

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
КАФЕДРА МО ЭВМ

ОТЧЕТ
по лабораторной работе №1
по дисциплине «Программирование»
Тема: Управляющие конструкции языка C

Студент гр. 1304

Ефремов А.А.

Преподаватель

Чайка К.В.

Санкт-Петербург

2021

Цель работы.

Научиться работать с управляющими конструкциям языка C, реализовать программу, выделив подзадачи в отдельные функции, использовать условия, циклы, оператор switch.

Задание.

Реализовать программу, на вход которой подается одно из значений 0, 1, 2, 3 и массив целых чисел размера не больше 20. Числа разделены пробелами. Строка заканчивается символом перевода строки.

В зависимости от значения, функция должна выводить следующее:

0 : Индекс первого отрицательного элемента. (index_first_negative)

1 : Индекс последнего отрицательного элемента. (index_last_negative)

2 : Найти произведение элементов массива, расположенных от первого отрицательного элемента (включая элемент) и до последнего отрицательного (не включая элемент). (multi_between_negative)

3 : Найти произведение элементов массива, расположенных до первого отрицательного элемента (не включая элемент) и после последнего отрицательного (включая элемент). (multi_before_and_after_negative)

Иначе необходимо вывести строку "Данные некорректны".

Выполнение работы.

fv - первое число, принимаемое на вход и определяющее последующую исполняемую функцию

err - состояние проверки на ошибки

ans - ответ, искомое значение

counter - счетчик количества чисел, принимаемых на вход в программу

arr - массив, принимаемый на вход в программу

ch - символ, считываемый после каждого числа из массива

i - шаг по массиву

res - результат произведения чисел из массива, удовлетворяющих условиям задачи согласно fv

index_f - индекс первого отрицательного числа в массиве

index_l - индекс последнего отрицательного числа в массиве

Разработанный программный код смотри в приложении А.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	0 -5 -3 -5 -8 3 -9 -3	0	-

Выводы.

Была написана программа, выполняющая различные операции над массивом чисел в зависимости от значения первого введенного числа. В процессе написания программы были использованы условия, циклы, оператор switch, а подзадачи выделены в отдельные функции.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: lab1.c

```
#include <stdio.h>

int index_first_negative(int arr[], int N)
{
    for (int i = 0; i < N; i++)
        if (arr[i] < 0)
            return i;
    return -1;
}

int index_last_negative(int arr[], int N)
{
    int index_l = -1;
    for (int i = 0; i < N; i++)
        if (arr[i] < 0)
            index_l = i;
    return index_l;
}

int multi_between_negative(int arr[], int N, int *err)
{
    int res = 1;
    int index_f = index_first_negative(arr, N);
    int index_l = index_last_negative(arr, N);
    if ((index_f == index_l) || (index_f < 0) || (index_l < 0))
    {
        *err = 1;
        return 0;
    }
    for (int i = index_f; i < index_l; i++)
        res *= arr[i];
    return res;
}
```

```

}

int multi_before_and_after_negative(int arr[], int N, int *err)
{
    int res = 1;
    int index_f = index_first_negative(arr, N);
    int index_l = index_last_negative(arr, N);
    if ((index_f == index_l) || (index_f < 0) || (index_l < 0))
    {
        *err = 1;
        return 0;
    }
    for (int i = 0; i < index_f; i++)
        res *= arr[i];
    for (int i = index_l; i < N; i++)
        res *= arr[i];
    return res;
}

int main()
{
    int fv, err = 0, ans = 0, counter = 0;
    int arr[20] = { 0 };
    char ch;
    scanf("%d", &fv);
    for (int i = 0; i < 20; i++)
    {
        scanf("%d%c", &arr[i], &ch);
        counter++;
        if (ch == '\n')
            break;
    }
    switch (fv)
    {
    case 0:
        ans = index_first_negative(arr, counter);

```

```

        if (ans != -1)
            printf("%d", ans);
        else
            printf("Данные некорректны");
        break;
case 1:
    ans = index_last_negative(arr, counter);
    if (ans != -1)
        printf("%d", ans);
    else
        printf("Данные некорректны");
    break;
case 2:
    ans = multi_between_negative(arr, counter, &err);
    if (err)
        printf("Данные некорректны");
    else
        printf("%d", ans);
    break;
case 3:
    ans = multi_before_and_after_negative(arr, counter, &err);
    if (err)
        printf("Данные некорректны");
    else
        printf("%d", ans);
    break;
default:
    printf("Данные некорректны");
    break;
}
}

```