**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра МОЭВМ**

отчет

**по лабораторной работе №2**

**по дисциплине «Программирование»**

Тема: Создание программы с использованием структур switch, if-else, for, **функций стандартной библиотеки для ввода-вывода.**

|  |  |  |
| --- | --- | --- |
| Студент гр. 6304 |  | Зыль С. Е. |
| Преподаватель |  | Кринкин К.В. |

Санкт-Петербург

2016

**Цель:** Создать программу с использованием структур switch, if-else, for, функций стандартной библиотеки для ввода-вывода.

**Задание:**

Напишите функцию-меню, на вход которой может подаваться число n, одно из значений 0,1,2,3 и массив целых чисел размера n.

В зависимости от аргумента функция должна выводить следующее:

0: максимальное **число** в массиве

1: минимальное **число** в массиве

2: **сумму** все четных элементов

3: **число** повторов нулевого элемента в массиве (не включая в это количество сам элемент)

**Иначе** вывести строку ‘Данные некорректны’

**Содержание:**

*void menu (int ch, int arr[], int n) \\* Объявление функции меню с параметрами и соответствующими аргументами \\

*if (n<=0) \\* Проверяем положительна ли требуемая длинна массива

*{*

*printf(“Данные некорректны”);*

*return;*

*}*

*else*

*{*

*switch (ch) \\* Оператор множественного выбора для переменной ch={0,1,2,3} *\\*

*{*

*case 0: printf("%d\n",get\_max(arr,n)); \\* В случае 0 печатаем результат функции поиска максимального элемента массива *\\*

*break;*

*case 1: printf("%d\n",get\_min(arr,n)); \\* В случае 1 печатаем результат функции поиска минимального элемента массива *\\*

*break;*

*case 2: printf("%d\n",get\_sum(arr,n)); \\* В случае 2 печатаем результат функции подсчета суммы четных элементов массива *\\*

*break;*

*case 3: printf("%d\n",get\_count\_first\_el(arr,n)); \\* В случае 3 печатаем результат функции подсчета повтора нулевого элемента массива*\\*

*break;*

*default: printf ("Данные некорректны\n"); \\* Иначе выводим сообщение о некорректности данных*\\*

*}*

*}*

*int get\_max(int arr[], int n)\\* Объявление функции поиска максимального элемента массива *\\*

*{*

*int max=arr[0];*

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

*{*

*if(max<arr[i]) \\* Ищем максимальный элемент*\\*

*max=arr[i];*

*}*

*return max; \\* Возвращаем максимальный элемент в вызывающую функцию*\\*

*}*

*int get\_min(int arr[], int n) \\* Объявление функции поиска минимального элемента массива

*\\*

*{*

*int min=arr[0];*

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

*{*

*if(min>arr[i]) \\* Ищем минимальный элемент*\\*

*min=arr[i];*

*}*

*return min; \\* Возвращаем минимальный элемент в вызывающую функцию*\\*

*}*

*int get\_sum(int arr[], int n) \\* Объявление функции подсчета суммы четных элементов*\\*

*{*

*int Sum=0;*

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

*{*

*if (arr[i]%2==0) \\* Проверяем элементы на четность и наращиваем их сумму\*\*

*Sum+=arr[i];*

*}*

*return Sum; \\* Возвращаем сумму в вызывающую функцию*\\*

*}*

*int get\_count\_first\_el(int arr[], int n)\\* Объявление функции подсчета повторов нулевого элемента\\

*{*

*int Sum=0;*

*for (int i=1;i<n;i++)*

*{*

*if (arr[i]==arr[0]) \\*Считаем повторение нулевого элемента*\\*

*Sum++;*

*}*

*return Sum; \\*Возвращаем результат в вызывающую функцию***\\***

*}*

*#include <stdio.h> // подключение стандартных*

*#include <stdlib.h> библиотек \\*

*int get\_max(int arr[], int n) )\\* Объявление функции поиска максимального элемента массива *\\*

*{*

*int i;* \\ инициализация

*int max; п*еременных\\

*max=arr[0];*

*for (i=0; i<n; ++i*)\\ реализация цикла for\\{

*if(arr[i]>max)* \\ поиск максимального значения\\

*max=arr[i];*

*}*

*return max;* \\ возвращение функции значения max\\

*}*

*int get\_min(int arr[], int n) )\\* Объявление функции поиска минимального элемента массива *\\*

*{*

*int i;* \\ инициализация

*int min;* переменных\\

*min=arr[0];*

*for (i=0; i<n; ++i)*[\\реализация](file:///\\реализация) цикла for\*\ {*

*if(arr[i]<min) \\* поиск минимального значения \\

*min=arr[i];*

*}*

*return min;* \\ возвращение функции значения min\\

*}*

*int get\_sum(int arr[], int n)* \\Объявление функции подсчета суммы четных элементов*\\{*

*int i; \\* инициализация

*int summ=0;* переменных\\

*for (i=0; i<n; ++i)*\\ реализация цикла for\\*{*

*if (arr[i]%2==0) \\* поиск четных элементов\\

*summ+=arr[i]; \\* подсчет суммы\\

*}*

*return summ; \\* возвращение функции значения summ\\

*}*

*int get\_count\_first\_el(int arr[], int n)\\* Объявление функции подсчета повторов нулевого элемента\\

*{*

*int i;* \\ инициализация

*int c=0; переменных\\*

*for (i=1; i<n; ++i)\\* реализация цикла for\\*{*

*if(arr[i]==arr[0])* \\ поиск значений нулевого элемента массива\\

*++c;*

*}*

*return c; \\* возвращение функции значения c\\

*}*

*void menu(int ch, int arr[], int n)* \\Объявление функции меню с параметрами и соответствующими аргументами \\

*{*

*switch ( ch )\\ оператор выбора значений ch={0; 1; 2; 3}\\ {*

*case 0:*

*printf("%d\n", get\_max(arr, n)); \\* В случае 0 печатается результат функции поиска максимального элемента массива\\

*return;* \\выход из оператора switch\\

*case 1:*

*printf("%d\n", get\_min(arr, n)); \\* В случае 1 печатается результат функции поиска минимального значения\\

*return;* \\выход из оператора switch\\

*case 2:*

*printf("%d\n", get\_sum(arr, n)); \\* В случае 2 печатается результат функции подсчета суммы четных элементов массива *\\*

*return;* \\выход из оператора switch\\

*case 3:*

*printf("%d\n", get\_count\_first\_el(arr, n));* В случае 3 печатается результат функции подсчета повтора нулевого элемента массива*\\*

*return;* \\выход из оператора switch\\

*default:*

*printf("Данные некорректны"); \\ Иначе печатается: «Данные некорректны»\\*

*return;* \\выход из оператора switch\\

*}*

*}*

**Вывод:** Выполнив данную лабораторную работу, мы освоили и закрепили на практике написание программы с применением функций стандартной библиотеки для ввода-вывода, структур for, if-else, switch.