

СОДЕРЖАНИЕ

1. Разработчик на C++. Нетология (Диплом о профессиональной переподготовке 616 ак. ч.).....	2
2. Windows CMD. Stepik (Сертификат)	5
3. Linux и Linux Bash (3 курса). Stepik	5
4. Основы программирования на Си для Linux. Stepik (Сертификат)	7
5. Операционные системы. Stepik (Сертификат)	8
6. Курс Компьютерные сети. МФТИ YouTube (Без сертификата)	8
7. Курс Архитектура ЭВМ. Computer Science Center YouTube (Без сертификата)	8
8. STL.....	8
9. BOOST	8
10. libiconv	9
11. Docker (дополнительно)	9
12. Книга Паттерны Объектно-Ориентированного Проектирования. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. (полностью).....	9
13. Книга Функциональное программирование на языке C++. Чукич И. (полностью).....	9
14. Книга C++ Concurrency in action. Williams A. (полностью)	9
15. Книга Грокаем алгоритмы. Бхаргава А. (полностью)	9
16. Книга C++. Полное руководство. Классическое издание. Герберт Шилдт (частично)	9
17. Книга Эффективный и современный C++: 42 рекомендации по использованию C++11 и C++14. Мейерс Скотт (частично)	9
18. Книга Beginning C++20: From Novice to Professional (2020). Ivor Horton, Peter Van Weert (частично) ..	9
19. Книга Learning Boost C++ Libraries. Arindam Mukherjee (частично)	9
20. Код-шпаргалки (мои примеры кода для копирования синтаксических конструкций)	9
21. Текстовые шпаргалки по программированию, лекции	9
22. Дипломы и сертификаты.....	9

1. Разработчик на C++. Нетология (Диплом о профессиональной переподготовке 616 ак. ч.)
 - 1.1. Основы программирования на C++
 - 1.1.1. Ввод/вывод
 - 1.1.2. Переменные и типы
 - 1.1.3. Приведение типов
 - 1.1.4. Арифметические операции
 - 1.1.5. Логические операции, ветвление
 - 1.1.6. Циклы
 - 1.1.7. Массивы
 - 1.1.8. Сортировки
 - 1.1.9. Функции
 - 1.1.10. Рекурсия
 - 1.1.11. Модель памяти, типы памяти, область видимости, пространства имен
 - 1.1.12. Работа с динамической памятью
 - 1.1.13. Строки
 - 1.1.14. Файлы
 - 1.1.15. Указатели на функции
 - 1.1.16. User defined literals
 - 1.1.17. Домашние работы (отметка Homework):
<https://replit.com/@IliiaSurikov>
 - 1.1.18. Курсовая работа (Project Life):
https://replit.com/@IliiaSurikov/20221225ProjectLife?v=1#_Project_Life.cpp
 - 1.2. Git - система контроля версий
 - 1.2.1. Установка и настройка Git
 - 1.2.2. Работа с локальным репозиторием
 - 1.2.3. Работа с удаленным репозиторием
 - 1.2.4. Командная работа с репозиторием
 - 1.2.5. Markdown, VS Code, .gitignore
 - 1.3. Базовое программирование на C++
 - 1.3.1. Microsoft Visual Studio
 - 1.3.2. Структуры и перечисления, выравнивание, битовые поля в структурах
 - 1.3.3. Классы
 - 1.3.4. Наследование, виртуальные и чисто виртуальные методы, абстрактные классы
 - 1.3.5. ООП. Абстракция, Инкапсуляция, Полиморфизм
 - 1.3.6. Перегрузка операторов, spaceship
 - 1.3.7. Множественное наследование, агрегация, композиция
 - 1.3.8. Многофайловые проекты, модули C++20
 - 1.3.9. Директивы препроцессора и макросы
 - 1.3.10. Коды возврата и исключения
 - 1.3.11. CMake
 - 1.3.12. Статические и динамические библиотеки, сборка и подключение
 - 1.3.13. Домашние работы:
<https://github.com/stars/CxxICE/lists/003-cpp-basic>
 - 1.3.14. Курсовая работа (Project Racing)
https://github.com/CxxICE/2_Project_Racing
 - 1.4. Алгоритмы и структуры данных
 - 1.4.1. Алгоритмическая сложность, бинарный поиск, инвариант
 - 1.4.2. Динамическое программирование, SQRT-декомпозиция, Рекурсия, «Жадные» алгоритмы
 - 1.4.3. Сортировки, устойчивость, адаптивность, сортировка «на месте»
 - 1.4.4. Динамические массивы, списки, стеки, очереди, деки, амортизационная асимптотика
 - 1.4.5. Двоичные деревья, пирамиды/кучи, операции с пирамидой, бинарное дерево поиска, однопроходность, сбалансированность, красно-черные деревья
 - 1.4.6. Ассоциативные массивы, хеш-функции, хеш-таблицы, алгоритм Рабина-Карпа
 - 1.4.7. Графы, реализация на ссылках и на матрице, достижимость, связность, DFS, BFS, поиск циклов и компонент связности
 - 1.4.8. Орграфы, топологическая сортировка, взвешенные орграфы, алгоритм Дейкстры, Форда-Беллмана, Флойда-Уоршелла.
