



**САМАРСКИЙ
ПОЛИТЕХ**
Опoрный университет

МИНОБРНАУКИ РОССИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
«Самарский государственный технический университет»
(ФГБОУ ВО «СамГТУ»)



УТВЕРЖДАЮ:

Первый проректор-проректор по
учебной работе

/ Д.Е. Овчинников

25 " 09 20 26 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Б1.В.02.01 «Объектно-ориентированное программирование»

Код и направление подготовки (специальность)	02.03.03 Математическое обеспечение и администрирование информационных систем
Направленность (профиль)	Анализ данных в бизнес-аналитике
Квалификация	Бакалавр
Форма обучения	Очная
Год начала подготовки	2026
Институт / факультет	Институт инженерно-экономического и гуманитарного образования
Выпускающая кафедра	кафедра "Высшая математика"
Кафедра-разработчик	кафедра "Высшая математика"
Объем дисциплины, ч. / з.е.	180 / 5
Форма контроля (промежуточная аттестация)	Зачет с оценкой, Экзамен

Содержание

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы	4
2. Место дисциплины (модуля) в структуре образовательной программы	5
3. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся	6
4. Содержание дисциплины (модуля), структурированное по темам (разделам), с указанием отведенного на них количества академических часов и видов учебных занятий	7
4.1 Содержание лекционных занятий	7
4.2 Содержание лабораторных занятий	9
4.3 Содержание практических занятий	11
4.4. Содержание самостоятельной работы	11
5. Перечень учебной литературы и учебно-методического обеспечения по дисциплине (модулю)	12
6. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения	13
7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», профессиональных баз данных, информационно-справочных систем	13
8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)	13
9. Методические материалы	14
10. Фонд оценочных средств по дисциплине (модулю)	15

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения образовательной программы

Наименование категории (группы) компетенций	Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения (знать, уметь, владеть, соотнесенные с индикаторами достижения компетенции)
Профессиональные компетенции			
Не предусмотрено	ПК-1 Способность использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования	ПК-1.1 Знать: современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	Знать принципы объектно-ориентированного программирования и современные методы представления алгоритмов и моделей в виде взаимодействующих классов и программных компонентов
		ПК-1.2 Уметь: разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Уметь проектировать объектную модель предметной области и реализовывать алгоритмы обработки данных средствами классов, композиции, наследования, полиморфизма и протоколов Python
		ПК-1.3 Владеть: практическим опытом разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования	Владеть практическим опытом разработки, тестирования и рефакторинга объектно-ориентированной программы, реализующей алгоритм модели или прикладной сценарий анализа данных
Универсальные компетенции			

Системное и критическое мышление	УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Знать: основы системного и критического анализа информации	Знать основы системного и критического анализа требований, объектов, связей, ограничений и альтернатив при построении программной модели предметной области
		УК-1.2 Уметь: разрабатывать стратегию действий, принимать конкретные решения для ее реализации	Уметь формировать стратегию разработки объектно-ориентированной программы, выбирать структуру классов и принимать обоснованные решения по реализации и проверке компонентов
		УК-1.3 Владеть: методиками постановки цели, определения способов ее достижения, разработки стратегий действий	Владеть методиками постановки цели программного проекта, декомпозиции задачи, определения последовательности разработки, тестирования и улучшения объектной модели

2. Место дисциплины (модуля) в структуре образовательной программы

Место дисциплины (модуля) в структуре образовательной программы: **часть, формируемая участниками образовательных отношений**

Код компетенции	Предшествующие дисциплины	Параллельно осваиваемые дисциплины	Последующие дисциплины
ПК-1			Введение в цифровую экономику; ИИ-ассистенты; Менеджмент в IT-отрасли; Методология разработки DevOps; Подготовка к процедуре защиты и защита выпускной квалификационной работы; Развертывание и жизненный цикл программного обеспечения; Цифровая трансформация бизнеса; Экономическое развитие и инновационная экономика; Языки программирования

УК-1	Учебная практика: проектная практика	Учебная практика: проектная практика; Финансовый менеджмент	Анализ данных и машинное обучение; Большие данные; Введение в цифровую экономику; Визуализация данных; Инженерия данных; Компьютерное зрение; Методология разработки DevOps; Моделирование бизнес-процессов; Нейронные сети; Основы работы с NoSQL-базами данных; Основы экономического анализа; Подготовка к процедуре защиты и защита выпускной квалификационной работы; Производственная практика: научно-исследовательская работа; Производственная практика: преддипломная практика; Развертывание и жизненный цикл программного обеспечения; Разработка и принятие управленческих решений; Стратегическое планирование; Технологии интернета вещей; Учебная практика: научно-исследовательская работа (получение первичных навыков научно-исследовательской работы); Учебная практика: проектная практика; Цифровая трансформация бизнеса; Экономика предприятия; Экономическое развитие и инновационная экономика; Языки программирования
------	--------------------------------------	---	--

3. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Вид учебной работы	Всего часов / часов в электронной форме	2 семестр часов / часов в электронной форме	3 семестр часов / часов в электронной форме
Аудиторная контактная работа (всего), в том числе:	128	64	64
Лабораторные работы	64	32	32
Лекции	64	32	32
Внеаудиторная контактная работа, КСР	4	0	4
Самостоятельная работа (всего), в том числе:	32	28	4
подготовка к зачету	2	0	2
подготовка к лабораторным работам	22	20	2

подготовка к экзамену	8	8	0
Контроль	16	16	0
Итого: час	180	108	72
Итого: з.е.	5	3	2

4. Содержание дисциплины (модуля), структурированное по темам (разделам), с указанием отведенного на них количества академических часов и видов учебных занятий

№ раздела	Наименование раздела дисциплины	Виды учебной нагрузки и их трудоемкость, часы				
		ЛЗ	ЛР	ПЗ	СРС	Всего часов
1	Основы объектной модели и инкапсуляция	16	16	0	14	46
2	Отношения между объектами, наследование и полиморфизм	16	16	0	14	46
3	Проектирование модульных и расширяемых программ	16	16	0	2	34
4	Качество объектно-ориентированных программ и прикладной проект	16	16	0	2	34
	КСР	0	0	0	0	4
	Контроль	0	0	0	0	16
	Итого	64	64	0	32	180

4.1 Содержание лекционных занятий

№ занятия	Наименование раздела	Тема лекции	Содержание лекции (перечень дидактических единиц: рассматриваемых подтем, вопросов)	Количество часов / часов в электронной форме
2 семестр				
1	Основы объектной модели и инкапсуляция	Объектно-ориентированный подход	Назначение объектного подхода. Сравнение с процедурным подходом. Объектная модель предметной области.	2
2	Основы объектной модели и инкапсуляция	Объект, класс, состояние и поведение	Идентичность объекта. Атрибуты и методы. Экземпляр и класс. Ответственность объекта.	2
3	Основы объектной модели и инкапсуляция	Определение классов в Python	Синтаксис класса. Создание экземпляров. Параметр self. Вызов методов.	2
4	Основы объектной модели и инкапсуляция	Инициализация и жизненный цикл объекта	Метод __init__. Начальное состояние. Допустимые и недопустимые состояния.	2
5	Основы объектной модели и инкапсуляция	Атрибуты экземпляра и класса	Пространства имен. Поиск атрибутов. Общие и индивидуальные данные объектов.	2
6	Основы объектной модели и инкапсуляция	Методы экземпляра, класса и статические методы	Назначение видов методов. Фабричные способы создания объектов. Границы применения.	2

