

**Федеральное государственное образовательное бюджетное учреждение
высшего образования
«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ
РОССИЙСКОЙ ФЕДЕРАЦИИ»
(Финансовый университет)**

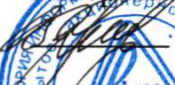
Тульский филиал Финуниверситета

Кафедра «Математика и информатика»

СОГЛАСОВАНО

ЗАО «ЛИМ»

Генеральный директор

 **В.В. Куликов**
«19» декабря 2024 г.



УТВЕРЖДАЮ

Директор филиала

 **Г.В. Кузнецов**

«24» декабря 2024 г.



Г.В. Ваныкина

**ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ
АРХИТЕКТУРА И ДИЗАЙН ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

для студентов, обучающихся по направлению подготовки
09.03.02 Информационные системы и технологии,
образовательная программа «Информационные технологии в экономике»,
профиль «Информационные технологии в управлении цифровой
экономикой»

Рекомендовано Ученым советом Тульского филиала Финуниверситета
(протокол от 24 декабря 2024 г. № 22)

Одобрено кафедрой «Математика и информатика»
(протокол от 02 декабря 2024 г. № 5)

Тула 2024

«Архитектура и дизайн программного обеспечения»

КОД И НАИМЕНОВАНИЕ ОПОП:

09.03.02 Информационные системы и технологии

Образовательная программа - Информационные технологии в экономике

Профиль «Информационные технологии в управлении цифровой экономикой»

Общее количество заданий по дисциплине

Код компетенции	Наименование компетенции	Количество заданий
ПКП-1	Способен описывать, анализировать и проектировать компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	40
Всего:		40

Количество заданий по дисциплине в одном варианте

Код компетенции	Наименование компетенции	Количество заданий
ПКП-1	Способен описывать, анализировать и проектировать компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	20
Всего:		20

Типы заданий

Наименование типа заданий	Номер типа заданий
Задание комбинированного типа с выбором одного верного ответа из предложенных	1
Задание комбинированного типа с выбором нескольких вариантов ответа из предложенных	2
Задание открытого типа	3
Задание открытого типа с развернутым ответом	4

Распределение заданий по компетенциям

Код компетенции	Наименование компетенции	Наименование индикаторов сформированности компетенции	Семестр	Номер задания
ПКП-1	Способен описывать, анализировать и проектировать компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	1. Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	7	1.1, 1.2, 2.1, 2.2, 3.1, 3.2, 4.1, 4.2, 5.1, 5.2, 6.1, 6.2, 7.1, 7.2, 8.1, 8.2, 9.1, 9.2, 10.1, 10.2
		2. Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	7	11.1, 11.2, 12.1, 12.2, 13.1, 13.2, 14.1, 14.2, 15.1, 15.2, 16.1, 16.2, 17.1, 17.2, 18.1, 18.2,

				19.1, 19.2, 20.1, 20.2
--	--	--	--	---------------------------

Типы, уровень сложности и время выполнения заданий

Код компетенции	Индикатор сформированности компетенции	Номер задания	Тип задания	Уровень сложности задания	Время выполнения задания (мин.)
ПКП-1	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	1.1, 1.2.	1	базовый	2
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	2.1, 2.2	1	базовый	2
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	3.1, 3.2	1	базовый	2
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	4.1, 4.2	1	базовый	2
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	5.1, 5.2	1	базовый	2
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	6.1, 6.2.	3	повышенный	4
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	7.1, 7.2	3	повышенный	4
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	8.1, 8.2	3	повышенный	4
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	9.1, 9.2	4	высокий	8
	Описывает и анализирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы.	10.1, 10.2	4	высокий	8
	Описывает и анализирует компьютерное программное	11.1, 11.2.	1	базовый	2

	обеспечение, структуру данных, базы данных, программные интерфейсы.				
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	12.1, 12.2	1	базовый	2
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	13.1, 13.2	2	базовый	2
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	14.1, 14.2	2	базовый	2
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	15.1, 15.2	1	базовый	2
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	16.1, 16.2.	3	повышенный	4
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	17.1, 17.2	3	повышенный	4
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	18.1, 18.2	4	высокий	8
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	19.1, 19.2	4	высокий	8
	Проектирует компьютерное программное обеспечение, структуру данных, базы данных, программные интерфейсы с учетом требований к ним	20.1, 20.2	4	высокий	8

Задания, позволяющие осуществлять оценку компетенций, установленных для дисциплины

Задание 1.1.

