

Практикум по C++

- 1) Перегрузить некоторую функцию (метод класса) не менее, чем тремя способами.
 - Примеры: функция вычисления наибольшего значения из двух заданных аргументов некоторого типа; конструктор класса.
 - Дополнительные вопросы: неоднозначности при перегрузке.
- 2) Написать функции демонстрирующие технику передачи аргументов по значению, по указателю и по ссылке.
 - Дополнительные вопросы: понятие о ссылке; константные ссылки как аргументы функции.
- 3) Продемонстрировать механизмы приведения типов в стиле C++ (`static cast` и `const cast`).
 - Дополнительные вопросы: назначение операторов приведения типов, в том числе `reinterpret cast` и `dynamic cast`.
- 4) Объявить и определить класс со статическим полем и статическим методом. Продемонстрировать использование статических членов класса.
 - Дополнительные вопросы: статические члены класса; статические переменные, определенные в функциях и методах.
- 5) Определить класс со встраиваемыми функциями-членами.
 - Дополнительные вопросы: понятие о встраиваемых функциях и ограничения, налагаемые на этот механизм.
- 6) Продемонстрировать динамическое выделение и освобождение памяти с помощью операторов `new/delete` и `new []/delete []`. Выделение и освобождение памяти осуществлять в блоке `try/catch` и обрабатывать исключение `bad alloc`.
- 7) Для пользовательского типа перегрузить операторы ввода и вывода.
- 8) Продемонстрировать механизм дружественных отношений между двумя классами.
- 9) Реализовать для класса методы управления копированием: копирующие конструктор и оператор присваивания, а также деструктор.
- 10) Для класса комплексного числа перегрузить следующие операторы: `+=` как член класса; `+` как внешняя функция через `+=`; `==` и через него `!=` как внешние функции.

- Дополнительные вопросы: способы перегрузки операторов; какие операторы нельзя перегружать.
- 11) Реализовать иерархию пользовательских типов и продемонстрировать на ее основе механизм динамического полиморфизма.
 - Примеры: геометрические фигуры; биологические сообщества.
 - Дополнительные вопросы: понятие о наследовании и полиморфизме в C++.
 - 12) Определить виртуальный деструктор для иерархии классов – владельцев ресурсов.
 - Дополнительные вопросы: назначение виртуального деструктора.
 - 13) Определить иерархию с абстрактным базовым классом и продемонстрировать механизм чисто виртуальных функций.
 - 14) Реализовать интеллектуальный указатель со счетчиком ссылок на разделяемый ресурс и привести пример его использования.
 - 15) Разместить библиотеку в собственном пространстве имен и продемонстрировать механизмы обращения к пространствам имен.
 - 16) На примере простой иерархии продемонстрировать механизм множественного наследования.
 - 17) На примере простой иерархии продемонстрировать механизм виртуального наследования.
 - 18) Определить функцию со спецификацией генерируемых исключений. В блоках `try/catch` обратиться к этой функции и обработать генерируемые ей исключения.
 - 19) Написать шаблон некоторой функции и продемонстрировать его использование в частных случаях.
 - Примеры: функция вычисления наибольшего значения из двух заданных аргументов некоторого типа; функция вычисления абсолютного значения аргумента.
 - 20) Реализовать шаблон для стека.