7	Основы объектной модели и инкапсуляция	Инкапсуляция и управляемый доступ	Соглашения доступа. Свойства. Проверка значений. Защита согласованности состояния.	2
8	Основы объектной модели и инкапсуляция	Контракты и инварианты класса	Предусловия и постусловия. Инварианты. Распределение ответственности. Проектирование интерфейса класса.	2
9	Отношения между объектами, наследование и полиморфизм	Связи между объектами	Ассоциация, агрегация и композиция. Мощность связей. Владение жизненным циклом объектов.	2
10	Отношения между объектами, наследование и полиморфизм	Композиция как средство построения модели	Делегирование поведения. Слабая связанность. Замена компонентов.	2
11	Отношения между объектами, наследование и полиморфизм	Наследование классов	Базовые и производные классы. Расширение поведения. Отношение является.	2
12	Отношения между объектами, наследование и полиморфизм	Переопределение методов	Замена реализации. Вызов базового поведения через super. Подстановка объектов.	2
13	Отношения между объектами, наследование и полиморфизм	Полиморфизм	Единый интерфейс для разных реализаций. Утиная типизация. Снижение зависимости от конкретных классов.	2
14	Отношения между объектами, наследование и полиморфизм	Абстракции и протоколы	Абстрактные базовые классы. Формальные и неформальные интерфейсы. Контроль обязательных операций.	2
15	Отношения между объектами, наследование и полиморфизм	Специальные методы Python	Представление, сравнение, длина, доступ к элементам и вызываемые объекты. Протоколы языка.	2
16	Отношения между объектами, наследование и полиморфизм	Классы данных и объекты-значения	Dataclass. Равенство по значению. Неизменяемые объекты. Моделирование записей предметной области.	2
Итого за семестр:				32
3 семестр				
17	Проектирование модульных и расширяемых программ	Исключения в объектной модели	Иерархии исключений. Проверка ошибок на границах класса. Сохранение корректного состояния.	2
18	Проектирование модульных и расширяемых программ	Итерируемые объекты и итераторы	Протокол итерации. Пользовательские коллекции. Последовательная обработка объектов.	2
19	Проектирование модульных и расширяемых программ	Генераторы и отложенная обработка	Генераторные функции. Экономная обработка последовательностей. Взаимодействие с объектными моделями.	2
20	Проектирование модульных и расширяемых программ	Аннотации типов и обобщенные структуры	Типы параметров и результатов. Коллекции объектов. Протоколы и параметризация типов.	2
21	Проектирование модульных и расширяемых программ	Модули и пакеты	Разделение ответственности между модулями. Публичный интерфейс пакета. Импорт и организация проекта.	2
22	Проектирование модульных и расширяемых программ	Принципы проектирования классов	Единственная ответственность. Открытость к расширению. Подстановка. Разделение интерфейсов.	2
23	Проектирование модульных и расширяемых программ	Управление зависимостями	Передача зависимостей. Работа через абстракции. Тестируемость и заменяемость компонентов.	2
24	Проектирование модульных и расширяемых программ	Базовые шаблоны проектирования	Стратегия, фабричный метод и адаптер. Выбор шаблона по проблеме. Ограничения шаблонов.	2

25	Качество объектно-ориентированных программ и прикладной проект	Хранение и восстановление состояния объектов	Сериализация. Форматы обмена. Преобразование между объектами и данными. Ограничения безопасности.	2
26	Качество объектно-ориентированных программ и прикладной проект	Модульное тестирование классов	Тестовый случай. Подготовка данных. Проверка результата. Тестирование состояния и поведения.	2
27	Качество объектно-ориентированных программ и прикладной проект	Изоляция зависимостей при тестировании	Тестовые заместители. Контроль внешних взаимодействий. Повторяемость тестов.	2
28	Качество объектно-ориентированных программ и прикладной проект	Документирование и наблюдаемость	Строки документации. Понятные интерфейсы. Журналирование событий и ошибок	2
29	Качество объектно-ориентированных программ и прикладной проект	Рефакторинг объектной модели	Признаки избыточной связанности. Дублирование. Большой класс. Извлечение класса и метода.	2
30	Качество объектно-ориентированных программ и прикладной проект	Моделирование предметной области	Сущности, объекты-значения, правила и сервисы. Перевод требований в объектную модель.	2
31	Качество объектно-ориентированных программ и прикладной проект	Архитектура небольшого приложения	Слои представления, прикладной логики и доступа к данным. Потoki данных и границы компонентов.	2
32	Качество объектно-ориентированных программ и прикладной проект	Развитие и сопровождение программного проекта	Версионирование изменений. Обратная совместимость. Анализ качества. Представление результатов проекта.	2
Итого за семестр:				32
Итого:				64

4.2 Содержание лабораторных занятий

№ занятия	Наименование раздела	Тема лабораторного занятия	Содержание лабораторного занятия (перечень дидактических единиц: рассматриваемых подтем, вопросов)	Количество часов / часов в электронной форме
2 семестр				
1	Основы объектной модели и инкапсуляция	Проектирование класса предметной области	Определение состояния, поведения и ответственности класса для объекта бизнес-процесса.	2
2	Основы объектной модели и инкапсуляция	Проектирование класса предметной области	Определение состояния, поведения и ответственности класса для объекта бизнес-процесса.	2
3	Основы объектной модели и инкапсуляция	Инициализация и проверка состояния объекта	Реализация конструктора, проверок входных данных и инвариантов.	2
4	Основы объектной модели и инкапсуляция	Инициализация и проверка состояния объекта	Реализация конструктора, проверок входных данных и инвариантов.	2
5	Основы объектной модели и инкапсуляция	Атрибуты и виды методов	Разделение данных экземпляра и класса. Применение методов экземпляра, класса и статических методов.	2
6	Основы объектной модели и инкапсуляция	Атрибуты и виды методов	Разделение данных экземпляра и класса. Применение методов экземпляра, класса и статических методов.	2