Что такое архитектура программного обеспечения?

- А) Процесс создания кода
- Б) Структура системы с описанием ее компонентов и их взаимодействия
- В) Указания по стилю оформления кода
- Г) Методология тестирования программного обеспечения

Ответ:

Задание 2.1.

3. Что такое крупномасштабная архитектура?

- А) Архитектура, не учитывающая производительность
- Б) Архитектура, которая используется исключительно для мобильных приложений
- В) Архитектура, ориентированная на множественные сервисы и взаимодействие между ними
- Г) Архитектура, используемая только для бизнес-приложений

Ответ:

Задание 3.1.

Что такое "шаблон проектирования" в контексте архитектуры ПО?

- А) Набор правил для написания кода
- Б) Повторяемое решение общей проблемы проектирования
- В) Интерфейс для взаимодействия с базой данных
- Г) Спецификация для тестирования системы

Ответ:

Задание 4.1.

Какой из следующих этапов является частью процесса проектирования архитектуры программного обеспечения?

- А) Тестирование
- Б) Разработка
- В) Анализ требований
- Г) Поддержка

Ответ:

Задание 5.1.

Какая архитектурная модель подходит для систем, требующих высокой гибкости и возможности быстрого внесения изменений?

- А) Монолитная архитектура
- Б) Микросервисная архитектура
- В) Многоуровневая архитектура
- Г) Архитектура с общей шиной

Ответ:

Задание 6.1.

Какой из принципов архитектуры программного обеспечения подразумевает разделение проекта на самостоятельные компоненты?

Ответ:

Задание 7.1.

Как называется подход, основанный на создании программного обеспечения с использованием небольших, независимых модулей, которые могут развиваться независимо?

Ответ:

Задание 8.1.

Какой тип архитектуры часто применяется при создании веб-приложений с использованием клиент-серверной модели?

Ответ:

Задание 9.1.

Каковы ключевые принципы проектирования архитектуры программного обеспечения, и как они влияют на масштабируемость и поддерживаемость системы?

Ответ:

Задание 10.1.

Какова роль паттернов проектирования в разработке программного обеспечения, и какие из них наиболее востребованы в современной практике?

Ответ:

Задание 11.1.

Какой из следующих компонентов является ключевым элементом многослойной архитектуры?

- А) Слой представления
- Б) Слой баз данных
- В) Слой бизнес-логики
- Г) Все вышеперечисленное

Ответ:

Задание 12.1.

Какой из следующих паттернов проектирования используется для создания объектов, которые впоследствии можно изменить?

- А) Паттерн "Фабрика"
- Б) Паттерн "Singleton"
- В) Паттерн "Стратегия"
- Г) Паттерн "Команда"

Ответ:

Задание 13.1.

Какие из следующих методов могут использоваться для создания архитектурных решений? (выберите все подходящие)

- А) Применение паттернов проектирования
- Б) Использование миграции данных
- В) Разработка на основе реальных потребностей пользователей
- Г) Создание документации по архитектуре

Ответ:

Задание 14.1.

Какие из следующих критериев могут использоваться для оценки качества архитектуры программного обеспечения? (выберите все подходящие)

- А) Масштабируемость
- Б) Безопасность
- В) Производительность
- Г) Многофункциональность

Ответ:

Задание 15.1.

Какой правильный порядок действий при построении архитектуры?

- А) архитектура – требования – логика – структура – код
- Б) требования – структура – архитектура – логика – код
- В) требования – архитектура – логика – структура – код
- Г) архитектура – логика – требования – структура – код

Ответ:

Задание 16.1.

Как называется принцип, согласно которому класс должен иметь одну и только одну причину для изменения?

Ответ:

Задание 17.1.

Какой паттерн проектирования облегчает добавление новых классов без изменения существующего кода?

Ответ:

Задание 18.1.

Какие аспекты безопасности должны учитываться при проектировании архитектуры программного обеспечения, особенно в контексте облачных решений?

Ответ:

Задание 19.1.

Как управлять техническим долгом в архитектуре программного обеспечения и какие стратегии могут помочь в его минимизации?

Ответ:

Задание 20.1.

Как современные технологии, такие как контейнеризация и оркестрация (например, Docker и Kubernetes), влияют на подход к архитектуре программного обеспечения?

Ответ:

Вариант 2

Задание 1.2.

Что такое микро сервисная архитектура?

