



Курс:

ОСНОВЫ ПРОГРАММИРОВАНИЯ НА ЯЗЫКЕ C++

Тема: СТРОКИ

Задача 1

Дана строка **str**. Пользователь вводит символ **sym**. Программа создаёт новую строку, которая состоит из символа **sym**. Длина новой строки равна количеству совпадений введённого символа в изначальной строке **str**.

Пример:

```
str = "Hello world!"
```

```
sym = 'o'
```

```
newStr = "oo"
```

Подсказка

Решение

Задача 2

Создайте функцию, которая принимает слово и возвращает **true**, если оно является палиндромом. В противном случае возвращается **false**. Функция должна быть нерегистрозависимой.



Задача 3

Создайте функцию, которая принимает строку. Функция считает все пробелы, точки, запятые, восклицательные и вопросительные знаки, которые есть в переданной строке и возвращает их общее количество.

Задача 4

Создайте функцию, которая принимает строку и символ. Функция ищет последнее вхождение символа в переданную строку и возвращает подстроку, начинающуюся с этого символа и заканчивающуюся концом строки. Если символа в строке нет, то функция возвращает пустую строку.

Пример вызова функции:

```
string str = "Hello world!";  
char sym = 'o';  
cout << func(str, sym) << endl;  
// на экране orld!
```

Результат работы: файлы проекта или ссылка на репозиторий **Git**.



Подсказка к задаче 1

1. Как создать строковую переменную?
2. Является ли строка массивом?
3. Как посчитать кол-во повторений элемента в массиве? Как посчитать кол-во повторений символа в строке?
4. Как заполнить строку нужным количеством символов?



Решение задачи 1

Для решения данной задачи необходимо создать переменную строкового типа. Создаём строку `str` и сразу же её инициализируем:

```
string str = "Hello world!";
```

Также создаём символьную переменную, значение которой будем искать в строке:

```
string sym = 'o';
```

Для подсчёта количества повторений символа в строке будем использовать обычный цикл `for`.

Чтобы записывать результат, нам необходима ещё одна переменная, которая будет хранить в себе все найденные символы:

```
string res;
```

Цикл должен перебрать все символы в строке, поэтому ему необходимо прописать правильное условие.

Для этого мы будем использовать метод `.length()`. Подключаем библиотеку `<string>` в самом начале программы:

```
#include <string>
```



Теперь используем синтаксис массивов и перебираем все элементы строки в цикле.

В теле цикла ставим сравнение текущего элемента строки с необходимым нам символом. Если условие верное, то выполняем конкатенацию строк, т. е. прибавляем к строке **res** значение символа **sym**:

```
for (int i = 0; i < str.length(); i++)  
    if (str[i] == sym)  
        res += sym;
```

Строка готова! Остаётся только вывести её значение в консоли:

```
cout << res << endl;  
// на экране oo
```