

## Templates &amp; Lambda

## 1 Count

[STL/02\_templates/task01\_count.cpp]

Разработайте класс с названием `STL`, содержащий статическую функцию, позволяющую подсчитывать «частотность» элемента в массиве. Функции в качестве аргументов передается массив (при необходимости, количество элементов) и искомое значение. Функция подсчитывает, сколько раз искомое значение встречается в массиве и возвращает результат подсчета.

Пример работы программы:

```
int grade[7] = {2, 3, 2, 3, 2, 2, 5};

// выведет 4 (количество двоек в массиве grade из 7-ми элементов)
cout << Счёт::счет<int*>(grade, 7, 2);

// выведет 3 (количество символов 'l' в строке из 11 символов)
cout << Счёт::счет<string>("Hello, world", 11, 'l');
```

## 2 Count If

[STL/02\_templates/task02\_countif.cpp]

Добавьте в класс, созданный в предыдущем задании, статическую функцию, которая подсчитывает количество элементов, удовлетворяющих некоторому условию.

Для проверки работы функции, создайте структуру `Pupil`, содержащую поля: `string имя`, `int возраст`, `int успеваемость`: значения(2,3,4,5). Создайте массив `Pupil` и заполните его. Используя, добавленную функцию и лямбда выражения подсчитайте:

- количество двоечников.
- количество хорошистов, включая отличников.

Подсказка: в качестве лямбда функции задайте функцию, которая получает элемент и возвращает true, если его необходимо учесть при подсчете (т.е. элемент удовлетворяет критериям подсчета).

## 3 For Each

[STL/02\_templates/task03\_foreach.cpp]

Добавьте в класс функцию, выполняющую с каждым элементом массива любое действие. Для проверки, выполните с использованием разработанной функции следующие действия:

- Вывод всех элементов массива в квадратных скобках через запятую
- Очистка массива (заполнить массив нулями)
- Возведение каждого элемента массива в квадрат

## 4 Comparator

[STL/02\_templates/task04\_comparator.cpp]

Реализуйте одну функцию, которая производит сортировку элементов в массиве любого типа методом пузырька. Для проверки работы, отсортируйте:

- числовой массив по возрастанию
- числовой массив по убыванию
- учеников в массиве `Pupil[]` по возрастанию имени

Подсказка: в качестве лямбда функции задайте функцию сравнения, которая получает текущий и следующий элемент массива и возвращает true, если их требуется поменять местами.