- А) Архитектура, основанная на монолитном подходе

Б) Архитектура, где приложение разбивается на множество мелких, независимых сервисов

В) Архитектура, использующая только серверные технологии

Г) Архитектура, требующая единого интерфейса для всех клиентов

Ответ:

Задание 2.2.

Какой из следующих подходов является примером архитектурного паттерна?

А) MVC

Б) Agile

Вс) Waterfall

Г) DevOps

Ответ:

Задание 3.2.

Какой из следующих эффектов часто является результатом неправильной архитектуры программного обеспечения?

А) Улучшение производительности

Б) Упрощение кода

В) Увеличение сложности и затрат на поддержку

Г) Легкость в масштабировании

Ответ:

Задание 4.2.

Какой из следующих типов диаграмм используется для визуального представления архитектуры программного обеспечения?

А) Диаграмма классов

Б) Диаграмма последовательностей

В) Диаграмма компонентов

Ответ:

Задание 5.2.

Какой из предложенных принципов помогает уменьшить зависимость между компонентами архитектуры системы?

А) Принцип единственной ответственности (SRP)

Б) Принцип открытости/закрытости (ОСР)

В) Принцип подстановки Лисков (LSP)

Г) Принцип инверсии зависимостей (DIP)

Ответ:

Задание 6.2.

Какой принцип проектирования программного обеспечения гласит, что изменения в одном модуле не должны влиять на другие модули?

Ответ:

Задание 7.2.

Что является основным артефактом, описывающим архитектуру программного обеспечения?

Ответ:

Задание 8.2.

Какой метод разработки программного обеспечения включает в себя быстрое создание рабочих прототипов?

Ответ:

Задание 9.2.

В чем главные отличия между микросервисной и монолитной архитектурой, и какие факторы следует учитывать при выборе одной из них для конкретного проекта?

Ответ:

Задание 10.2.

Как архитектура программного обеспечения может интегрироваться с методологиями разработки, такими как Agile или DevOps, и какие преимущества это приносит?

Ответ:

Задание 11.2.

Что из перечисленного НЕ относится к принципам SOLID?

- А) Single Responsibility Principle (SRP)
- Б) Open/Closed Principle (OCP)
- В) Liskov Substitution Principle (LSP)
- Г) Dependency Inversion Principle (DIP)
- Д) Resource Utilization Principle (RUP)

Ответ:

Задание 12.2.

Одно из основных архитектурных правил гласит:

- А) Высокое зацепление (coupling), низкая связность (cohesion)
- Б) Низкое зацепление (coupling), высокая связность (cohesion)
- В) Высокое зацепление (coupling), высокая связность (cohesion)
- Г) Низкое зацепление (coupling), низкая связность (cohesion)

Ответ:

Задание 13.2.

Какие из следующих методов разработки программного обеспечения помогают в документировании архитектуры? (выберите все подходящие)

- А) UML-диаграммы
- Б) Архитектурные описания на естественном языке
- В) Float-диаграммы
- Г) Модели данных

Ответ:

Задание 14.2.

Какие проблемы решает Архитектура?

- А) Уменьшает time to market
- Б) Позволяет наращивать темп разработки
- В) Уменьшает сложность разработки
- Г) Упрощает интеграцию разных компонент системы

Ответ:

Задание 15.2.

На каком уровне архитектуры используется архитектурный шаблон MVC?

- А) На уровне предприятия
- Б) На уровне приложения
- В) На уровне кода

Ответ:

Задание 16.2.

Какой подход к архитектуре программного обеспечения использует события как основу для взаимодействия компонентов?

Ответ:

Задание 17.2.

Какой стиль архитектуры предполагает разделение приложения на отдельные модули с четкими интерфейсами?

Ответ:

Задание 18.2.

Какие метрики и показатели могут использоваться для оценки качества архитектуры программного обеспечения?

Ответ:

Задание 19.2.

В чем заключается важность документирования архитектурных решений, и как это может повлиять на командную работу и передачу знаний?

Ответ:

Задание 20.2.

Какие вызовы и возможности представляют собой распределенные системы в контексте архитектуры и дизайна программного обеспечения?

