

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Структурное подразделение «Институт информационных технологий и анализа данных»

УТВЕРЖДЕНА:

на заседании Совета института ИТиАД им. Е.И.Попова

Протокол №8 от 24 февраля 2025 г.

Рабочая программа дисциплины

«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»

Направление: 09.03.02 Информационные системы и технологии

Информационные системы и технологии в административном управлении

Квалификация: Бакалавр

Форма обучения: очная

Документ подписан простой
электронной подписью
Составитель программы:
Аршинский Вадим
Леонидович
Дата подписания: 11.06.2025

Документ подписан простой
электронной подписью
Утвердил: Говорков Алексей
Сергеевич
Дата подписания: 16.06.2025

Документ подписан простой
электронной подписью
Согласовал: Аршинский
Вадим Леонидович
Дата подписания: 11.06.2025

Год набора – 2025

Иркутск, 2025 г.

1 Перечень планируемых результатов обучения по дисциплине, соотнесённых с планируемыми результатами освоения образовательной программы

1.1 Дисциплина «Объектно-ориентированное программирование» обеспечивает формирование следующих компетенций с учётом индикаторов их достижения

Код, наименование компетенции	Код индикатора компетенции
ПКС-2 Способен разрабатывать программное обеспечение (ПО), включая проектирование, отладку, проверку работоспособности и модификацию ПО	ПКС-2.1, ПКС-2.2

1.2 В результате освоения дисциплины у обучающихся должны быть сформированы

Код индикатора	Содержание индикатора	Результат обучения
ПКС-2.1	Способен выполнять объектную декомпозицию задачи и реализовывать классы с использованием объектно-ориентированного языка программирования	Знать основные понятия и принципы объектного подхода; синтаксис и основные конструкции изучаемого языка программирования Уметь выполнять объектную декомпозицию задачи; реализовывать выделенные в ходе анализа задачи классы с использованием объектно-ориентированного языка программирования Владеть современным объектно-ориентированным языком программирования; современной интегрированной средой разработки
ПКС-2.2	Способен использовать объектно-ориентированный подход в процессе создания программного обеспечения	Знать методы и технологии объектно-ориентированного программирования; основные шаблоны проектирования Уметь применять объектно-ориентированную методологию программирования; использовать современные программные библиотеки и фреймворки Владеть методологией объектно-ориентированного подхода к разработке программного обеспечения; стандартными средствами проектирования, отладки и тестирования программного обеспечения

2 Место дисциплины в структуре ООП

Изучение дисциплины «Объектно-ориентированное программирование» базируется на результатах освоения следующих дисциплин/практик: «Базы данных», «Информатика», «Оздоровительная физическая культура», «Операционные системы», «Программирование на языке высокого уровня», «Учебная практика: технологическая (проектно-технологическая) практика», «Математическая логика и дискретная математика»

Дисциплина является предшествующей для дисциплин/практик: «Основы мобильной разработки», «Технологии разработки программных комплексов», «Проектная деятельность»

3 Объем дисциплины

Объем дисциплины составляет – 8 ЗЕТ

Вид учебной работы	Трудоемкость в академических часах (Один академический час соответствует 45 минутам астрономического часа)		
	Всего	Семестр № 3	Семестр № 4
Общая трудоемкость дисциплины	288	144	144
Аудиторные занятия, в том числе:	112	64	48
лекции	32	16	16
лабораторные работы	80	48	32
практические/семинарские занятия	0	0	0
Самостоятельная работа (в т.ч. курсовое проектирование)	140	80	60
Трудоемкость промежуточной аттестации	36	0	36
Вид промежуточной аттестации (итогового контроля по дисциплине)	Зачет, Экзамен, Курсовой проект	Зачет	Экзамен, Курсовой проект