7	Основы объектной модели и инкапсуляция	Инкапсуляция и свойства	Создание управляемого интерфейса доступа к данным и исключение некорректных состояний.	2
8	Основы объектной модели и инкапсуляция	Инкапсуляция и свойства	Создание управляемого интерфейса доступа к данным и исключение некорректных состояний.	2
9	Основы объектной модели и инкапсуляция	Композиция объектов	Построение модели из взаимодействующих объектов без избыточного наследования.	2
10	Основы объектной модели и инкапсуляция	Композиция объектов	Построение модели из взаимодействующих объектов без избыточного наследования.	2
12	Отношения между объектами, наследование и полиморфизм	Полиморфизм и абстрактный интерфейс	Реализация взаимозаменяемых стратегий обработки данных через общий интерфейс.	2
13	Отношения между объектами, наследование и полиморфизм	Классы данных и специальные методы	Создание объектов-значений. Реализация представления, сравнения и сортировки.	2
14	Отношения между объектами, наследование и полиморфизм	Классы данных и специальные методы	Создание объектов-значений. Реализация представления, сравнения и сортировки.	2
19	Проектирование модульных и расширяемых программ	Модульная организация и аннотации типов	Разделение программы на пакеты. Определение публичных интерфейсов и типов данных.	2
31	Отношения между объектами, наследование и полиморфизм	Иерархия классов	Выделение общего поведения. Реализация базового и производных классов.	2
32	Отношения между объектами, наследование и полиморфизм	Иерархия классов	Выделение общего поведения. Реализация базового и производных классов.	2
33	Отношения между объектами, наследование и полиморфизм	Полиморфизм и абстрактный интерфейс	Реализация взаимозаменяемых стратегий обработки данных через общий интерфейс.	2
Итого за семестр:				34
3 семестр				
15	Проектирование модульных и расширяемых программ	Пользовательская коллекция и итератор	Реализация контейнера объектов и последовательного обхода элементов.	2
16	Проектирование модульных и расширяемых программ	Пользовательская коллекция и итератор	Реализация контейнера объектов и последовательного обхода элементов.	2
17	Проектирование модульных и расширяемых программ	Иерархия исключений приложения	Обработка ошибок предметной области и сохранение согласованности объектов.	2
18	Проектирование модульных и расширяемых программ	Иерархия исключений приложения	Обработка ошибок предметной области и сохранение согласованности объектов.	2
20	Проектирование модульных и расширяемых программ	Модульная организация и аннотации типов	Разделение программы на пакеты. Определение публичных интерфейсов и типов данных.	2
21	Проектирование модульных и расширяемых программ	Применение шаблонов проектирования	Реализация стратегии обработки и фабричного создания объектов для расширяемого сценария.	2
22	Проектирование модульных и расширяемых программ	Применение шаблонов проектирования	Реализация стратегии обработки и фабричного создания объектов для расширяемого сценария.	2
23	Качество объектно-ориентированных программ и прикладной проект	Сохранение объектной модели	Преобразование объектов в формат хранения и обратное восстановление.	2

24	Качество объектно-ориентированных программ и прикладной проект	Сохранение объектной модели	Преобразование объектов в формат хранения и обратное восстановление.	2
25	Качество объектно-ориентированных программ и прикладной проект	Модульное тестирование классов	Создание набора тестов для нормальных, граничных и ошибочных сценариев.	2
26	Качество объектно-ориентированных программ и прикладной проект	Модульное тестирование классов	Создание набора тестов для нормальных, граничных и ошибочных сценариев.	2
27	Качество объектно-ориентированных программ и прикладной проект	Рефакторинг существующей программы	Выявление проблем структуры и улучшение модели без изменения внешнего поведения.	2
28	Качество объектно-ориентированных программ и прикладной проект	Рефакторинг существующей программы	Выявление проблем структуры и улучшение модели без изменения внешнего поведения.	2
29	Качество объектно-ориентированных программ и прикладной проект	Прикладной объектно-ориентированный проект	Создание небольшого приложения для анализа показателей, учета объектов или поддержки бизнес-решения.	2
30	Качество объектно-ориентированных программ и прикладной проект	Прикладной объектно-ориентированный проект	Создание небольшого приложения для анализа показателей, учета объектов или поддержки бизнес-решения.	2
Итого за семестр:				30
Итого:				64

4.3 Содержание практических занятий

Учебные занятия не реализуются.

4.4. Содержание самостоятельной работы

Наименование раздела	Вид самостоятельной работы	Содержание самостоятельной работы (перечень дидактических единиц: рассматриваемых подтем, вопросов)	Количество часов
2 семестр			
Основы объектной модели и инкапсуляция	Подготовка к лекционным и лабораторным занятиям	Объект, класс, состояние, поведение, инкапсуляция и контракты. Проектирование интерфейса класса и тестовых примеров.	10
Отношения между объектами, наследование и полиморфизм	Подготовка к лекционным и лабораторным занятиям	Композиция и наследование. Полиморфизм, абстракции, протоколы, специальные методы и классы данных.	10
Основы объектной модели и инкапсуляция	Подготовка к экзамену	Изучение теоретического материала	4
Отношения между объектами, наследование и полиморфизм	Подготовка к экзамену	Изучение теоретического материала	4
Итого за семестр:			28
3 семестр			

Проектирование модульных и расширяемых программ	Подготовка к лекционным и лабораторным занятиям	Модули и пакеты. Зависимости. Аннотации типов. Шаблоны проектирования.	1
Качество объектно-ориентированных программ и прикладной проект	Подготовка к лекционным и лабораторным занятиям	Модель предметной области. Тестирование, рефакторинг, документирование и демонстрация результата.	1
Проектирование модульных и расширяемых программ	Подготовка к зачету	Изучение теоретического материала	1
Качество объектно-ориентированных программ и прикладной проект	Подготовка к зачету	Изучение теоретического материала	1
Итого за семестр:			4
Итого:			32

5. Перечень учебной литературы и учебно-методического обеспечения по дисциплине (модулю)

№ п/п	Библиографическое описание	Ресурс НТБ СамГТУ (ЭБС СамГТУ, IPRbooks и т.д.)
Основная литература		
1	Python — от базовых конструкций до обработки данных: Учебное пособие / Агалаков А.А., Дементьева К.И., Профобразование: 2025.- Режим доступа: https://elib.samgtu.ru/getinfo?uid=els_samgtu iprbooks 150102	Электронный ресурс
2	Объектно-ориентированное программирование: Учебник / Маляров А.Н., Профобразование: 2022.- Режим доступа: https://elib.samgtu.ru/getinfo?uid=els_samgtu iprbooks 132418	Электронный ресурс
Дополнительная литература		
3	Основы алгоритмизации и объектно-ориентированного программирования: Учебное пособие / Куренкова Т.В., Светозарова Г.И., Издательский Дом МИСиС: 2011.- Режим доступа: https://elib.samgtu.ru/getinfo?uid=els_samgtu iprbooks 98879	Электронный ресурс
4	Основы прикладного программирования на языке Python: Учебное пособие / Шустров А.С., Ай Пи Ар Медиа: 2026.- Режим доступа: https://elib.samgtu.ru/getinfo?uid=els_samgtu iprbooks 154932	Электронный ресурс
5	Языки программирования. Концепции и принципы: Учебное пособие / Кауфман В.Ш., Профобразование: 2024.- Режим доступа: https://elib.samgtu.ru/getinfo?uid=els_samgtu iprbooks 145939	Электронный ресурс

Доступ обучающихся к ЭР НТБ СамГТУ (elib.samgtu.ru) осуществляется посредством электронной информационной образовательной среды университета и сайта НТБ СамГТУ по логину и паролю.

6. Перечень информационных технологий, используемых при осуществлении

образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения

При проведении лекционных занятий используется мультимедийное оборудование.

Организовано взаимодействие обучающегося и преподавателя с использованием электронной ин-формационной образовательной среды университета.

№ п/п	Наименование	Производитель	Способ распространения
1	Python	Python Software Foundation (Зарубежный)	Свободно распространяемое
2	Visual Studio Code	Microsoft Corporation (Зарубежный)	Свободно распространяемое
3	Git	Software Freedom Conservancy (Зарубежный)	Свободно распространяемое
4	Альт Рабочая станция	ООО «Базальт СПО» (Отечественный)	Лицензионное

7. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», профессиональных баз данных, информационно-справочных систем

№ п/п	Наименование	Краткое описание	Режим доступа
1	Библиотека учебно-методической литературы системы "Единое окно"	http://window.edu.ru/	Ресурсы открытого доступа
2	Научная электронная библиотека	http://elibrary.ru/	Ресурсы открытого доступа
3	Научная электронная библиотека «Киберленинка»	https://cyberleninka.ru	Ресурсы открытого доступа

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)

Лекционные занятия

Аудитория, оснащенная презентационной техникой: проектор, экран, компьютер или ноутбук. Аудитория оснащенная учебной мебелью.