Ответ:

Ключи к оцениванию заданий

Вариант 1

№ задания	Верный ответ	Критерии оценивания
1.1	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
2.1	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
3.1	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
4.1	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
5.1	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
6.1	Микросервисная архитектура	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
7.1	Модульное программирование	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
8.1	Слоевая архитектура	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
9.1	Ключевые принципы проектирования архитектуры программного обеспечения и их влияние на масштабируемость и поддерживаемость: - Разделение ответственности - Инкапсуляция - Соблюдение стандартов и интерфейсов - Устойчивость к изменениям	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
10.1	Роль паттернов проектирования в разработке программного обеспечения: - Паттерны проектирования предлагают проверенные решения распространенных проблем, позволяя разработчикам избегать ошибок и упрощать код. - Наиболее востребованные паттерны: Singleton, Factory, Observer, Strategy, Decorator. Они помогают в организации кода, улучшая его читаемость и поддержку.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
11.1	Г	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
12.1	А	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
13.1	А, В, Г	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
14.1	А, Б, В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
15.1	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
16.1	Единственная ответственность	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
17.1	Открытость/закрытость	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
18.1	Аспекты безопасности при проектировании архитектуры:	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие

	- Аутентификация и авторизация: необходимо эффективно управлять доступом пользователей. - Шифрование данных: данные должны быть защищены как в покое, так и в передаче. - Мониторинг и аудит: встроенные средства для отслеживания действий пользователей и системных событий. - Защита от уязвимостей: применение практик безопасного кодирования и регулярное обновление систем	
19.1	Управление техническим долгом и стратегии минимизации: - Регулярные рефакторинг и ревизия кода: периодическое улучшение кода с фокусом на устранение долгов. - Документирование и код-ревью: создание и поддержание документации и регулярные обсуждения кода с командой. - Определение приоритетов: фокусироваться на наиболее критичных участках кода для списания долга.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
20.1	Влияние контейнеризации и оркестрации на подход к архитектуре: - Контейнеры (например, Docker) позволяют легко развертывать и масштабировать приложения. - Оркестрация (например, Kubernetes) автоматизирует управление контейнерами, обеспечивая надежность и высокую доступность. - Упрощает внедрение микросервисной архитектуры и гибкое масштабирование.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие

Вариант 2

№ задания	Верный ответ	Критерии оценивания
1.2	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
2.2	А	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
3.2	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
4.2	В	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
5.2	Г	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
6.2	Принцип слабой связанности	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
7.2	Архитектурное описание	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
8.2	Прототипирование	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
9.2	Монолитная архитектура: все компоненты системы объединены в одно приложение, что упрощает развертывание, но затрудняет масштабируемость и обновления. - Микросервисная архитектура: система разбита на независимые сервисы, что улучшает масштабируемость и позволяет командам работать параллельно, но требует более сложного управления зависимостями и инфраструктурой. - Факторы выбора: размер и сложность проекта, команда разработчиков, требования к масштабированию и устойчивости приложений.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
10.2	Интеграция архитектуры с методологиями разработки (Agile, DevOps): - Архитектура должна быть гибкой и адаптируемой, что соответствует принципам Agile. - В DevOps важно ориентироваться на автоматизацию тестирования и развертывания; архитектура должна это поддерживать. - Преимущества: уменьшение времени на выпуск обновлений, быстрая обратная связь и улучшение взаимодействия между командами разработки и эксплуатации.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
11.2	Д	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
12.2	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
13.2	А, Б, Г	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
14.2	Б, В, Г	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
15.2	Б	1 балл – полное совпадение с верным ответом 0 баллов – неверный ответ или его отсутствие
16.2	Событийно-ориентированная архитектура	2 балла – полное совпадение с верным ответом

		1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
17.2	Модульная архитектура	2 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
18.2	Метрики и показатели для оценки качества архитектуры: - Устойчивость к изменениям: оценка сложности внесения изменений в архитектуру. - Производительность: время отклика системы и её способность справляться с нагрузками. - Доступность: процент времени, когда система доступна для пользователей. - Легкость в использовании: насколько просто новым разработчикам встраиваться в проект и понимать архитектуру.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
19.2	Важность документирования архитектурных решений: - Улучшает понимание архитектуры новой командой и служит справочным материалом. - Облегчает процесс обучения новых сотрудников. - Упрощает обсуждение и принятие решений в команде о будущем развитии системы.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие
20.2	Вызовы и возможности распределенных систем: - Вызовы: управление состоянием, сложность общения между сервисами, проблемы с задержкой и согласованностью данных. - Возможности: масштабируемость, устойчивость к сбоям, возможность использования различных технологий для разных сервисов, что обогащает архитектуру и снижает риски.	3 балла – полное совпадение с верным ответом 1 балл – правильный ответ, без обоснования 0 баллов – неверный ответ или его отсутствие