

Функторы

№ урока: 2 **Курс:** C++ Advanced

Средства обучения: Qt Creator

Обзор, цель и назначение урока

Научить студентов понимать и применять на практике функторы и их разновидности\смежные инструменты: указатели на функции, `std::function`, `std::bind`, лямбда-выражения.

Изучив материал данного занятия, учащийся сможет

- Понимать, что такое функтор.
- Уметь объяснить разницу между функтором и указателем на функцию, в чем их недостатки и преимущества.
- Понимать синтаксис лямбда-выражения, стандарт C++11, C++14.
- Знать для чего необходим `std::bind` и уметь применять на практике.

Содержание урока

1. Что такое функторы?
2. Указатели на функции и их недостатки по сравнению с функторами
3. `std::function`
4. Лямбда выражения (C++11, C++14)

Резюме

- `class Functor` – класс, в котором перегружен оператор `()`, является функтором.
- `[]{}()` – минимальная запись лямбда-выражения
- `std::function` – полиморфная обертка функции общего назначения.
- `Init capture` – особенность лямбд, добавленная в C++14.
- `std::bind` – служит для переадресации функции, с возможным видоизмененной сигнатурой (порядком) или значением параметров.
- Указатель на функцию – обычный указатель, который ссылается на функцию, должен иметь сигнатуру функции. Ведет себя, как функция (вызов производится обычным способом).

Закрепление материала

- Что такое указатель на функцию? Опишите указатель на функцию, которая принимает 2 строки, принимает другой указатель на функцию, которая возвращает строку и принимает 2 целочисленных значения; и возвращает указатель на функцию, которая, ничего не принимает и ничего не возвращает.
- Что такое функтор? В чем преимущества перед указателем на функцию.
- Что может делать `std::function`. Опишите пример работы с ним.
- Опишите синтаксис лямбда-выражения, полный.
- Как пользоваться `std::bind`?

Дополнительное задание

Задание

Рассмотрите 17 стандарт, какие в нем есть изменения касательно лямбд.

Напишите на C++11 несколько лямбд внутри друг друга, запустите дебаггер или в общем – просмотрите стек вызовов в среде разработки – в чем явное неудобство?

Самостоятельная деятельность учащегося

Задание 1

Выучите основные понятия, рассмотренные на уроке.

Задание 2

Изучите самостоятельно шаблон Listener и понятие callback'a, реализуйте этот шаблон, убедитесь, что коммуникация происходит верным образом. Реализуйте с помощью всех инструментов, изученных на уроке.

Задание 3

Зайдите на сайт MSDN.

Используя поисковые механизмы MSDN, найдите самостоятельно описание темы по каждому примеру, который был рассмотрен на уроке, так, как это представлено ниже, в разделе «Рекомендуемые ресурсы», описания данного урока. Сохраните ссылки и дайте им короткое описание.

Рекомендуемые ресурсы

<https://ru.cppreference.com/w/cpp/language/lambda>

<https://docs.microsoft.com/en-us/cpp/standard-library/function-class?view=vs-2017>

<https://docs.microsoft.com/en-us/cpp/standard-library/functional?view=vs-2017>