

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: Условия, циклы, оператор switch

Студентка гр. 0382

Здобнова К.Д.

Преподаватель

Жангиров Т.Р.

Санкт-Петербург

2020

Цель работы.

Научиться работать с условным оператором (if), оператором switch, с циклом for; научиться инициализировать массивы, передавать их в функцию.

Задание.

Вариант 5.

Напишите программу, выделив каждую подзадачу в отдельную функцию.

Реализуйте программу, на вход которой подается одно из значений 0, 1, 2, 3 и массив целых чисел размера не больше 100. Числа разделены пробелами. Строка заканчивается символом перевода строки.

В зависимости от значения, функция должна выводить следующее:

0 : максимальное по модулю число в массиве. (abs_max)

1 : минимальное по модулю число в массиве. (abs_min)

2 : разницу между максимальным по модулю и минимальным по модулю элементом. (diff)

3 : сумму элементов массива, расположенных после максимального по модулю элемента (включая этот элемент). (sum)

иначе необходимо вывести строку "Данные некорректны".

Выполнение работы.

Первое входное число записывается в *command* – переменная, для хранения команды, остальные числа записываются в массив *array*, его размер считается в цикле и записывается в переменную *array_size*. В зависимости от значения переменной *command*, выполняются следующие функции и печатается на экран их результат:

«0»: *int abs_max(int array_size, int array[])* – функция, принимающая на вход целочисленный массив и его размер и возвращающая индекс максимального по модулю числа в массиве;

«1»: *int abs_min(int array_size, int array[])* – функция, принимающая на вход целочисленный массив и его размер и возвращающая индекс минимального по модулю числа в массиве;

«2»: *int diff(int array_size, int array[])* – функция, принимающая на вход целочисленный массив и его размер и возвращающая разницу между максимальным по модулю и минимальным по модулю элементом в массиве;

«3»: *int sum(int array_size, int array[])* – функция, принимающая на вход целочисленный массив и его размер и возвращающая сумму элементов массива, расположенных после максимального по модулю элемента (включая этот элемент);

Разработанный программный код см. в приложении А.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	0 -28 26 30 22 -13 -28 3 -12 8 10 -19 -26 11 -6 -18 -3 -2 - 26 18 8 -19 -17 -11 -12 -23 19 -16 -11 9	30	Программа напечатала на экран максимальный элемент по модулю в массиве
2.	1 764 -345 87 45 -23 43 -123	-23	Программа напечатала на экран минимальный элемент по модулю в массиве
3.	2 345 56 34 56 78 2 -3 54 87 -121 234	343	Программа напечатала на экран разность между максимальным по модулю и минимальным по модулю элементом
4	3 20 23 22 10 -21 7 8 -9	40	Программа напечатала на экран сумму элементов, начиная с максимального, в массиве
5	5 65 8 2 3 44 -45 34	Данные некорректны	Т.к. в входных данных первое число должно быть от 0 до 3, программа не смогла распознать команду
6	0	Данные некорректны	Не были введены элементы массива

Выводы.

Были изучены основные конструкции языка C, такие как: оператор условия, оператор *swift* и работа цикла *for*. Было исследовано инициализация массивов и работа с ними в функциях.

Разработана программа, выполняющая считывание с клавиатуры исходных данных и команды пользователя. Для обработки команд пользователя использовался оператор *switch* и *if-else*. Для работы с массивами в функции использовались условный оператор *if* и функция нахождения модуля числа *abs()*.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: main.cpp

```
#include <stdio.h>
#include <stdlib.h>

int abs_max(int array_size, int array[]){
    int max_index = 0;
    for (int i = 1; i < array_size; i++){
        if (abs(array[i]) > abs(array[max_index])){
            max_index = i;
        }
    }
    return max_index;
}

int abs_min(int array_size, int array[]){
    int min_index = 0;
    for (int i = 1; i < array_size; i++){
        if (abs(array[i]) < abs(array[min_index])){
            min_index = i;
        }
    }
    return min_index;
}

int diff(int array_size, int array[]){
    return (array[abs_max(array_size, array)] -
array[abs_min(array_size, array)]);
}

int sum(int array_size, int array[]){
    int sum = 0;
    for (int i = abs_max(array_size, array); i < array_size; i++){
        sum += array[i];
    }
    return sum;
}

int main()
{
    int array[101];
    char character;
    int command, array_size = 0;
    scanf("%d%c", &command, &character);
    while (array_size <= 100 && character == ' '){
        scanf("%d%c", &array[array_size], &character);
        array_size++;
    }
    if (array_size == 0)
        printf("Данные некорректны\n");
    else {
        switch (command){
            case 0:
                printf("%d\n", array[abs_max(array_size, array)]);
                break;
        }
    }
}
```

```

    case 1:
        printf("%d\n", array[abs_min(array_size, array)]);
        break;
    case 2:
        printf("%d\n", diff(array_size, array));
        break;
    case 3:
        printf("%d\n", sum(array_size, array));
        break;
    default:
        printf("Данные некорректны\n");
        break;
}
}
return 0;
}

```