 - 1.4.9. Домашние работы (отметка Algo):
<https://replit.com/@IliiaSurikov>
 - 1.5. Продвинутое программирование на C++
 - 1.5.1. Обзор C++11, C++14, C++17, C++20

- 1.5.2. Препроцессинг, компиляция, линковка, синтаксические и семантические ошибки
- 1.5.3. Идиома RAII
- 1.5.4. Тестирование, Catch2
- 1.5.5. Шаблоны, вариативные шаблоны, auto, концепты C++20
- 1.5.6. Функторы, лямбды, предикаты
- 1.5.7. Контейнеры STL
- 1.5.8. Итераторы, их типы и применение в алгоритмах STL
- 1.5.9. Умные указатели
- 1.5.10. Move семантика, RVO, NRVO, ref и refref методы классов
- 1.5.11. Ranges
- 1.5.12. Алгоритмы STL
- 1.5.13. Домашние работы
<https://github.com/stars/CxxICE/lists/005-cpp-advanced>
- 1.5.14. Курсовая работа (Parser ini-file)
https://github.com/CxxICE/5_Parser_ini_File
- 1.6. Работа с базами данных
 - 1.6.1. Типы баз данных, структура, атрибуты, кортежи, отношения, первичные ключи, внешние ключи
 - 1.6.2. BASE vs ACID
 - 1.6.3. CAP и PACELC
 - 1.6.4. Нормальные формы
 - 1.6.5. DDL, DML, TCL, DCL
 - 1.6.6. Типы данных Postgre SQL, преобразование типов, работа со строками и датами
 - 1.6.7. Связи между отношениями
 - 1.6.8. Последовательность инструкций при написании и при выполнении
 - 1.6.9. Создание таблиц и запросы
 - 1.6.10. Объединение таблиц
 - 1.6.11. Индексы
 - 1.6.12. Вложенные запросы
 - 1.6.13. Агрегация и группировка
 - 1.6.14. Оконные функции
 - 1.6.15. Представления
 - 1.6.16. libpqxx
 - 1.6.17. SQL Injection
 - 1.6.18. ORM, wtdbo
 - 1.6.19. Домашние работы
<https://github.com/stars/CxxICE/lists/006-cpp-sql>
- 1.7. Многопоточное и асинхронное программирование
 - 1.7.1. Многозадачность и параллелизм
 - 1.7.2. Chrono STL
 - 1.7.3. Политики выполнения STL
 - 1.7.4. Поток, запуск, передача и возврат данных потока (thread, jthread, call_once)
 - 1.7.5. Конкуренция, состояние гонки, гонка за данными
 - 1.7.6. Синхронизация, Deadlock-и (mutex, barrier, latch, counting_semaphore, condition_variable, lock_guard, scoped_lock, unique_lock, shared_lock)
 - 1.7.7. Атомарные типы данных и операции, упорядочивание доступа к памяти
 - 1.7.8. Асинхронное программирование (future, promise, packaged_task, async)
 - 1.7.9. Рефакторинг
 - 1.7.10. Docker
 - 1.7.11. Корутины C++20, Корутины Boost
 - 1.7.12. Домашние работы
<https://github.com/stars/CxxICE/lists/007-multithreading>
 - 1.7.13. Курсовая работа (Threadsafe queue)
https://github.com/CxxICE/7_Threadsafe_queue
- 1.8. Шаблоны проектирования в C++
 - 1.8.1. DRY и SOLID
 - 1.8.2. Порождающие паттерны
 - 1.8.2.1. Abstract Factory (Абстрактная фабрика)
