

В заданиях используйте метод `assert` из заголовочного файла `cassert`. Это даст вам возможность проверить работоспособность кода. Если условие внутри `assert` `false`, то будет выведена ошибка. В противном случае если условие `true` - ошибки не будет.

### **Задание 1.**

Реализовать собственную функцию `int my_strlen(const char * s)`.

Функция должна возвращать длину переданной строки `s` (количество символов до `\0`).

Приведенный код ниже должен работать:

```
int my_strlen(const char * s) {
    /* Ваш код здесь */
}

#include <iostream>
#include <cassert>
using namespace std;

int main() {
    assert(strlen("hello") == 5);
    return 0;
}
```

### **Задание 2.**

Реализовать функцию `int my_strcmp(const char* str1, const char* str2)`.

Функция возвращает `true` в случае, если строки равны посимвольно.

Например: `my_strcmp("Hello", "Hello")` // `true`

`my_strcmp("Hi", "hi")` // `false`

Приведенный код ниже должен работать:

```
bool my_strcmp(const char * str1, const char * str2) {
    /* Ваш код здесь */
}

#include <iostream>
#include <cassert>
using namespace std;

int main() {
    assert(my_strcmp("hi", "hi"));
    assert(my_strcmp("Hello", "Hello"));
    return 0;
}
```

### Задание 3.

Реализовать функцию `int my_strcpy(char* dest, const char* source)`.

Функция посимвольно копирует строку `source` в строку `dest` (не забудьте, в конце должен быть `\0`) и возвращает `dest`.

Приведенный код ниже должен работать:

```
char * my_strcpy(char * dest, const char * source) {
    /* Ваш код здесь */
}

bool my_strcmp(const char * str1, const char * str2);

#include <iostream>
#include <cassert>
using namespace std;

int main() {
    char str[128];

    // копирование Hello в str
    my_strcpy(str, "Hello");

    // Проверка: str должна быть равна Hello
    assert(my_strcmp(str, "Hello"));

    return 0;
}
```

Задание 1 – Подсказка:

Используйте цикл с условием выхода `s[i] == \0`

Задание 2 – Подсказка:

Используйте сравнение длин с помощью уже написанной `my_strlen` (должны быть равны) и проход в цикле по первой строке сравнивая каждый её символ с таким же символом под тем же индексом, что и во второй строке.

Задание 3 – Подсказка:

Используйте цикл по всем элементам строки `source`, пока `source[i]` не будет равен `\0`. После этого не забудьте добавить `\0` к `dest`, чтобы строка `dest` считалась завершённой. Возвращайте `dest`.