

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

**ОТЧЕТ**  
**по лабораторной работе №2**  
**по дисциплине «Программирование»**  
**Тема: «Условия, циклы, оператор switch»**

Студент гр. 7381

\_\_\_\_\_

Трушников А.П.

Преподаватель

\_\_\_\_\_

Берленко Т.А.

Санкт-Петербург

2017

## Цель работы.

Познакомиться с массивами; познакомиться с оператором выбора switch; познакомиться с циклами `for (;);`, `while ()`, `do while ()`; познакомиться с операторами `break` и `return`; в текущей директории создать проект с make-файлом.

Главная цель должна приводить к сборке проекта. Файл, который **реализует главную функцию**, должен называться `menu.c`; **исполняемый файл** - `menu`.

Определение каждой функции должно быть расположено в **отдельном файле**, название файлов указано в скобках около описания каждой функции.

Реализовать функцию-меню, 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 чётного элемента. (`index_first_even.c`)

1: индекс последнего нечётного элемента. (`index_last_odd.c`)

2: Найти сумму модулей элементов массива, расположенных от первого чётного элемента и до последнего нечётного, включая первый и не включая последний. (`sum_between_even_odd.c`)

3: Найти сумму модулей элементов массива, расположенных до первого чётного элемента (не включая элемент) и после последнего нечетного (включая элемент). (`sum_before_even_and_after_odd.c`)

Иначе необходимо вывести строку “Данные некорректны”.

## Основные теоретические положения.

Заголовочные файлы стандартной библиотеки языка C, необходимые для выполнения данной лабораторной работы:

1. Функция `<int main()>`. В этой функции заголовочный файл `<stdio.h>` (стандартный заголовочный файл ввода-вывода) подключает прототип функций `<int scanf(const char* format [, argument]...)>`, `<int printf (const char* format [,`

argument]...);», которые используются для ввода из потока ввода и вывода в поток вывода. А так же прототипы функций «`int getc ( FILE * filestream);`» и «`int ungetc( int character, FILE * filestream );`».

«`int getc ( FILE * filestream)`»:

### **Описание**

Функция возвращает символ из потока `filestream`, на который ссылается внутренний индикатор позиции файла. Внутренний индикатор позиции в файле, после срабатывания этой функции сдвигается на один символ, таким образом, теперь он указывает на следующий символ.

### **Параметры:**

- `filestream`

Указатель на объект типа `FILE`, который идентифицирует поток, для выполнения с ним различных операций.

### **Возвращаемое значение:**

Считанный символ возвращается в виде целого значения.

Если конец файла достигнут или в процессе чтения происходит ошибка, функция возвращает `EOF` и соответствующие индикаторы ошибки или конца файла устанавливаются. Вы можете использовать любую функцию `ferror` или `feof` чтобы определить, произошла ошибка или был достигнут конец файла.

«`int ungetc( int character, FILE * filestream )`»:

### **Описание:**

Функция `ungetc` возвращает только что прочитанный символ обратно в поток ввода `filestream`, через параметр `character`. Внутренний индикатор позиции файла уменьшается обратно, на предыдущее положение, так что этот символ возвращается при следующем вызове операции чтения для этого потока.

Параметр `character` может содержать любой символ, например, последний символ прочитанный из потока в предыдущей операции или любой другой. В обоих случаях, значение, полученное по следующей операции чтения является значением функции `ungetc`, независимо от символа `character`.

**Параметры:**

- character

Символ, возвращаемый обратно в поток. Символ передается, как значение типа int.

- filestream

Указатель на объект типа FILE, который идентифицирует входной поток.

**Возвращаемое значение:**

В случае успеха, возвращается целочисленное значение символа, который был перенесен обратно в поток. В противном случае, возвращается значение EOF, и поток остается неизменным.

2. В функциях «sum\_between\_even\_odd(int \*a, int len);» и «sum\_before\_even\_and\_after\_odd(int \*a, int len);». В них заголовочный файл «stdlib.h» подключает прототип функции «int abs(int n);».

«int abs(int n);»:

**Описание:**

Функция вычисляет абсолютную величину (модуль) значения, передаваемого в качестве аргумента через параметр n.

**Параметры:**

- n

Целое значение.

**Возвращаемое значение:**

Модуль числа n.

**Вывод:**

Познакомился с массивами в C: синтаксисом, использованием, расположением в памяти, и т. д. Познакомился с оператором выбора switch: синтаксисом, использованием, операторами case, break и default. Познакомился с циклами for (;;), while (), do while (): синтаксисом, использованием, операторами break для выхода из цикла.

