

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

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

Студент гр. 7381

Тарасенко Е.А.

Преподаватель

Берленко Т.А.

Санкт-Петербург

2017

Цель работы

Создать проект, задача которого будет заключаться в работе с введенным массивом (поиск индекса первого или последнего отрицательного элемента, нахождение произведения элементов, находящихся между первым и последним отрицательными, нахождение произведения элементов, находящихся до первого отрицательного и после последнего отрицательного).

Дополнительная информация для выполнения лабораторной работы

Требуется 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.

Требуется реализовать функцию-меню, на вход которой подается одно из значений 0, 1, 2, 3 и массив целых чисел **размера не больше 20**. Числа разделены пробелами. Строка заканчивается символом перевода строки.

В зависимости от **значения**, функция должна выводить следующее:

0 : индекс первого отрицательного элемента. (`index_first_negative.c`)

1 : индекс последнего отрицательного элемента. (`index_last_negative.c`)

2 : Найти произведение элементов массива, расположенных от первого отрицательного элемента (**включая элемент**) и до последнего отрицательного (**не включая элемент**). (`multi_between_negative.c`)

3 : Найти произведение элементов массива, расположенных до первого отрицательного элемента (**не включая элемент**) и после последнего отрицательного (**включая элемент**). (`multi_before_and_after_negative.c`)

иначе необходимо вывести строку **"Данные некорректны"**.

Вывод

В ходе данной работы были приобретены некоторые навыки работы с циклами в языке СИ, оператором множественного выбора «switch». Был создан проект, состоящий из десяти файлов, способный принимать на вход цифру (номер операции от 0 до 3) и массив из целых чисел. В зависимости от номера операции программа выводила индекс первого или последнего отрицательного элемента массива, произведение элементов, стоящих между первым и последним отрицательными или произведение элементов, стоящих до первого и после последнего отрицательных.

Множественное определение функций было предотвращено при помощи директивы «#ifndef»:

```
#ifndef INDEX_FIRST_NEGATIVE
#define INDEX_FIRST_NEGATIVE
    int index_first_negative(int a[], int length);
#endif
```

Приложение

Главная функция:

```
int main()
{
    int length = 0;

    char c;

    int a[100];

    int type_of_operation;

    scanf("%d", &type_of_operation);

    while(c!='\n')
    {
        scanf("%d%c", &a[length], &c);
```

```

        length++;
    }

    switch(type_of_operation)
    {
        case 0:
            printf("%d\n", index_first_negative(a, length));
            break;

        case 1:
            printf("%d\n", index_last_negative(a, length));
            break;

        case 2:
            printf("%d\n", multi_between_negative(a, length));
            break;

        case 3:
            printf("%d\n", multi_before_and_after_negative(a, length));
            break;

        default:
            printf("Data is not correct!\n");
            break;
    }

    return 0;
}

```

Функция поиска индекса первого отрицательного элемента:

```

int index_first_negative(int a[], int length)
{

```

```

int i, neg;

for (i=0; i<length; i++)
{
    if(a[i] < 0)
    {
        neg = i;

        return neg;

        break;
    }
}

```

Функция поиска индекса последнего отрицательного элемента:

```
int index_last_negative(int a[], int length)
```

```

{
    int i, neg;

    for (i=length-1; i>=0; i--)
    {
        if(a[i] < 0)
        {
            neg = i;

            return neg;

            break;
        }
    }
}

```

Функция поиска произведения элементов, стоящих между первым и последним отрицательными:

```
int multi_between_negative(int a[], int length)
{
    int i, first_neg, last_neg;
    int multi = 1;
    first_neg = index_first_negative(a, length);
    last_neg = index_last_negative(a, length);
    for (i=0; i<length; i++)
    {
        if((i >= first_neg) && (i < last_neg))
        {
            multi = multi * a[i];
        }
    }
    return multi;
}
```

Функция поиска произведения элементов, стоящих до первого и после последнего отрицательного:

```
int multi_before_and_after_negative(int a[], int length)
{
    int i, first_neg, last_neg;
    int multi = 1;
    first_neg = index_first_negative(a, length);
    last_neg = index_last_negative(a, length);
    for (i=0; i<length; i++)
    {
        if(((i>=0) && (i<first_neg)) || ((i>=last_neg) && (i<=length)))
        {
            multi = multi * a[i];
        }
    }
}
```

```
        return multi;
    }
}
```

Makefile:

```
all: menu.o index_first_negative.o index_last_negative.o multi_between_negative.o
multi_before_and_after_negative.o
```

```
gcc menu.o index_first_negative.o index_last_negative.o multi_between_negative.o
multi_before_and_after_negative.o -o menu
```

```
menu.o: menu.c index_first_negative.h index_last_negative.h multi_between_negative.h
multi_before_and_after_negative.h
```

```
gcc -c menu.c
```

```
index_first_negative.o: index_first_negative.c index_first_negative.h
```

```
gcc -c index_first_negative.c
```

```
index_last_negative.o: index_last_negative.c index_last_negative.h
```

```
gcc -c index_last_negative.c
```

```
multi_between_negative.o: multi_between_negative.c multi_between_negative.h
index_first_negative.h index_last_negative.h
```

```
gcc -c multi_between_negative.c
```

```
multi_before_and_after_negative.o: multi_before_and_after_negative.c
multi_before_and_after_negative.h index_first_negative.h index_last_negative.h
```

```
gcc -c multi_before_and_after_negative.c
```