

Отчёт по лабораторной работе 9

Корнеев Егор, СКБ-231

Вступление:

В данной лабораторной работе я создал общий класс для условий, функторы для различных типов данных и условий, реализовал требуемую шаблонную функцию.

Условия:

Вариант 4: Шаблонная функция для проверки условия

Напишите шаблонную функцию, которая будет проверять заданное условие для элементов массива и возвращать количество элементов, удовлетворяющих условию. Проверьте работу функции на массивах разных типов данных и разных условиях.

Процедура:

1. Создал шаблонный класс-предок `SampleCondition` с виртуальным определением оператора “()”, чтобы далее наследовать от этого класса другие классы условий (функторы);
2. Создал классы-наследники (функторы), проверяющие различные условия и работающие с различными типами данных:
 - a. `IsEven`, который возвращает `true`, если целое число чётное;
 - b. `IsOdd`, который возвращает `true`, если целое число нечётное;
 - c. `HasLetter`, который возвращает `true`, если в слове есть выбранная буква;
 - d. `IsFalse`, который возвращает `true`, если встречается значение `false`;
 - e. `IsBiggerThanPi`, который возвращает `true`, если нецелое число больше числа π (число π записано с точностью до 13 знака после запятой).
3. Создал функцию `CountMatching`, которая подсчитывает количество элементов в массиве, удовлетворяющих выбранному условию.
4. Создал функцию `main`, в которой определил массивы с различными типами данных, а также проверил работу условий с этими массивами.

Код:

<https://github.com/egoridze74/cpp-lab-9>

Об std::function:

Шаблонный класс `std::function` – своеобразная обёртка, служащая для хранения и передачи функций как объектов. Кроме обычных функций, в `std::function` можно передавать также и лямбда-функции (с ними потенциал `std::function` раскрывается ещё лучше).

Заключение:

В ходе лабораторной работы я познакомился с шаблонными функциями и функторами, реализовал проверку различных условий для массивов с разными типами данных.