТЕСТЫ

Тест № 1

Проверка на защиту ввода:

```
Введите размер массива (>1, < 100):1
Введите размер массива (>1, < 100):100
Введите размер массива (>1, < 100):2п
Ошибка! Переменная k – натуральное число на (1, 100).
```

```
Введите размер массива (>1, < 100):4
Введите y: 2п
Ошибка! Переменная y – вещественное число.
```

```
Введите размер массива (>1, < 100):6
Введите y: -1
Ошибка! Переменная y – вещественное положительное число.
```

```
Введите размер массива (>1, < 100):5
Введите y: 10
Введите 1-й элемент: 2
Введите 2-й элемент: 3
Введите 3-й элемент: 7выф8
Ошибка! Элементы массива – вещественные числа.
```

Тест № 2

Случай, когда максимума не нашлось:

```
Введите размер массива (>1, < 100):5
Введите y: 1
Введите 1-й элемент: 2
Введите 2-й элемент: 3
Введите 3-й элемент: 4
Введите 4-й элемент: 5
Введите 5-й элемент: 6
Максимума не существует.
mean = 4.000000
```

Тест № 3

Проверка максимума с отрицательными значениями:

```
Введите размер массива (>1, < 100):5
Введите y: 9
Введите 1-й элемент: -8
Введите 2-й элемент: 7
Введите 3-й элемент: -7
Введите 4-й элемент: 6
Введите 5-й элемент: 5
max = 8.000000
mean = -0.500000
```

Проверка максимума с вещественными значениями:

```
Введите размер массива (>1, < 100):5
Введите y: 10
Введите 1-й элемент: 9.8
Введите 2-й элемент: 9
Введите 3-й элемент: -9.88
Введите 4-й элемент: 8
Введите 5-й элемент: 7
max = 9.880000
mean = 2.373333
```

Тест № 4

Проверка среднего с отрицательными числами:

```
Введите размер массива (>1, < 100):6
Введите y: 10
Введите 1-й элемент: 1
Введите 2-й элемент: -2
Введите 3-й элемент: -3
Введите 4-й элемент: 5
Введите 5-й элемент: 10
Введите 6-й элемент: 2
max = 5.000000
mean = 2.500000
```

Проверка среднего с вещественными числами:

```
Введите размер массива (>1, < 100):5
Введите y: 10
Введите 1-й элемент: 1
Введите 2-й элемент: 2.3
Введите 3-й элемент: -1.2
Введите 4-й элемент: 5.21
Введите 5-й элемент: 3
max = 5.210000
mean = 2.103333
```

Тест № 5

Проверка среднего, когда первый и последний элемента массива положительны:

```
Введите y: 10
Введите 1-й элемент: 1
Введите 2-й элемент: 2
Введите 3-й элемент: 3
Введите 4-й элемент: 4
Введите 5-й элемент: 5
Введите 6-й элемент: 6
Введите 7-й элемент: 7
Введите 8-й элемент: 8
max = 8.000000
mean = 4.500000
```

Проверка среднего, когда первые и последние элементы не положительны:

```
Введите размер массива (>1, < 100):10
Введите y: 10
Введите 1-й элемент: -1
Введите 2-й элемент: -2
Введите 3-й элемент: 3
Введите 4-й элемент: 4
Введите 5-й элемент: 5
Введите 6-й элемент: 6
Введите 7-й элемент: 7
Введите 8-й элемент: 8
Введите 9-й элемент: -9
Введите 10-й элемент: -10
max = 9.000000
mean = 5.500000
```

Тест № 6

Проверка среднего, когда нет элементов между первым и последним положительными элементами:

```
Введите размер массива (>1, < 100):6
```

```
Введите y: 10
```

```
Введите 1-й элемент: -1
```

```
Введите 2-й элемент: -2
```

```
Введите 3-й элемент: 1
```

```
Введите 4-й элемент: 2
```

```
Введите 5-й элемент: -3
```

```
Введите 6-й элемент: -4
```

```
max = 4.000000
```

```
Не обнаружено элементов между первым и последним положительными элементами.
```