**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. 2383 |  | Анищенко А.И. |
| Преподаватель |  | Гаврилов А.В. |

Санкт-Петербург

2022

## Цель работы.

На практике научиться пользоваться условными операторами, операторами циклов, операторами выбора, разобраться с работой массивов.

## Задание.

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 больше100. Числа разделены пробелами. Строка заканчивается символом перевода строки.

В зависимости от значения, функция должна выводить следующее:

0 : индекс первого нулевого элемента.

1 : индекс последнего нулевого элемента.

2 : Найти сумму модулей элементов массива, расположенных от первого нулевого элемента и до последнего.

3 : Найти сумму модулей элементов массива, расположенных до первого нулевого элемента и после последнего.

иначе необходимо вывести строку "Данные некорректны".

## Выполнение работы.

Использованы заголовочные файлы стандартной библиотеки языка Си:

<stdout.h> - для ввода-вывода, <stdlib.h> - для использования функции нахождения модуля числа.

Функция index\_first\_zero, на вход которой подаётся массив и количество элементов в нём, с помощью оператора цикла for проходится по массиву с конца, возвращая индекс первого 0.

Функция index\_last\_zero, на вход которой подаётся массив и количество элементов в нём, с помощью оператора цикла for проходится по массиву с начала, возвращая индекс последнего 0.

Функция sum\_between, на вход которой подаётся массив и количество элементов в нём, с помощью оператора цикла for проходится по массиву, с помощью условного оператора if ищет элементы массива, находящиеся между первым и последним 0, и возвращает сумму модулей этих элементов, записанную в переменную sum.

Функция sum\_bef\_and\_aft, на вход которой подаётся массив и количество элементов в нём, с помощью оператора цикла for проходится по массиву, с помощью условного оператора if ищет элементы массива, находящиеся до первого и после последнего 0, и возвращает сумму модулей этих элементов, записанную в переменную sum12.

В функции main обозначена переменная vvod, отвечающая за поданную на вход команду. Оператор множественного выбора switch case выводит результат в зависимости от полученной команды.

Разработанный программный код и результаты тестирования см. в приложении А и приложении Б.

## Выводы.

Была изучена работа операторов цикла for и do while, условного оператора if и оператора сложного выбора switch case. Также исследована работа с массивами.

В результате разработана программа, которая в зависимости от поданного значения выводит результат заданных команд. Чтобы работа программы была более понятной, каждая подзадача была выделена в отдельную функцию.

# Приложение А

**Название файла: lab1**

#include <stdio.h>

#include <stdlib.h>

#define N 100

int index\_first\_zero(int \*mas, int n){

int first\_zero=0;

for (int i=n-1; i > 0 ; i--){

if (mas[i]==0){

first\_zero=i;

}

}

return first\_zero;

}

int index\_last\_zero(int \*mas, int n){

int last\_zero;

for(int i=0; i < n; i++){

if (mas[i]==0){

last\_zero=i;

}

}

return last\_zero;

}

int sum\_between(int \*mas, int n){

int sum=0;

for(int i=0; i < n; i++){

if(i > index\_first\_zero(mas, n) && i < index\_last\_zero(mas, i)){

sum+=abs(mas[i]);

}

}

return sum;

}

int sum\_bef\_and\_aft(int \*mas, int n){

int sum12=0;

int sum1=0;

int sum2=0;

for(int i=0; i < n; i++){

if (i < index\_first\_zero(mas, n)){

sum1+=abs(mas[i]);

}

if (i > index\_last\_zero(mas, n)){

sum2+=abs(mas[i]);

}

}

sum12 = sum1 + sum2;

return sum12;

}

int main(){

int mass[N];

int size = 0;

int vvod;

char s;

scanf("%d\n", &vvod);

do{

scanf("%d%c", &mass[size++], &s);

}while(s != '\n' && size < N);

switch(vvod){

case 0:

{

printf("%d\n", index\_first\_zero(mass, size));

break;

}

case 1:

{

printf("%d\n", index\_last\_zero(mass, size));

break;

}

case 2:

{

printf("%d\n", sum\_between(mass, size));

break;

}

case 3:

{

printf("%d\n", sum\_bef\_and\_aft(mass, size));

break;

}

default:

{

printf("Данные некорректны\n");

break;

}

}

}

# Приложение Б Тестирование

Таблица 1 - Примеры тестовых случаев

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Входные данные | Выходные данные | Комментарии |
|  | 0 -21 10 0 -23 -7 -15 -14  8 -9 10 -13 -14 -27 0 -7 12 -18 | 2 | Выведен индекс первого 0 |
|  | 1 -21 10 0 -23 -7 -15 -14  8 -9 10 -13 -14 -27 0 -7 12 -18 | 17 | Выведен индекс последнего 0 |
|  | 2 -21 10 0 -23 -7 -15 -14  8 -9 10 -13 -14 -27 0 -7 12 -18 | 177 | Выведена сумма модулей элементов между первым и последним 0 |
| 4. | 3 -21 10 0 -23 -7 -15 -14  8 -9 10 -13 -14 -27 0 -7 12 -18 | 31 | Выведена сумма элементов до первого и после последнего 0 |