Лабораторные занятия

Компьютерный класс с доступом в локальную сеть и Интернет, оснащенный компьютерами, интерпретатором Python, редактором программного кода и средствами демонстрации учебных материалов.

Самостоятельная работа

Компьютерный класс с доступом в Интернет и электронную информационно-образовательную среду СамГТУ. Читальный зал НТБ СамГТУ. Ресурсы УИТ СамГТУ.

9. Методические материалы

Методические рекомендации при работе на лекции

До лекции студент должен просмотреть учебно-методическую и научную литературу по теме лекции с тем, чтобы иметь представление о проблемах, которые будут разбираться в лекции.

Перед началом лекции обучающимся сообщается тема лекции, план, вопросы, подлежащие рассмотрению, доводятся основные литературные источники. Весь учебный материал, сообщаемый преподавателем, должен не просто прослушиваться. Он должен быть активно воспринят, т.е. услышан, осмыслен, понят, зафиксирован на бумаге и закреплён в памяти. Приступая к слушанию нового учебного материала, полезно мысленно установить его связь с ранее изученным. Следя за техникой чтения лекции (акцент на существенном, повышение тона, изменение ритма, пауза и т.п.), необходимо вслед за преподавателем уметь выделять основные категории, законы и определять их содержание, проблемы, предполагать их возможные решения, доказательства и выводы. Осуществляя такую работу, можно значительно облегчить себе понимание учебного материала, его конспектирование и дальнейшее изучение.

Конспектирование лекции позволяет обработать, систематизировать и лучше сохранить полученную информацию с тем, чтобы в будущем можно было восстановить в памяти основные, содержательные моменты. Типичная ошибка, совершаемая обучающимся, дословное конспектирование речи преподавателя. Как правило, при записи «слово в слово» не остается времени на обдумывание, анализ и синтез информации. Отбирая нужную информацию, главные мысли, проблемы, решения и выводы, необходимо сокращать текст, строить его таким образом, чтобы потом можно было легко в нем разобраться. Желательно оставить в рабочих конспектах поля, на которых можно будет делать пометки из рекомендованной литературы, дополняющие материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений. С окончанием лекции работа над конспектом не может считаться завершённой. Нужно еще восстановить отдельные места, проверить, все ли понятно, уточнить что-то на консультации и т.п. с тем, чтобы конспект мог быть использован в процессе подготовки к практическим занятиям, зачету, экзамену. Конспект лекции – незаменимый учебный документ, необходимый для самостоятельной работы.

Методические рекомендации при работе на лабораторном занятии

Проведение лабораторной работы делится на две условные части: теоретическую и практическую.

Необходимыми структурными элементами занятия являются проведение лабораторной работы, проверка усвоенного материала, включающая обсуждение теоретических основ выполняемой работы.

Перед лабораторной работой, как правило, проводится технико-теоретический инструктаж по использованию необходимого оборудования. Преподаватель корректирует деятельность обучающегося в процессе выполнения работы (при необходимости). После завершения лабораторной работы подводятся итоги, обсуждаются результаты деятельности.

Возможны следующие формы организации лабораторных работ: фронтальная, групповая и индивидуальная. При фронтальной форме выполняется одна и та же работа (при этом возможны различные варианты заданий). При групповой форме работа выполняется группой (командой). При индивидуальной форме обучающимися выполняются индивидуальные работы.

По каждой лабораторной работе имеются методические указания по их выполнению, включающие необходимый теоретический и практический материал, содержащие элементы и последовательную инструкцию по проведению выбранной работы, индивидуальные варианты заданий, требования и форму отчётности по данной работе.

Методические рекомендации по выполнению самостоятельной работы

Организация самостоятельной работы обучающихся ориентируется на активные методы овладения знаниями, развитие творческих способностей, переход от поточного к индивидуализированному обучению с учетом потребностей и возможностей обучающегося.

Самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения

дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала. Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса.

Самостоятельная работа реализуется:

- непосредственно в процессе аудиторных занятий;
- на лекциях, практических занятиях;
- в контакте с преподавателем вне рамок расписания;
- на консультациях по учебным вопросам, в ходе творческих контактов, при ликвидации задолженностей, при выполнении индивидуальных заданий и т.д.;
- в библиотеке, дома, на кафедре при выполнении обучающимся учебных и практических задач.

Эффективным средством осуществления обучающимся самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к учебным планам, рабочим программам дисциплин (модулей), практик, к изданиям электронных библиотечных систем.

10. Фонд оценочных средств по дисциплине (модулю)

Фонд оценочных средств представлен в приложении № 1.

Приложение 1 к рабочей программе дисциплины
Б1.В.02.01 «Объектно-ориентированное
программирование»

**Фонд оценочных средств
по дисциплине
Б1.В.02.01 «Объектно-ориентированное программирование»**

Код и направление подготовки (специальность)	02.03.03 Математическое обеспечение и администрирование информационных систем
Направленность (профиль)	Анализ данных в бизнес-аналитике
Квалификация	Бакалавр
Форма обучения	Очная
Год начала подготовки	2026
Институт / факультет	Институт инженерно-экономического и гуманитарного образования
Выпускающая кафедра	кафедра "Высшая математика"
Кафедра-разработчик	кафедра "Высшая математика"
Объем дисциплины, ч. / з.е.	180 / 5
Форма контроля (промежуточная аттестация)	Зачет с оценкой, Экзамен

**Перечень планируемых результатов обучения по дисциплине (модулю),
соотнесенных с планируемыми результатами освоения образовательной
программы**

Наименование категории (группы) компетенций	Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения (знать, уметь, владеть, соотнесенные с индикаторами достижения компетенции)
Профессиональные компетенции			
Не предусмотрено	ПК-1 Способность использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования	ПК-1.1 Знать: современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	Знать принципы объектно-ориентированного программирования и современные методы представления алгоритмов и моделей в виде взаимодействующих классов и программных компонентов
		ПК-1.2 Уметь: разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Уметь проектировать объектную модель предметной области и реализовывать алгоритмы обработки данных средствами классов, композиции, наследования, полиморфизма и протоколов Python
		ПК-1.3 Владеть: практическим опытом разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования	Владеть практическим опытом разработки, тестирования и рефакторинга объектно-ориентированной программы, реализующей алгоритм модели или прикладной сценарий анализа данных
Универсальные компетенции			

Системное и критическое мышление	УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1 Знать: основы системного и критического анализа информации	Знать основы системного и критического анализа требований, объектов, связей, ограничений и альтернатив при построении программной модели предметной области
		УК-1.2 Уметь: разрабатывать стратегию действий, принимать конкретные решения для ее реализации	Уметь формировать стратегию разработки объектно-ориентированной программы, выбирать структуру классов и принимать обоснованные решения по реализации и проверке компонентов
		УК-1.3 Владеть: методиками постановки цели, определения способов ее достижения, разработки стратегий действий	Владеть методиками постановки цели программного проекта, декомпозиции задачи, определения последовательности разработки, тестирования и улучшения объектной модели

