

Простейшие пользовательские типы данных

№ урока: 1 **Курс:** C++ Essential

Средства обучения: Qt Creator

Обзор, цель и назначение урока

Научить студентов создавать структуру, объединение и перечисление, объяснить разницу каждого, показать примеры использования.

Изучив материал данного занятия, учащийся сможет

- Понимать, для чего нужны пользовательские типы данных.
- Создавать перечисления и классы перечисления, структуры, объединения.
- Понимать и уметь объяснить разницу каждого типа данных.
- Применять полученные знания на практике.

Содержание урока

1. Что такое пользовательский тип данных, для чего он нужен?
2. Переопределение (псевдоним) типа.
3. Перечисления, свойства и применимость.
4. Класс перечисления, разница с обычным перечислением.
5. Понятие структуры.
6. Применение структур на практике.
7. Объединения, их особенности и использование.
8. Вложенные типы данных.

Резюме

- `struct Car` – описывает машину как отдельную сущность, имеющую свои характеристики: длину, ширину, количество колес, тип двери и т.д.
- `enum class Color` – проводит соответствие: «название цвета» - число;
- `struct Door` – описывает дверь как сущность
- `struct Glass` – описание стекла
- `sizeof(argument)` – вычисляет размер аргумента, который он занимает в памяти
- `my_car.length` – обращение к полю `length` объекта `my_car` структуры `Car`
- `typedef unsigned int uint` - пример переопределения названия типа `unsigned int` на собственный
- `union Container` – объединение, которое содержит в себе два поля, занимаемый размер объединения – размер `double_field`

Закрепление материала

- Что такое пользовательский тип данных? Какие типы Вы знаете?
- Что позволяет сделать `enum`? Каковы преимущества `enum class`?
- Чем отличается `struct` от `union`? Что схожего у них?
- Сколько места в памяти занимает структура? А объединение?

Дополнительное задание

Задание

Создать описание жилой комнаты с помощью всех пользовательских типов данных, которые были рассмотрены на уроке. Свое решение сделать максимально емким по памяти.

Самостоятельная деятельность учащегося

Задание 1

Выучите основные понятия, рассмотренные на уроке.

Задание 2

Создайте свою структуру данных медицинской карточки пациента. Подумайте, каким образом другому программисту было бы удобно работать с этой структурой (логический переход по полям структуры, где целесообразно использовать перечисления и объединения).

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

C++, enum, struct, union

<https://msdn.microsoft.com/ru-ru/library/2dzy4k6e.aspx>