

Правила оформления кода для лабораторных работ

Код должен содержать осмысленные, говорящие имена переменных и функций

Пример функции, которая считает, сколько раз символ letter встретится в строке string:

```
int letterCounter(char *string, char letter)
{
    int letterCount = 0;
    for(int i = 0; string[i]; i++)
        if(string[i] == letter)
            letterCount++;
    return letterCount;
}
```

Исключение: имена переменных в цикле со счётчиком

Пример использования цикла со счётчиком:

```
for(i = 0; i < array_size; i++){
    ... // тело цикла
}
```

Для считывания строк (в том числе с консоли) используйте функцию fgets. Запрещается использовать scanf и gets

Пример использования fgets для считывания с консоли:

```
int size = 100;
char string[size];
fgets(string, size, stdin);
```

Помимо функции fgets можно посимвольно считывать строку через getchar, однако вариант с fgets является более правильным.

Если есть гарантии по условию, что на вход будет подаваться какой-то **один конкретный тип данных кроме строки** (например, целочисленное число), то тогда можно использовать функцию scanf. Например, следующий код считывает число между двумя вертикальными чертами (например, |15|):

```
int x;
scanf("|%d|", &x);
```

Освобождайте выделенную динамическую память

Пример выделения и освобождения памяти:

```
char* buffer = (char*)malloc(sizeof(char) * buffer_size);  
...  
free(buffer);
```

From:

<http://se.moevm.info/> - **se.moevm.info**

Permanent link:

<http://se.moevm.info/doku.php/courses:programming:rules>



Last update: **2023/09/13 08:32**