 - 1.8.2.2. Builder (Строитель)
 - 1.8.2.3. Factory Method (Фабричный метод)
 - 1.8.2.4. Prototype (Прототип)
 - 1.8.2.5. Singleton (Синглтон)
 - 1.8.3. Структурные паттерны

- 1.8.3.1. Adapter (Адаптер)
- 1.8.3.2. Bridge (Мост)
- 1.8.3.3. Composite (Компоновщик)
- 1.8.3.4. Decorator (Декоратор)
- 1.8.3.5. Facade (Фасад)
- 1.8.3.6. Flyweight (Приспособленец)
- 1.8.3.7. Proxy (Заместитель)
- 1.8.4. Паттерны Поведения
 - 1.8.4.1. Chain of Responsibility (Цепочка ответственности/обязанностей)
 - 1.8.4.2. Command (Команда)
 - 1.8.4.3. Interpreter (Интерпретатор)
 - 1.8.4.4. Iterator (Итератор)
 - 1.8.4.5. Mediator (Посредник)
 - 1.8.4.6. Memento (Хранитель)
 - 1.8.4.7. Observer (Наблюдатель)
 - 1.8.4.8. State (Состояние)
 - 1.8.4.9. Strategy (Стратегия)
 - 1.8.4.10. Template Method (Шаблонный метод)
 - 1.8.4.11. Visitor (Посетитель)
- 1.8.5. Домашние работы
 - <https://github.com/stars/CxxICE/lists/008-patterns>
- 1.9. Кроссплатформенная разработка на Qt
 - 1.9.1. Обзор Qt 6.4.0, лицензирование, установка, настройка
 - 1.9.2. Модули Qt
 - 1.9.3. Qt Designer
 - 1.9.4. QWidget
 - 1.9.5. QObject, Слоты и сигналы, Метаобъектный компилятор
 - 1.9.6. Qt Concurrent
 - 1.9.7. QtCharts и QCustomPlot
 - 1.9.8. Работа с базами данных в Qt
 - 1.9.9. Работа с сетью в Qt (TCP, UDP)
 - 1.9.10. Wireshark
 - 1.9.11. Домашние работы
 - <https://github.com/stars/CxxICE/lists/009-qt>
 - 1.9.12. Курсовая работа (Airport Inspector)
 - https://github.com/CxxICE/9_Qt_Airport_Inspector
- 1.10. Разработка умных вещей (IoT)
 - 1.10.1. Arduino, основные команды и понятия
 - 1.10.2. Дискретные входы и выходы, подавлениедребезга
 - 1.10.3. Аналоговый выход, ШИМ
 - 1.10.4. Аналоговый вход, АЦП
 - 1.10.5. Джойстик
 - 1.10.6. Матричная клавиатура
 - 1.10.7. Семисегментные индикаторы
 - 1.10.8. Жидкокристаллический дисплей
 - 1.10.9. NTC термистор
 - 1.10.10. Ультразвуковой дальномер
 - 1.10.11. Датчик движения на основе пироприемника
 - 1.10.12. МЭМС (микроэлектромеханическая системы). Акселерометр, гироскоп
 - 1.10.13. I2C
 - 1.10.14. Шаговый электродвигатель, потенциометр, сервопривод
 - 1.10.15. ИК пульт
 - 1.10.16. Прерывания
 - 1.10.17. SPI, сдвиговый регистр, работа с SD картой
 - 1.10.18. Wi-Fi, Bluetooth, ZigBee
 - 1.10.19. ESP32, HTTPClient
 - 1.10.20. 1-Wire, Датчик температуры и влажности, ПИД-регулятор
 - 1.10.21. Домашние работы:
 - <https://wokwi.com/makers/cxxice>
 - 1.10.22. Курсовая работа (Smart Home):
 - <https://wokwi.com/projects/383835342143394817>
- 1.11. Разработка видеоигр на Unreal Engine 5
 - 1.11.1. GameDev, общие сведения