Матрица соответствия оценочных средств запланированным результатам обучения

Код индикатора достижения компетенции	Результаты обучения	Оценочные средства	Текущий контроль успеваемости	Промежуточная аттестация
Основы объектной модели и инкапсуляция				
ПК-1.1 Знать: современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	Знать принципы объектно-ориентированного программирования и современные методы представления алгоритмов и моделей в виде взаимодействующих классов и программных компонентов	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
ПК-1.2 Уметь: разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Уметь проектировать объектную модель предметной области и реализовывать алгоритмы обработки данных средствами классов, композиции, наследования, полиморфизма и протоколов Python	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да

ПК-1.3 Владеть: практическим опытом разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования	Владеть практическим опытом разработки, тестирования и рефакторинга объектно-ориентированной программы, реализующей алгоритм модели или прикладной сценарий анализа данных	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.1 Знать: основы системного и критического анализа информации	Знать основы системного и критического анализа требований, объектов, связей, ограничений и альтернатив при построении программной модели предметной области	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.2 Уметь: разрабатывать стратегию действий, принимать конкретные решения для ее реализации	Уметь формировать стратегию разработки объектно-ориентированной программы, выбирать структуру классов и принимать обоснованные решения по реализации и проверке компонентов	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.3 Владеть: методиками постановки цели, определения способов ее достижения, разработки стратегий действий	Владеть методиками постановки цели программного проекта, декомпозиции задачи, определения последовательности разработки, тестирования и улучшения объектной модели	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
Отношения между объектами, наследование и полиморфизм				
ПК-1.1 Знать: современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	Знать принципы объектно-ориентированного программирования и современные методы представления алгоритмов и моделей в виде взаимодействующих классов и программных компонентов	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
ПК-1.2 Уметь: разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Уметь проектировать объектную модель предметной области и реализовывать алгоритмы обработки данных средствами классов, композиции, наследования, полиморфизма и протоколов Python	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
ПК-1.3 Владеть: практическим опытом разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования	Владеть практическим опытом разработки, тестирования и рефакторинга объектно-ориентированной программы, реализующей алгоритм модели или прикладной сценарий анализа данных	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.1 Знать: основы системного и критического анализа информации	Знать основы системного и критического анализа требований, объектов, связей, ограничений и альтернатив при построении программной модели предметной области	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да

УК-1.2 Уметь: разрабатывать стратегию действий, принимать конкретные решения для ее реализации	Уметь формировать стратегию разработки объектно-ориентированной программы, выбирать структуру классов и принимать обоснованные решения по реализации и проверке компонентов	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.3 Владеть: методиками постановки цели, определения способов ее достижения, разработки стратегий действий	Владеть методиками постановки цели программного проекта, декомпозиции задачи, определения последовательности разработки, тестирования и улучшения объектной модели	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
Проектирование модульных и расширяемых программ				
ПК-1.1 Знать: современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	Знать принципы объектно- ориентированного программирования и современные методы представления алгоритмов и моделей в виде взаимодействующих классов и программных компонентов	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
ПК-1.2 Уметь: разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Уметь проектировать объектную модель предметной области и реализовывать алгоритмы обработки данных средствами классов, композиции, наследования, полиморфизма и протоколов Python	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
ПК-1.3 Владеть: практическим опытом разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования	Владеть практическим опытом разработки, тестирования и рефакторинга объектно- ориентированной программы, реализующей алгоритм модели или прикладной сценарий анализа данных	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.1 Знать: основы системного и критического анализа информации	Знать основы системного и критического анализа требований, объектов, связей, ограничений и альтернатив при построении программной модели предметной области	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.2 Уметь: разрабатывать стратегию действий, принимать конкретные решения для ее реализации	Уметь формировать стратегию разработки объектно-ориентированной программы, выбирать структуру классов и принимать обоснованные решения по реализации и проверке компонентов	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.3 Владеть: методиками постановки цели, определения способов ее достижения, разработки стратегий действий	Владеть методиками постановки цели программного проекта, декомпозиции задачи, определения последовательности разработки, тестирования и улучшения объектной модели	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
Качество объектно-ориентированных программ и прикладной проект				

ПК-1.1 Знать: современные методы разработки и реализации алгоритмов математических моделей на базе языков и пакетов прикладных программ моделирования	Знать принципы объектно-ориентированного программирования и современные методы представления алгоритмов и моделей в виде взаимодействующих классов и программных компонентов	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
ПК-1.2 Уметь: разрабатывать и реализовывать алгоритмы математических моделей на базе языков и пакетов прикладных программ моделирования	Уметь проектировать объектную модель предметной области и реализовывать алгоритмы обработки данных средствами классов, композиции, наследования, полиморфизма и протоколов Python	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
ПК-1.3 Владеть: практическим опытом разработки и реализации алгоритмов их на базе языков и пакетов прикладных программ моделирования	Владеть практическим опытом разработки, тестирования и рефакторинга объектно-ориентированной программы, реализующей алгоритм модели или прикладной сценарий анализа данных	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.1 Знать: основы системного и критического анализа информации	Знать основы системного и критического анализа требований, объектов, связей, ограничений и альтернатив при построении программной модели предметной области	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.2 Уметь: разрабатывать стратегию действий, принимать конкретные решения для ее реализации	Уметь формировать стратегию разработки объектно-ориентированной программы, выбирать структуру классов и принимать обоснованные решения по реализации и проверке компонентов	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да
УК-1.3 Владеть: методиками постановки цели, определения способов ее достижения, разработки стратегий действий	Владеть методиками постановки цели программного проекта, декомпозиции задачи, определения последовательности разработки, тестирования и улучшения объектной модели	Подготовка к ЛЗ и ЛР, подготовка к экзамену	Да	Да

Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций в ходе освоения образовательной программы

Формы текущего контроля успеваемости Текущий контроль осуществляется по результатам выполнения и защиты лабораторных работ, контрольных заданий и тестирования по материалам дисциплины.

Пример контрольной работы по разделу «Основы объектной модели и инкапсуляция»

1. Разработать класс для представления показателя бизнес-процесса и определить его ответственность.
2. Реализовать инициализацию объекта и проверку допустимости начального состояния.
3. Разделить атрибуты экземпляра и класса, применить методы разных видов.
4. Организовать управляемый доступ к данным объекта с помощью свойств.
5. Определить контракт класса и проверить его инварианты на контрольных примерах.

Пример контрольной работы по разделу «Отношения между объектами, наследование и полиморфизм»

1. Построить модель связанных объектов с использованием композиции.
2. Разработать обоснованную иерархию базового и производных классов.
3. Реализовать переопределение метода и вызов базового поведения.
4. Создать взаимозаменяемые способы расчета через общий интерфейс.
5. Реализовать объект-значение и необходимые специальные методы.

Пример контрольной работы по разделу «Проектирование модульных и расширяемых программ»

1. Реализовать пользовательскую коллекцию и последовательный обход объектов.
2. Создать иерархию исключений предметной области.
3. Разделить приложение на модули и определить публичные интерфейсы.
4. Передать зависимость через абстракцию и проверить заменяемость реализации.
5. Применить шаблон проектирования по заданному сценарию изменения.

Пример контрольной работы по разделу «Качество объектно-ориентированных программ и прикладной проект»

1. Организовать сохранение и восстановление объектной модели.
2. Подготовить модульные тесты нормальных, граничных и ошибочных сценариев.
3. Изолировать внешнюю зависимость с помощью тестового заместителя.
4. Выполнить рефакторинг класса без изменения внешнего поведения.
5. Разработать и представить прикладной объектно-ориентированный проект.

