

Часть 1. Объектно-ориентированное программирование на C++

1. Проблема сложности программного обеспечения. Механизмы контроля сложности, абстрагирование, многократное использование.
2. Понятие объектно-ориентированного программирования. Основные принципы.
3. Понятие класса. Понятие метода класса. Понятие сообщения. Классы и функции-члены классов в C++.
4. Экземпляры классов. Создание, инициализация и разрушение. Конструкторы и деструкторы классов в C++.
5. Понятие наследования. Принцип подстановки. Формы наследования. Наследование в C++.
6. Множественное наследование. Проблемы множественного наследования. Множественное и виртуальное наследование в C++.
7. Понятие полиморфизма. Совместное использование (перегрузка) функций в C++.
8. Совместное использование (перегрузка) операций в C++.
9. Замещение методов. Виртуальные функции и абстрактные классы в C++.
10. Видимость. Связность, разновидности связности. Управление видимостью в C++

Часть 2. Современные парадигмы программирования в C++

1. Механизм шаблонов: что такое шаблоны, зачем они нужны, как используются. Шаблоны функций. Настройка шаблонов функций. Пример шаблона функции и настройки шаблона функции.
2. Шаблоны классов. Понятие настройки шаблона класса. Синтаксис и семантика настроек. Пример шаблона класса и его настройки.
3. Обобщенное программирование: что это такое, в чем общий смысл понятия. Библиотека STL как пример реализации принципов обобщенного программирования. Основные компоненты STL, их назначение и смысл.

4. Понятие итератора как обобщенного указателя. Пример итератора для односвязного списка.
5. Итераторы в STL. Иерархия итераторов. Виды итераторов и их назначение. Итераторы, контейнеры и алгоритмы в библиотеке STL: их назначение и взаимосвязь.
6. Концепция исключений (исключительных ситуаций) как метод разработки надежных и безопасных программ. Исключительные ситуации в Си++: подход и общая модель.
7. Синтаксис и семантика конструкций возбуждения исключений и обработки исключений. Привести короткий, но заверченный и осмысленный пример программы, работающей с исключительными ситуациями. Спецификация ситуаций для функций: синтаксис и правила задания исключений. Использование объектно-ориентированной методологии при обработке исключений.
8. Понятие физической и логической модульности в языке Си++. Механизм пространств имен как средство реализации логической модульности. Синтаксис и семантика пространств имен (namespaces) в языке Си++.
9. Именованные и неименованные пространства имен. Вложенные пространства имен. Синонимы для имен пространств имен. Синтаксис и семантика доступа к именам из пространств имен. Конструкции объявления использования и директива использования. Правила использования указанных конструкций в классах.
10. Преобразования типов в С++: конструкции `static_cast` и `dynamic_cast`. RTTI.