- 1.11.2. Unreal Engine 5 установка и настройка, UE Editor
- 1.11.3. Основные классы Unreal Engine 5
- 1.11.4. Строки, OutputLog, UPROPERTY, UFUNCTION, USTRUCT, UENUM
- 1.11.5. Компоненты в UE5
- 1.11.6. Работа в коде C++ и Blueprints
- 1.11.7. Делегаты UE5
- 1.11.8. Структура проекта, именование сущностей, .clang-format, .gitignore
- 1.11.9. ACharacter, APlayerController, AGameModeBase, C++ Classes, Blueprint classes
- 1.11.10. Capsule component, Arrow component, Mesh, Character movement
- 1.11.11. CameraComponent, SpringArmComponent
- 1.11.12. Mapping axis, actions, trace, collision
- 1.11.13. Анимация, миграция, ретаргетинг
- 1.11.14. Animation blueprint (Event graph, Anim graph, State Machine), Blend Space
- 1.11.15. Health Component, Damage Actor, Animation montage, Spectator Pawn
- 1.11.16. Health Pickup
- 1.11.17. Weapon component, Skeletal mesh, Sockets, Trace
- 1.11.18. AnimNotify, Blend per bone
- 1.11.19. UI и HUD
- 1.11.20. Меню и уровни
- 1.11.21. Level blueprint, Timer, Events, TriggerBox
- 1.11.22. Освещение, звук, Спецэффекты
- 1.11.23. Искусственный интеллект в игре (AIController)
- 1.11.24. Behavior Tree, Blackboard, Tasks, Decorator, Services, Navigation
- 1.11.25. Курсовая работа (Leave Me Alone Game):
https://github.com/CxxICE/11_LeaveMeAlone
- 1.12. Дипломная работа C++
https://github.com/CxxICE/12_Crowler_Search_Engine
- 1.13. Все домашние работы и курсовые разбитые по категориям
 - 1.13.1. Основной раздел:
<https://github.com/CxxICE?tab=stars>
 - 1.13.2. Основы C++, Алгоритмы и структуры данных
<https://replit.com/@IliaSurikov>
 - 1.13.3. IoT:
<https://wokwi.com/makers/cxxice>
- 2. Windows CMD. Stepik (Сертификат)**
<https://stepik.org/course/114966/>
 - 2.1. Командная строка: введение
 - 2.1.1. Варианты запуска Windows CMD
 - 2.1.2. Настройка оболочки - CMD
 - 2.1.3. Общие сведения о Windows CMD
 - 2.1.4. Горячие клавиши для Windows CMD
 - 2.1.5. Переменные среды окружения Windows
 - 2.1.6. Перенаправление ввода/вывода
 - 2.1.7. Системные команды CMD
 - 2.1.8. Базовые команды CMD
 - 2.2. Командная строка: сценарии
 - 2.2.1. Создание сценариев пакетных файлов
 - 2.2.2. Windows CMD переменные
 - 2.2.3. Применение математических выражений
 - 2.2.4. Условные операторы
 - 2.2.5. Циклы в командной строке
 - 2.2.6. Создание подпрограмм и процедур
- 3. Linux и Linux Bash (3 курса). Stepik**
<https://stepik.org/course/126085/>
 - 3.1. Linux - начало пути. Stepik (Без сертификата)
 - 3.1.1. Первоначальное знакомство
 - 3.1.1.1. Linux: что это?
 - 3.1.1.2. Отличие ОС Linux от других операционных систем.