Семестр 2 Форма промежуточной аттестации — экзамен. Экзамен проводится в письменно-устной форме. Билет включает два теоретических вопроса и одно практическое задание. Результаты выполненных и защищенных лабораторных работ учитываются при промежуточной аттестации.

Федеральное государственное бюджетное образовательное учреждение высшего образования
Самарский государственный технический университет
ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1
по дисциплине Б1.В.02.01 «Объектно-ориентированное программирование»
Направление подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем
1. Объектно-ориентированный и процедурный подходы. Области применения.
2. Переопределение методов и функция super.
3. Практическое задание. Реализовать класс Observation для наблюдения с датой, именем показателя и значением. Запретить пустое имя и недопустимое значение.

Вопросы к экзамену (2 семестр)

1. Объектно-ориентированный и процедурный подходы. Области применения.
2. Объект, класс, экземпляр, идентичность, состояние и поведение.
3. Определение класса и создание объектов в Python.
4. Инициализация объекта и назначение метода `__init__`.
5. Атрибуты экземпляра и атрибуты класса.
6. Методы экземпляра, методы класса и статические методы.
7. Инкапсуляция и соглашения об ограничении доступа в Python.
8. Свойства и проверка корректности состояния объекта.
9. Контракты, предусловия, постусловия и инварианты класса.
10. Ассоциация, агрегация и композиция объектов.
11. Композиция и делегирование поведения.
12. Наследование и критерии его обоснованного применения.
13. Переопределение методов и функция super.
14. Принцип подстановки объектов производного класса.
15. Полиморфизм и утиная типизация.
16. Абстрактные базовые классы и интерфейсы.
17. Специальные методы модели данных Python.
18. Объекты-значения и классы данных.
19. Выбор между композицией и наследованием.
20. Системный анализ предметной области перед построением классов.
21. Выделение ответственности классов по требованиям задачи.
22. Критерии качества интерфейса класса.
23. Анализ альтернатив объектной модели.
24. Проверка объектной модели на контрольных примерах.

Типовые практические задания к экзамену

1. Реализовать класс Observation для наблюдения с датой, именем показателя и значением. Запретить пустое имя и недопустимое значение.
2. Реализовать класс Dataset, который хранит наблюдения, вычисляет среднее значение и поддерживает перебор элементов.
3. Построить модель клиента и заказов через композицию. Рассчитать общую стоимость и число завершенных заказов.

4. Реализовать иерархию способов расчета скидки и полиморфное применение выбранной стратегии.
5. Создать абстрактный класс преобразователя данных и реализации нормализации и масштабирования.
6. Реализовать объект-значение Period и специальные методы сравнения периодов.
7. Найти нарушение инкапсуляции в предложенном классе и выполнить рефакторинг.
8. Сравнить две модели одной предметной области и выбрать вариант с меньшей связанностью.

Семестр 3 Форма промежуточной аттестации — зачет. Зачет проводится в письменно-устной форме. Билет включает два теоретических вопроса и одно практическое задание. Результаты выполненных и защищенных лабораторных работ учитываются при промежуточной аттестации.

Федеральное государственное бюджетное образовательное учреждение высшего образования
Самарский государственный технический университет
ЗАЧЕТНЫЙ БИЛЕТ № 1
по дисциплине Б1.В.02.01 «Объектно-ориентированное программирование»
Направление подготовки 02.03.03 Математическое обеспечение и администрирование информационных систем
1. Исключения предметной области и обеспечение корректного состояния объектов.
2. Шаблон «Стратегия».
3. Практическое задание. Реализовать итератор для коллекции временных рядов с фильтрацией по периоду.

Вопросы к зачету (3 семестр)

1. Исключения предметной области и обеспечение корректного состояния объектов.
2. Итерируемые объекты, итераторы и протокол итерации.
3. Генераторы и отложенная обработка последовательностей.
4. Аннотации типов в интерфейсах классов.
5. Обобщенные коллекции и протоколы типов.
6. Модули, пакеты и публичный интерфейс программы.
7. Принцип единственной ответственности.
8. Открытость классов к расширению и устойчивость к изменению.
9. Управление зависимостями и работа через абстракции.
10. Передача зависимостей и тестируемость.
11. Шаблон «Стратегия».
12. Фабричный метод.
13. Шаблон «Адаптер».
14. Сериализация и восстановление объектов.
15. Модульное тестирование состояния и поведения классов.
16. Изоляция внешних зависимостей при тестировании.
17. Документирование интерфейсов и журналирование.
18. Признаки проблемной объектной модели.
19. Рефакторинг без изменения внешнего поведения.
20. Архитектура небольшого прикладного приложения.

Типовые практические задания к зачету

1. Реализовать итератор для коллекции временных рядов с фильтрацией по периоду.
2. Создать стратегию расчета агрегированного показателя и обеспечить замену алгоритма без изменения клиента.
3. Разработать фабрику обработчиков для разных форматов входных данных.
4. Адаптировать внешний источник данных к интерфейсу приложения.
5. Реализовать сохранение и восстановление объектов с проверкой обязательных полей.
6. Написать модульные тесты класса сервиса с изолированной зависимостью хранилища.
7. Выполнить рефакторинг большого класса с выделением сущности, сервиса и хранилища.
8. Спроектировать слои небольшого приложения для анализа показателей.

Тесты по компетенциям

по дисциплине Б1.В.02.01 «Объектно-ориентированное программирование»

(шифр и наименование дисциплины)

№ задания	Содержание задания	Ключ к ответу	Наименование темы РПД	Семестр																								
ПК-1 Способность использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования																												
1	Что в объектной модели описывает допустимое состояние и поведение группы однотипных объектов? 1. Класс. 2. Цикл. 3. Файл. 4. Выражение.	1.	Объект, класс, состояние и поведение	2																								
2	Условие, которое должно оставаться истинным для каждого корректного состояния объекта, называется _____ класса.	инвариантом.	Контракты и инварианты класса	2																								
3	Установите соответствие между понятием и назначением. <table><tr><td>А. Атрибут экземпляра.</td><td>1. Управляемый доступ к данным объекта.</td></tr><tr><td>Б. Атрибут класса.</td><td>2. Данные конкретного объекта.</td></tr><tr><td>В. Свойство.</td><td>3. Общие данные класса.</td></tr><tr><td>Г. Метод экземпляра.</td><td>4. Операция, использующая состояние конкретного объекта.</td></tr></table> Запишите выбранные цифры под соответствующими буквами в виде таблицы: <table><tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr><tr><td></td><td></td><td></td><td></td></tr></table>	А. Атрибут экземпляра.	1. Управляемый доступ к данным объекта.	Б. Атрибут класса.	2. Данные конкретного объекта.	В. Свойство.	3. Общие данные класса.	Г. Метод экземпляра.	4. Операция, использующая состояние конкретного объекта.	А	Б	В	Г					<table><tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr><tr><td>2</td><td>3</td><td>1</td><td>4</td></tr></table>	А	Б	В	Г	2	3	1	4	Атрибуты экземпляра и класса	2
А. Атрибут экземпляра.	1. Управляемый доступ к данным объекта.																											
Б. Атрибут класса.	2. Данные конкретного объекта.																											
В. Свойство.	3. Общие данные класса.																											
Г. Метод экземпляра.	4. Операция, использующая состояние конкретного объекта.																											
А	Б	В	Г																									
А	Б	В	Г																									
2	3	1	4																									
4	Какие признаки указывают на корректное применение композиции? 1. Объект состоит из других объектов. 2. Компоненту можно передать отдельную ответственность. 3. Между объектами обязательно существует отношение «является». 4. Реализацию компонента требуется заменять независимо.	1, 2, 4.	Композиция как средство построения модели	2																								
5	Установите последовательность действий при разработке класса: А. Определить ответственность. Б. Выделить данные состояния. В. Задать публичные операции. Г. Проверить инварианты на примерах.	А, Б, В, Г.	Контракты и инварианты класса	2																								
6	Для взаимозаменяемых способов расчета показателя предпочтительно использовать общий интерфейс или одну функцию с большим числом условных ветвей?	общий интерфейс и полиморфные реализации, поскольку новый способ можно добавить без изменения клиентского кода.	Полиморфизм	2																								
7	Какие операции обычно реализуются специальными методами Python? 1. Текстовое представление. 2. Сравнение объектов. 3. Вычисление длины. 4. Установка операционной системы.	1, 2, 3.	Специальные методы Python	2																								
8	Механизм создания производного класса на основе базового класса называется _____.	наследованием.	Наследование классов	2																								

