

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

Студент гр. 8303

Гришин. К. И.

Преподаватель

Чайка К. В.

Санкт-Петербург

2018

Цель работы

Изучить условия, циклы, оператор switch в языке C на примере программы, которая в зависимости от запрошенной функции обрабатывает массив целых чисел тем или иным образом.

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 больше 100. Числа разделены пробелами.

Строка заканчивается символом перевода строки.

В зависимости от значения, функция должна выводить следующее:

```
0 : индекс первого нулевого элемента. (index_first_zero)
1 : индекс последнего нулевого элемента.
  (index_last_zero)
2 : Найти сумму модулей элементов массива, расположенных
  от первого нулевого элемента и до последнего.
  (sum_between)
3 : Найти сумму модулей элементов массива, расположенных
  до первого нулевого элемента и после последнего.
  (sum_before_and_after)
иначе необходимо вывести строку "Данные некорректны".
```

Основные теоретические положения

Основным типом данных, с которым работает программа — является массив целых чисел.

Условный оператор «If» будет определять какую функцию, в зависимости от запроса, нужно вызвать.

Обработка массива будет происходить путем прохождения по массиву циклом «For» и проверкой условия с помощью оператора «If».

Ход выполнения работы

На вход подается ряд чисел, ввод оканчивается символом переноса строки. За окончание ввода отвечает переменная «finish» типа char.

Введенные числа записываются в массив «nums» типа int.

После определения номера функции программа решает, какую операцию применить путем цепочки условных операторов «If».

Каждому числу от 0 до 3 соответствует своя функция: index_first_zero, index_last_zero, sum_between, sum_before_and_after соответственно.

Функция index_first_zero инкрементирует переменную-счетчик, изначально равную нулю, пока не дойдет от начала массива до элемента массива равного нулю, получая тем самым индекс первого нулевого элемента.

Функция index_last_zero декрементирует переменную счетчик, изначально равную индексу последнего введенного элемента массива, пока не дойдет от индекса последнего введенного числа до элемента массива равного нулю, получая тем самым индекс последнего нулевого элемента.

Функция sum_between изначально находит значения функций index_first_zero и index_last_zero. Далее с помощью цикла «For» от первого нулевого элемента массива и до последнего нулевого элемента массива суммируются модули (используя функцию abs) элементов данного промежутка и возвращается значение суммы.

Функция sum_before_and_after также сначала находит значения функций index_first_zero и index_last_zero. Далее дважды используя цикл «For» находятся сумма модулей элементов до первого нулевого элемента (before) и сумма модулей после последнего нулевого элемента (after) затем возвращается значение before+after.

Вывод

В ходе написания работы были изучены цикл «For», условный оператор «If» и «Switch», способ инициализации и описания функции, а также различные алгоритмы обработки массива, описанные в ходе выполнения работы.

ПРИЛОЖЕНИЕ

КОД ПРОГРАММЫ

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

int index_first_zero(int *arr){
    int index = 0;
    while (arr[index] != 0)
        index++;
    return index;
}

int index_last_zero(int *arr, int size){
    int index = size-1;
    while (arr[index] != 0)
        index--;
    return index;
}

int sum_between(int *arr, int size){
    int first = index_first_zero(arr);
    int last = index_last_zero(arr, size);
    int sum = 0;
    for (size_t i = first+1; i < last; i++)
        sum += abs(arr[i]);
    return sum;
}

int sum_before_and_after(int *arr, int size){
    int first = index_first_zero(arr);
    int last = index_last_zero(arr, size);
    int before = 0;
    int after = 0;
    for (size_t i = 0; i < first; i++)
        before += abs(arr[i]);
    for (size_t i = last+1; i < size; i++)
        after += abs(arr[i]);
    return before+after;
}

int main(){
    char finish;
    int nums[100];
    int ind = 0;
    size_t op;
    scanf("%hd", &op);
    while (finish != '\n'){
        scanf("%d%c", &nums[ind], &finish);
        ind++;
    }
    switch (op){
        case 0:
            printf("%d\n", index_first_zero(nums));
            break;
        case 1:
            printf("%d\n", index_last_zero(nums, ind));
            break;
        case 2:
            printf("%d\n", sum_between(nums, ind));
            break;
        case 3:
            printf("%d\n", sum_before_and_after(nums, ind));
            break;
        default:
            printf("Данные некорректны\n");
            break;
    }
    return 0;
}
```