 - 3.1.1.3. Дистрибутивы Linux
 - 3.1.1.4. Графическая оболочка (рабочий стол) Linux
 - 3.1.1.5. Лабораторная работа № 1 Установка ОС Linux Debian 11
 - 3.1.1.6. Лабораторная работ № 2 Работа с рабочим столом Linux
 - 3.1.1.7. Лабораторная работа № 3 Управление файлами и папками

- 3.1.1.8. Тест по разделу № 1
- 3.1.2. Знакомство с графической оболочкой
 - 3.1.2.1. Что такое оболочка Linux
 - 3.1.2.2. Запуск команд
 - 3.1.2.3. Вызов команды
 - 3.1.2.4. Соединение и расширение команд
 - 3.1.2.5. Информация о командах
 - 3.1.2.6. Лабораторная работа № 4 Настройка подключения к системе
 - 3.1.2.7. Тест по разделу 2
- 3.1.3. Файловая система Linux
 - 3.1.3.1. Файловая система ОС linux
 - 3.1.3.2. Базовые команды файловой системы
 - 3.1.3.3. Метасимволы и операторы
 - 3.1.3.4. Владельцы и права доступа к файлам
 - 3.1.3.5. Лабораторная работа № 5 Структура ФС в Linux
- 3.1.4. Текстовые файлы в Linux
 - 3.1.4.1. Редактирование файлов
 - 3.1.4.2. Поиск файлов
 - 3.1.4.3. Лабораторная работа № 6 Работа с файлами
- 3.2. Основы Linux. Stepik (Без сертификата)
<https://stepik.org/course/762/>
 - 3.2.1. Вводная информация о Linux
 - 3.2.1.1. История Linux
 - 3.2.1.2. Дистрибутивы
 - 3.2.1.3. Лицензирование
 - 3.2.1.4. Установка Linux в домашних условиях
 - 3.2.2. Первые шаги в изучении интерфейса командной строки
 - 3.2.2.1. Страницы руководств man
 - 3.2.2.2. Работа с директориями
 - 3.2.2.3. Работа с файлами
 - 3.2.2.4. Работа с содержимым файлов
 - 3.2.2.5. Дерево директорий Linux
 - 3.2.3. Раскрытие команд командной оболочкой
 - 3.2.3.1. Команды и аргументы
 - 3.2.3.2. Операторы управления
 - 3.2.3.3. Переменные командной оболочки
 - 3.2.3.4. Встраивание и параметры командных оболочек
 - 3.2.3.5. История команд командной оболочки
 - 3.2.3.6. Формирование списков имен файлов на основе шаблонов
 - 3.2.4. Программные каналы и команды
 - 3.2.4.1. Перенаправление потоков ввода/вывода
 - 3.2.4.2. Фильтры
 - 3.2.4.3. Стандартные инструменты систем Unix
 - 3.2.4.4. Регулярные выражения
 - 3.2.5. Текстовый редактор vi
 - 3.2.5.1. Начальные сведения о текстовом редакторе vi
 - 3.2.6. Сценарии
 - 3.2.6.1. Введение в разработку сценариев
 - 3.2.6.2. Циклы в сценариях
 - 3.2.6.3. Параметры сценариев
 - 3.2.6.4. Дополнительная информация о сценариях
 - 3.2.7. Управление локальными учетными записями пользователей
 - 3.2.7.1. Вводная информация об учетных записях пользователей
 - 3.2.7.2. Управление учетными записями пользователей
 - 3.2.7.3. Пароли пользователей
 - 3.2.7.4. Профили пользователей
 - 3.2.7.5. Группы пользователей
 - 3.2.8. Механизмы безопасной работы с файлами
 - 3.2.8.1. Стандартные права доступа к файлам
 - 3.2.8.2. Расширенные права доступа к файлам
 - 3.2.8.3. Списки контроля доступа
 - 3.2.8.4. Ссылки на файлы
 - 3.2.9. Приложения

- 3.2.9.1. Приложение В. Аппаратное обеспечение
- 3.3. Введение в Linux. Stepik (Сертификат)
<https://stepik.org/course/73/>
 - 3.3.1. Введение
 - 3.3.1.1. Общая информация о курсе
 - 3.3.1.2. Как установить Linux
 - 3.3.1.3. Осваиваем Linux
