# ЭКЗАМЕНАЦИОННЫЕ ВОПРОСЫ ПО ДИСЦИПЛИНЕ «КОНСТРУИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ»

2023 год.

## ЧАСТЬ 0. ОСНОВЫ КОНСТРУИРОВАНИЕ ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ

1. Конструирование программного обеспечения (Software Construction, SwC). Основные процессы разработки программного обеспечения.
2. Процессы и подпроцессы разработки программного обеспечения.
3. Жизненный цикл программного продукта.
4. Искусственный интеллект. Нейронные сети. Роль и влияние искусственного интеллекта в программировании. Инструменты искусственного интеллекта в программировании.
5. Машинное обучение. Инструменты машинного обучения.
6. Методология DevOps.

#### (6)

## ЧАСТЬ I. ШАБЛОНЫ ПРОЕКТИРОВАНИЯ, ПАТТЕРНЫ, ИДИОМЫ

1. Основные группы шаблонов проектирования. Идиомы программирования. Преимущества и недостатки использования шаблонов проектирования.
2. Шаблоны архитектуры системы. Их назначение и классификация. Наиболее популярные и актуальные шаблоны.
3. Архитектура клиент-серверных приложений, межпроцессное взаимодействие. Средства передачи данных между процессами.
4. Архитектура Web-приложений. Межпроцессное взаимодействие в Web-приложениях.
5. Объектно-ориентированные паттерны по предназначению. Типы паттернов по предназначению.
6. Порождающие паттерны по предназначению. Преимущества и недостатки. Наиболее популярные и актуальные шаблоны.
7. Структурные паттерны по предназначению. Преимущества и недостатки. Наиболее популярные и актуальные шаблоны.
8. Поведенческие паттерны по предназначению. Преимущества и недостатки. Наиболее популярные и актуальные шаблоны.

#### (8)

## ЧАСТЬ II. ПАРАДИГМЫ ПРОГРАММИРОВАНИЯ

1. **Понятие «Парадигма программирования», Структурное программирование. Управляющие структуры в программировании. Блоки и блочная видимость. Подпрограммы.**
2. **Многофайловые проекты. Подпрограммы, макросы, библиотеки, плагины. Дублирование кода. Программные модули. Преимущества и недостатки.**
3. **Многофайловые проекты. Библиотеки. Статические и динамические библиотеки. Статическое и динамическое подключение библиотек. Ад dll (Dll hell).**
4. **Императивная парадигма программирования. Классификация императивной парадигмы. Присвоение и переменные.**
5. **Декларативная парадигма программирования. Классификация декларативной парадигмы. Функциональная парадигма. Логическая парадигма. Управление данными (базы данных).**
6. **Процедурная парадигма программирования (Procedural programming). Понятия процедуры и присвоения. Алгоритм и алгоритмическая декомпозиция.**
7. **Объектно-ориентированная парадигма программирования. Понятие «Объект» и «Класс».**
8. **Парадигма параллельных вычислений. Многопоточность и многозадачность. Синхронизация и ассинхронизация.**

#### (8)

## ЧАСТЬ III. ОСНОВНЫЕ ПРИНЦИПЫ ООП

1. **Основные четыре принципа ООП. Гибкость ООП. Процедурная парадигма в ООП (Статические классы и члены классов).**
2. **Принцип ООП – Наследование. Агрегирование, делегирование. Виды агрегирования: агрегация и композиция.**
3. **Принцип ООП – Наследование. Шаблон делегирования. Модель «Включение-делегирование». Модель внутренних типов, контейнерные и внутренние классы. Вложенные типы.**
4. **Принцип ООП – Наследование. Иерархии классов. Множественное наследование, «Смертельный ромб». Наследование конструкторов. Запрет наследования.**
5. **ООП. Интерфейсы и инкапсуляция. Типы интерфейсов. Потенциальный конфликт между именами методов разных интерфейсов.**
6. **ООП. Интерфейсы и наследование. Типы интерфейсов. Множественное наследование интерфейсов. «Смертельный ромб».**
7. **Сериализация. Технологии сериализации: Двоичная сериализация, Сериализация XML и SOAP. Сериализация JSON.**

#### (7)

/////////////////////////////////////////////////////////

## ЧАСТЬ IV. ИНСТРУМЕНТЫ ДЛЯ КОНСТРУИРОВАНИЯ ПО

1. Системы контроля версиями. Наиболее популярная система контроля версий – Git. Удаленные и локальные репозитории. Стратегии разработки в системах контроля версий.
2. Контейнеризация, технология Docker.
3. Типы ошибок. Сообщения об ошибках. Отладка приложения. Инструменты отладки. Трансляция программы под отладчиком.
4. Архиваторы. Система управления пакетами, инсталляторы и пакетные менеджеры.
5. Интерпретаторы. Скриптовы языки программирования. Сценарии и макросы.
6. Регулярные выражения. Инструменты для проверки работы регулярных выражений. Области применения регулярных выражений.

#### (6)

## ЧАСТЬ V. МЕТОДОЛОГИИ РАЗРАБОТКИ

1. Кризис программного обеспечения. Закон Гордана Мура. Законы Паркенсона.
2. Кризис программного обеспечения. Макаронный (спагетти) код. Кенгуру-код и инструкция «jump». Оператор GOTO.
3. Программная инженерия (Software engineering). Элементы программной инженерии. Жизненный цикл ПО.
4. Программная инженерия (Software engineering). Рефакторинг. Оптимизации производительности.
5. Реинжиниринг. Методики обратной разработки ПО: Анализ обмена данными. Дизассемблирование машинного кода программы. Декомпиляция машинного или байт-кода программы.
6. Проприетарное, свободное и открытое ПО. Конфликт с авторскими правами. Проект GNU и лицензия GPL. Критерии свободной программы: 4 степени свободы свободного ПО.
7. Лицензирование программного обеспечения, типы лицензий.
8. Процесс компиляции приложения. Декомпиляция. Препроцессор. Ассемблирование и деассемблирование. Используемые для этого средства.
9. Композиция приложения. «Хороший» стиль программирования. Принцип «разделения интересов (ответственности)» (Separation of concerns). Принцип единственной обязанности (Single Responsibility Principle, SRP).
10. Две стратегии разработки программ: Нисходящее и восходящее программирование. Дедукция и индукция.
11. Принципы разработки программного обеспечения SOLID.

#### (11)

////////////////////////////////////////\

### 6 + 7 + 8 + 7 + 6 + 11 = 44 вопроса.