

Вопросы к коллоквиуму №1
по курсу «Программирование на C++»
Лекции 1-8

1. Язык C++: назначение, основные принципы, история развития и современные стандарты. Сравнение C++ и Python.
2. Препроцессор, компиляция, компоновка и исполнение программы. Выбор стандарта языка в параметрах проекта.
3. Структура простейшей программы. Точка входа main. Подключение заголовочных файлов. Пространства имен. Консольный ввод-вывод: `std::cin`, `std::cout`, `std::print` и `std::format`.
4. Фундаментальные типы данных. Размеры и предельные значения типов. Литералы. Объявление и инициализация переменных. Вывод типа `auto`. Константы `const` и `constexpr`.
5. Явные и неявные преобразования типов. Переполнение целых типов. Особенности вычислений и сравнения вещественных чисел.
6. Основные операции и их приоритет. Арифметические, логические и сравнительные операции. Тернарная операция и операция запятая.
7. Управляющие конструкции `if-else` и `switch`. Циклы `for`, `while` и `do-while`. Операторы `break` и `continue`. Вложенные циклы. Математические функции и `assert`.
8. Функции. Объявление, определение и прототип функции. Возвращаемое значение и вывод возвращаемого типа. `void`-функции и досрочный выход.
9. Передача параметров по значению, ссылке и константной ссылке. Ссылки на структуры.
10. Перегрузка функций. Аргументы по умолчанию. Инлайн-функции. Накладные расходы при вызове функции.
11. Многофайловая программа. Заголовочные файлы и файлы реализации. Директивы препроцессора. Условная компиляция и стражи включения.
12. Этапы сборки проекта. Ошибки компиляции и компоновки. Повторное и отсутствующее определение.
13. Объявления и определения. Глобальные и локальные объекты. `extern`. Пространства имен, вложенные и анонимные пространства имен, псевдонимы и `using`-директивы.
14. Модули C++20: назначение, структура и сравнение с заголовочными файлами.
15. Модульное тестирование. Организация основного и тестового проектов. Оформление, запуск и анализ тестов.
16. Тестирование целых и вещественных значений, статических и динамических массивов, исключений, `std::vector` и `std::string`.
17. Структуры. Объявление, инициализация и доступ к полям. Передача структуры по значению, ссылке и константной ссылке. Возврат структуры и ссылки на структуру.
18. Вложенные структуры, массивы структур и структурированные привязки.
19. Перечисления `enum` и `enum class`. Базовый тип перечисления. Использование перечислений в `switch`. `std::to_underlying`.
20. Кортежи `std::tuple` и пары `std::pair`. Доступ и распаковка элементов. `std::tie`, `std::apply` и структурированные привязки.
21. `std::initializer_list`. Псевдонимы типов `using` и `typedef`. Псевдонимы шаблонов.
22. Системы счисления. Двоичная, восьмеричная и шестнадцатеричная запись. Представление целых и вещественных чисел в памяти.
23. Побитовые операции. Побитовое И, ИЛИ, исключающее ИЛИ, инверсия и сдвиги.

24. Битовые маски: установка, сброс, проверка и переключение битов. Упаковка данных.
25. Генерация псевдослучайных чисел. `std::random_device`, `std::mt19937`, `std::default_random_engine` и `std::seed_seq`. Распределения. Сравнение с `rand()`.
26. Статические одномерные массивы. Размещение в памяти, объявление, инициализация и определение размера. Обход и передача массива в функцию. Защита массива от изменения.
27. Указатели. Объявление и разыменование. `nullptr`. Указатели и ссылки. Указатель `void*`. Указатели на структуры.
28. Массивы указателей и указатели на указатели. Указатели и константность. Связь массивов и указателей. Адресная арифметика.
29. Статические двумерные массивы. Организация и инициализация матрицы. Передача двумерного массива в функцию. Связь матриц и указателей.
30. `std::array`: создание, инициализация, методы, итераторы и передача в функцию. Сравнение `std::array` со встроенным массивом.
31. `std::span` как невладеющее представление непрерывного диапазона. Создание, методы, время жизни и ограничения. `std::span::at` в C++26.
32. Указатели на функции. Объявление, инициализация и вызов. Передача указателя на функцию в качестве параметра. Callback-вызовы, предикаты и компараторы.
33. Перегруженные функции и указатели на них. `std::function`. Псевдонимы типов и чтение сложных объявлений.
34. Сортировка пузырьком, выбором и вставками. Идея, реализация и свойства алгоритмов.
35. Быстрая сортировка. Разбиение по схеме Хоара. Выбор опорного элемента. Сложность и оптимизация быстрой сортировки.
36. O-нотация оценки сложности алгоритмов. Основные классы сложности. Лучший, средний и худший случаи. Временная и пространственная сложность. Устойчивость и сортировка на месте.
37. Ошибки времени выполнения. Традиционные способы обработки ошибок. Функции с побочным эффектом.
38. Механизм исключений: `throw`, `try` и `catch`. Перехват нескольких типов и всех исключений. Стандартные и пользовательские исключения. `noexcept`.
39. Гарантии безопасности исключений. Идиома `copy-and-swap`. `std::expected`. Сравнение способов обработки ошибок.