 - 3.3.1.4. Terminal: основы
 - 3.3.1.5. Запуск исполняемых файлов
 - 3.3.1.6. Ввод / вывод
 - 3.3.1.7. Скачивание файлов из интернета
 - 3.3.1.8. Работа с архивами
 - 3.3.1.9. Поиск файлов и слов в файлах
 - 3.3.2. Работа на сервере
 - 3.3.2.1. Знакомство с сервером
 - 3.3.2.2. Обмен файлами
 - 3.3.2.3. Запуск приложений
 - 3.3.2.4. Контроль запускаемых программ
 - 3.3.2.5. Многопоточные приложения
 - 3.3.2.6. Менеджер терминалов tmux
 - 3.3.2.7. Как установить Linux: расширенное руководство
 - 3.3.3. Продвинутое темы
 - 3.3.3.1. Текстовый редактор vim
 - 3.3.3.2. Скрипты на bash: основы
 - 3.3.3.3. Скрипты на bash: ветвления и циклы
 - 3.3.3.4. Скрипты на bash: разное
 - 3.3.3.5. Продвинутый поиск и редактирование
- 4. Основы программирования на Си для Linux. Stepik (Сертификат)
<https://stepik.org/course/548/>
 - 4.1. Введение в разработку для Linux
 - 4.1.1. Основной инструментарий разработчика Linux
 - 4.1.2. Динамические библиотеки и линковка
 - 4.1.3. Runtime линковка. Libdl
 - 4.2. Файлы и файловые системы
 - 4.2.1. Базовые функции работы с файлами
 - 4.2.2. Работа с каталогами и путями
 - 4.2.3. Ссылки
 - 4.2.4. Файловая система /proc. Построение дерева процессов
 - 4.2.5. Отладка программ в Linux
 - 4.3. Процессы и потоки
 - 4.3.1. Жизненный цикл процесса в Linux
 - 4.3.2. Создание процессов fork/exec
 - 4.3.3. Атрибуты создаваемого процесса. Системный вызов clone
 - 4.3.4. Процессы-демоны
 - 4.3.5. Задачи на программирование
 - 4.4. Взаимодействие процессов
 - 4.4.1. Виды межпроцессного взаимодействия в Linux
 - 4.4.2. Каналы
 - 4.4.3. Сигналы
 - 4.4.4. Разделяемая память
 - 4.4.5. Мультиплексирование ввода-вывода.
 - 4.4.6. Задачи на программирование
 - 4.5. Сетевые возможности
 - 4.5.1. Введение
 - 4.5.2. Принципы организации IP сетей
 - 4.5.3. Понятие сокета
 - 4.5.4. Разрешение имен
 - 4.5.5. Сокеты дейтаграм. Протокол UDP
 - 4.5.6. Сокеты ориентированные на соединения. Протокол TCP
 - 4.5.7. Задачи на программирование
 - 4.6. Заключительный модуль
 - 4.6.1. Финальный урок
 - 4.7. Дополнительный и справочный материал

- 4.7.1. Дополнительные материалы
 - 4.8. Дополнительные темы
 - 4.8.1. Обработка параметров командной строки. Getopt
- 5. **Операционные системы. Stepik (Сертификат)**
<https://stepik.org/course/1780/>
 - 5.1. Введение
 - 5.1.1. Введение
 - 5.1.2. Язык ассемблера 1
 - 5.1.3. Язык ассемблера 2
 - 5.1.4. Прерывания
 - 5.1.5. Загрузка ОС
 - 5.2. Управление памятью
 - 5.2.1. Физическая память
 - 5.2.2. Логическая память
 - 5.2.3. Сегментация
 - 5.2.4. Страничная организация памяти
 - 5.2.5. Простой подход к аллокации памяти
 - 5.2.6. Buddy аллокатор
 - 5.2.7. SLAB аллокатор
 - 5.3. Планирование и многозадачность
 - 5.3.1. Потоки исполнения и многопоточность
 - 5.3.2. Переключение потоков
 - 5.3.3. Кооперативная и вытесняющая многозадачности
 - 5.3.4. Планирование и критерии планирования
 - 5.3.5. Реалистичное планирование
 - 5.4. Средства синхронизации потоков
 - 5.4.1. Состояние гонки по данным и взаимное исключение