№ за- дания	Содержание задания	Ключ к ответу	Наименование темы РПД	Семестр																								
9	Объясните различие между объектом-сущностью и объектом-значением. Приведите пример из задачи анализа данных.	сущность различается по идентичности и сохраняет ее при изменении состояния. Объект-значение определяется совокупностью значений. Пример: набор данных как сущность и период наблюдения как значение.	Классы данных и объекты-значения	2																								
10	Класс отчета самостоятельно загружает данные, рассчитывает показатели и выводит результат. Предложите улучшение структуры.	разделить загрузку, расчет и представление между отдельными компонентами. Передавать источники и способы расчета через интерфейсы. Класс отчета должен координировать работу компонентов.	Композиция как средство построения модели	2																								
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач																												
11	Какое действие выполняется первым при системном анализе предметной области? 1. Определение цели и границ задачи. 2. Написание всех классов. 3. Выбор цвета интерфейса. 4. Оптимизация готового кода.	1.	Объектно-ориентированный подход	2																								
12	Установите соответствие между элементом анализа и вопросом. <table border="1"><tr><td>А. Цель.</td><td>1. Как определить достижение результата?</td></tr><tr><td>Б. Ограничение.</td><td>2. Какой результат требуется получить?</td></tr><tr><td>В. Критерий проверки.</td><td>3. Какие условия нельзя нарушить?</td></tr><tr><td>Г. Альтернатива.</td><td>4. Какой иной способ решения можно использовать?</td></tr></table> Запишите выбранные цифры под соответствующими буквами в виде таблицы: <table border="1"><tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr><tr><td></td><td></td><td></td><td></td></tr></table>	А. Цель.	1. Как определить достижение результата?	Б. Ограничение.	2. Какой результат требуется получить?	В. Критерий проверки.	3. Какие условия нельзя нарушить?	Г. Альтернатива.	4. Какой иной способ решения можно использовать?	А	Б	В	Г					<table border="1"><tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr><tr><td>2</td><td>3</td><td>1</td><td>4</td></tr></table>	А	Б	В	Г	2	3	1	4	Объектно-ориентированный подход	2
А. Цель.	1. Как определить достижение результата?																											
Б. Ограничение.	2. Какой результат требуется получить?																											
В. Критерий проверки.	3. Какие условия нельзя нарушить?																											
Г. Альтернатива.	4. Какой иной способ решения можно использовать?																											
А	Б	В	Г																									
А	Б	В	Г																									
2	3	1	4																									
13	Какие сведения нужны для критического сравнения двух объектных моделей? 1. Соответствие требованиям. 2. Связанность компонентов. 3. Проверимость поведения. 4. Длина имен всех переменных как единственный критерий.	1, 2, 3.	Объектно-ориентированный подход	2																								
14	Установите последовательность решения: А. Сформулировать цель и собрать требования. Б. Сравнить варианты модели. В. Реализовать выбранный вариант. Г. Проверить результат.	А, Б, В, Г.	Объектно-ориентированный подход	2																								
15	Разбиение сложной задачи на связанные подзадачи называется _____.	декомпозицией.	Композиция как средство построения модели	2																								
16	Требования к системе противоречат друг другу. Следует сразу писать код или сначала уточнить приоритеты и критерии?	сначала уточнить приоритеты, ограничения и критерии. Реализация без разрешения противоречий повышает риск неверной модели.	Объектно-ориентированный подход	2																								

№ за- дания	Содержание задания	Ключ к ответу	Наименование темы РПД	Семестр
17	Что делает решение о применении наследования обоснованным? 1. Производный объект действительно является разновидностью базового. 2. Подстановка не нарушает ожидаемое поведение. 3. Требуется только повторно использовать две строки кода. 4. Иерархия отражает устойчивое отношение предметной области.	1, 2, 4.	Наследование классов	2
18	Измеримое условие, по которому проверяется достижение цели, называется _____.	критерием.	Контракты и инварианты класса	2
19	Опишите стратегию выбора между композицией и наследованием для модели способов расчета.	определить устойчивые отношения предметной области, требования к замене поведения и подстановке. При отношении «имеет» и необходимости замены компонента выбрать композицию. Наследование применять при обоснованном отношении «является» и сохранении контракта.	Композиция как средство построения модели	2
20	Заказчик просит добавить новый тип показателя, но текущая программа содержит условия по типу во многих функциях. Определите цель и стратегию изменения.	цель — локализовать различия типов. Выделить общий интерфейс, перенести специфическое поведение в отдельные классы, заменить условия полиморфным вызовом и проверить старые сценарии тестами.	Полиморфизм	2
ПК-1 Способность использовать современные методы разработки и реализации конкретных алгоритмов математических моделей на базе языков программирования и пакетов прикладных программ моделирования				
21	Какой метод возвращает итератор объекта в протоколе Python? 1. __iter__. 2. __init__. 3. __repr__. 4. __len__.	1.	Итерируемые объекты и итераторы	3
22	Объект, который создает элементы последовательности по мере запроса, называется _____.	генератором.	Генераторы и отложенная обработка	3
23	Установите соответствие между шаблоном и назначением.		Базовые шаблоны проектирования	3
	А. Стратегия.	1. Приведение внешнего интерфейса к требуемому.		
	Б. Фабричный метод.	2. Замена алгоритма через общий интерфейс.		
	В. Адаптер.	3. Передача создания конкретного объекта отдельному компоненту.		
	Г. Наблюдатель.	4. Уведомление зависимых объектов об изменении состояния.		
Запишите выбранные цифры под соответствующими буквами в виде таблицы:				
А	Б	В	Г	