 - 5.4.2. Взаимное исключение с использованием RW регистров, часть 1
 - 5.4.3. Взаимное исключение с использованием RW регистров, часть 2
 - 5.4.4. Взаимное исключение с использованием RMW регистров
 - 5.4.5. Прочие примитивы синхронизации
 - 5.4.6. Deadlock-и и средства борьбы с ними
 - 5.5. Пространство пользователя
 - 5.5.1. Исполняемые файлы и процессы
 - 5.5.2. Динамические библиотеки
 - 5.5.3. Системные вызовы
 - 5.5.4. Средства управления процессами
 - 5.5.5. Средства межпроцессного взаимодействия
- 6. **Курс Компьютерные сети. МФТИ YouTube (Без сертификата)**
https://www.youtube.com/playlist?list=PLthfp5exSWErPFK_-EAhVtxO3XoY0gsSe
 - 6.1. Модель OSI (Open Systems Interconnection)
 - 6.2. Frame Ethernet
 - 6.3. IPv4 Frame
 - 6.4. IPv4 адресация
 - 6.5. ARP, Proxy ARP, DORA, DHCP
 - 6.6. UDP frame
 - 6.7. ICMP
 - 6.8. TCP segment, установление и закрытие соединения, оптимизации
 - 6.9. Протокол маршрутизации RIP
 - 6.10. IPv6 адресация, SLAAC
 - 6.11. NAT, PAT
 - 6.12. VLAN, QnQ
 - 6.13. Протоколы резервирования STP, RSTP, FHRP
 - 6.14. PortChannel, LACP
 - 6.15. Протокол маршрутизации OSPF
 - 6.16. Туннели IPIP, VxLAN
- 7. **Курс Архитектура ЭВМ. Computer Science Center YouTube (Без сертификата)**
<https://www.youtube.com/playlist?list=PLnseyzyGdZdfv8H7LkvyVVE33fbBZaSdH>
- 8. **STL**
- 9. **BOOST**
 - 9.1. Asio
 - 9.2. Beast
 - 9.3. Coroutine

- 9.4. Phoenix
- 9.5. URL
- 10. libiconv
- 11. Docker (дополнительно)
<https://www.youtube.com/watch?v=O8N1lvkljig&list=PLMj0F28ZbVqMP7E5QXlrAMCWTjSliTbui&index=1>
- 12. Книга Паттерны Объектно-Ориентированного Проектирования. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. (полностью)
- 13. Книга Функциональное программирование на языке C++. Чукич И. (полностью)
- 14. Книга C++ Concurrency in action. Williams A. (полностью)
- 15. Книга Грокаем алгоритмы. Бхаргава А. (полностью)
- 16. Книга C++. Полное руководство. Классическое издание. Герберт Шилдт (частично)
- 17. Книга Эффективный и современный C++: 42 рекомендации по использованию C++11 и C++14. Мейерс Скотт (частично)
- 18. Книга Beginning C++20: From Novice to Professional (2020). Ivor Horton, Peter Van Weert (частично)
- 19. Книга Learning Boost C++ Libraries. Arindam Mukherjee (частично)
- 20. Код-шпаргалки (мои примеры кода для копирования синтаксических конструкций)
 - 20.1. C++
https://drive.google.com/drive/folders/1LhtBWmUUXrAAMANe_ad3fuO40DEL60RC?usp=sharing
 - 20.2. CLR .Net Framework
https://drive.google.com/drive/folders/1_vBlcM6oozSb9O6v5cuZawDjOn-9acET?usp=sharing
 - 20.3. Операционные системы
<https://drive.google.com/drive/folders/10-YOy9cxCJiS64qCgsE5UQJYvfxCdMUh?usp=sharing>
 - 20.4. Linux C++
<https://drive.google.com/drive/folders/1CMc1NXByeGAjZNcmXtfsD7tbbWLttWdV?usp=sharing>
- 21. Текстовые шпаргалки по программированию, лекции
https://drive.google.com/drive/folders/1KYI2GDd5zjWUwjj_oPxtJF_9HnMkuLOL?usp=sharing
- 22. Дипломы и сертификаты
<https://drive.google.com/drive/folders/1yA-aLgQqHpQ0X1z9dU2CzFymQd-qwWpQ?usp=sharing>