№ задания	Содержание задания	Ключ к ответу	Наименование темы РПД	Семестр
24	Что повышает тестируемость класса? 1. Передача зависимостей извне. 2. Работа через небольшой интерфейс. 3. Создание всех внешних ресурсов внутри каждого метода. 4. Отделение вычислений от ввода и вывода.	1, 2, 4.	Управление зависимостями	3
25	Установите последовательность тестирования: А. Подготовить исходное состояние. Б. Выполнить операцию. В. Получить фактический результат. Г. Сравнить с ожидаемым.	А, Б, В, Г.	Модульное тестирование классов	3
26	Где следует проверять обязательные поля при восстановлении объекта из JSON?	на границе преобразования данных в объект до передачи объекта прикладной логике. Это предотвращает появление некорректного состояния внутри модели.	Хранение и восстановление состояния объектов	3
27	Какие признаки могут требовать рефакторинга класса? 1. Несвязанные обязанности. 2. Большое число зависимостей. 3. Дублирование правил. 4. Наличие короткого понятного интерфейса.	1, 2, 3.	Рефакторинг объектной модели	3
28	Изменение внутренней структуры программы без изменения наблюдаемого поведения называется _____.	рефакторингом.	Рефакторинг объектной модели	3
29	Объясните, как разделить алгоритм расчета показателя от загрузки и сохранения данных.	выделить чистый компонент расчета с входными объектами и результатом, определить интерфейсы источника и хранилища, передавать реализации зависимостей прикладному сервису, отдельно тестировать расчет.	Архитектура небольшого приложения	3
30	Приложение должно обрабатывать CSV и JSON одинаковым алгоритмом. Предложите объектную структуру.	общий интерфейс источника данных, отдельные адаптеры CSV и JSON, единая модель записей и сервис обработки, зависящий от интерфейса источника.	Базовые шаблоны проектирования	3
УК-1 Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач				
31	Какое действие должно предшествовать рефакторингу работающей модели? 1. Фиксация текущего поведения тестами. 2. Удаление всех проверок. 3. Замена языка программирования. 4. Объединение всех классов.	1.	Рефакторинг объектной модели	3

№ задания	Содержание задания	Ключ к ответу	Наименование темы РПД	Семестр								
32	Установите соответствие между проектным решением и проверяемым критерием.	<table><tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr><tr><td>2</td><td>3</td><td>1</td><td>4</td></tr></table>	А	Б	В	Г	2	3	1	4	Управление зависимостями	3
	А		Б	В	Г							
	2		3	1	4							
	А. Передача зависимости.		1. Возможность заменить алгоритм.									
Б. Выделение слоя доступа к данным.	2. Возможность тестировать без реального внешнего ресурса.											
В. Стратегия обработки.	3. Отделение хранения от прикладной логики.											
	Г. Фасад компонента.	4. Упрощение доступа клиента к группе операций.										
	Запишите выбранные цифры под соответствующими буквами в виде таблицы:											
	<table><tr><td>А</td><td>Б</td><td>В</td><td>Г</td></tr><tr><td></td><td></td><td></td><td></td></tr></table>	А	Б	В	Г							
А	Б	В	Г									
33	Какие вопросы помогают определить границы компонента? 1. За какой результат он отвечает? 2. Какие данные получает и возвращает? 3. От каких внешних средств зависит? 4. Сколько случайных методов можно в него добавить?	1, 2, 3.	Принципы проектирования классов	3								
34	Установите последовательность проектного изменения: А. Определить проблему и критерий улучшения. Б. Подготовить тесты. В. Выполнить изменение. Г. Проверить критерий и тесты.	А, Б, В, Г.	Рефакторинг объектной модели	3								
35	Зависимость компонента от минимального набора сведений о другом компоненте называется слабой _____.	связанностью.	Модули и пакеты	3								
36	Проект использует внешний сервис, который нестабилен в учебной среде. Как обеспечить повторяемость модульных тестов?	определить интерфейс сервиса и использовать контролируемый тестовый заместитель. Реальный сервис проверять отдельными интеграционными тестами.	Изоляция зависимостей при тестировании	3								
37	Что учитывается при выборе формата хранения объектной модели? 1. Состав и типы данных. 2. Требования к обмену и совместимости. 3. Риски загрузки недоверенных данных. 4. Только длина расширения файла.	1, 2, 3.	Хранение и восстановление состояния объектов	3								
38	Представление сложной системы в виде уровней с различной ответственностью называется _____ архитектурой.	слоистой.	Архитектура небольшого приложения	3								
39	Опишите стратегию разработки небольшого приложения от требования до проверенного результата.	определить цель, границы и критерии. Выделить модель предметной области и компоненты. Сравнить альтернативы. Реализовать минимальный сценарий. Добавить тесты, проверить требования, выполнить рефакторинг и представить результат.	Развитие и сопровождение программного проекта	3								

№ задания	Содержание задания	Ключ к ответу	Наименование темы РПД	Семестр
40	В проекте одновременно меняются правила расчета, формат данных и способ вывода. Предложите последовательность безопасных изменений.	зафиксировать текущее поведение тестами, разделить изменения по компонентам, определить интерфейсы, сначала стабилизировать модель данных, затем алгоритм расчета, после этого адаптеры ввода и вывода. После каждого шага выполнять тесты и проверять критерии.	Развитие и сопровождение программного проекта	3

Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие процесс формирования компетенций

Оценка достижения запланированных результатов обучения производится по итогам отдельных видов текущего контроля и промежуточной аттестации.

Шкала оценивания

«Зачтено» выставляется, если сформированность заявленных знаний, умений и навыков обучающегося составляет 60% и более и ни одна обязательная часть зачета не оценена «неудовлетворительно».

«Отлично» (95–100%) выставляется, если обучающийся показал прочные систематизированные знания, самостоятельно решает практические задачи повышенной сложности, свободно использует справочную литературу, аргументирует решение и делает обоснованные выводы.

«Хорошо» (75–94%) выставляется, если обучающийся показал прочные знания, самостоятельно решает предусмотренные рабочей программой практические задачи, ориентируется в рекомендованной литературе и правильно оценивает полученные результаты с отдельными неточностями.

«Удовлетворительно» (60–74%) выставляется, если обучающийся знает основные положения учебного материала, знаком с рекомендованной литературой и способен получить правильное решение типовой практической задачи самостоятельно или с ограниченной помощью преподавателя.

«Неудовлетворительно» или «не зачтено» (менее 60%) выставляется, если выявлены существенные пробелы в знаниях основных положений учебного материала и обучающийся не может получить правильное решение типовой практической задачи с помощью преподавателя.

Ответы и решения обучающихся оцениваются по следующим общим критериям: распознавание проблемы; определение значимой информации; анализ проблемы; аргументированность; использование подходящей стратегии; правильность решения; обоснованность выводов; творческий подход; общая грамотность.

Текущий контроль

Результат тестирования определяется как доля набранных баллов от максимально возможного количества баллов. Положительным считается результат не менее 60%. Практическое или ситуационное задание оценивается с учетом правильности решения, полноты анализа, самостоятельности, аргументированности и качества выводов.

Промежуточная аттестация

Экзамен во 2-м семестре проводится по вопросам и практическим заданиям дисциплины. Билет включает два теоретических вопроса и одно практическое задание. Зачет в 3-м семестре проводится по вопросам и практическим заданиям дисциплины. Теоретическая и практическая части оцениваются по традиционной шкале. При проведении экзамена и зачета могут учитываться положительные результаты текущего контроля за соответствующий семестр.

Уровень освоения	Традиционная оценка	Оценка при зачете
95–100%	5 — отлично	Зачтено
75–94%	4 — хорошо	Зачтено
60–74%	3 — удовлетворительно	Зачтено
Менее 60%	2 — неудовлетворительно	Не зачтено

Обучающиеся выполняют задания в сроки, установленные преподавателем. Оценка «зачтено» может выставляться при неполной сформированности отдельных компетенций, если их формирование продолжается на последующих этапах образовательной программы.