4 Структура и содержание дисциплины

4.1 Сводные данные по содержанию дисциплины

Семестр № 3

№ п/п	Наименование раздела и темы дисциплины	Виды контактной работы						СРС		Форма текущего контроля
		Лекции		ЛР		ПЗ(СЕМ)		№	Кол. Час.	
		№	Кол. Час.	№	Кол. Час.	№	Кол. Час.			
1	2	3	4	5	6	7	8	9	10	11
1	Введение в объектно-ориентированное программирование	4	4	1	8			1, 2, 3, 4, 5	30	Отчет по лабораторной работе
2	Основы объектно-ориентированного	1	4	2	10			1, 3, 4, 5	16	Отчет по лабораторной работе

	о подхода и базовые средства выражения объектной абстракции									ной работе
3	Объектно-ориентированный подход к обработке ошибок и тестированию	2	4	3	16			1, 3, 4, 5	16	Отчет по лабораторной работе
4	Основные принципы объектно-ориентированного подхода	3	4	4	14			1, 3, 4, 5	18	Отчет по лабораторной работе
	Промежуточная аттестация									Зачет
	Всего		16		48				80	

Семестр № 4

№ п/п	Наименование раздела и темы дисциплины	Виды контактной работы						СРС		Форма текущего контроля
		Лекции		ЛР		ПЗ(СЕМ)		№	Кол. Час.	
		№	Кол. Час.	№	Кол. Час.	№	Кол. Час.			
1	2	3	4	5	6	7	8	9	10	11
1	Применение объектно-ориентированного подхода при разработке приложений с графическим пользовательским интерфейсом	1	8	1	12			1, 2, 3, 4, 5	40	Отчет по лабораторной работе
2	Объектный подход к организации работы с базами данных	3	4	2	8			2, 3, 4, 5	10	Отчет по лабораторной работе
3	Применение объектно-ориентированного подхода при разработке Android-приложений	2	4	3	12			2, 3, 4, 5	10	Отчет по лабораторной работе
	Промежуточная аттестация								36	Экзамен, Курсовой проект
	Всего		16		32				96	

4.2 Краткое содержание разделов и тем занятий

Семестр № 3

№	Тема	Краткое содержание
1	Введение в объектно-ориентированное	Сложность разработки программного обеспечения. Способы борьбы со сложностью - абстракция,

	программирование	декомпозиция, обобщение. Эволюция средств выражения абстракции в языках программирования. Роль проектирования в процессе разработки программного обеспечения. Основы синтаксиса современного объектно-ориентированного языка программирования.
2	Основы объектно-ориентированного подхода и базовые средства выражения объектной абстракции	Понятия объекта и класса. Краткая информация о платформе Java. Основные языковые конструкции Java: объявление классов, примитивные типы данных, объявление переменных и методов, операторы управления порядком выполнения. Ссылочные типы данных и создание объектов. Инкапсуляция. Модификаторы доступа. Конструкторы. Контекст выполнения this. Массивы и строки. Пакеты Java. Статические поля и методы.
3	Объектно-ориентированный подход к обработке ошибок и тестированию	Проблема реализации системы ввода-вывода и объектный подход к её решению. Понятие потока данных в Java, иерархия классов потоков данных. Низкоуровневые и высокоуровневые потоки данных. Наследование. Исключительная ситуация как объект. Механизм обработки исключений, Делегирование обработки исключений. Механизм генерации (возбуждения) исключений. Модульные тесты.
4	Основные принципы объектно-ориентированного подхода	Инкапсуляция, полиморфизм, наследование и их взаимосвязь. Применение абстрактных классов. Проблема множественного наследования, интерфейсы, переопределение методов. Обобщения (генерики). Механизмы рефлексии (интроспекция).

Семестр № 4

№	Тема	Краткое содержание
1	Применение объектно-ориентированного подхода при разработке приложений с графическим пользовательским интерфейсом	Понятие графического пользовательского интерфейса и подходы к его реализации при разработке приложений. Концепция Модель/Вид/Контроллер (MVC). Интерфейсы программирования приложений (API) и понятие компонента графического интерфейса. Базовые библиотеки и инструменты создания GUI Java (AWT, Swing, JavaFX). Иерархия компонентов и контейнеров и работа с ними. Событийно-управляемый механизм взаимодействия объектов. Шаблон программирования Listener. Реализация обработчиков событий объектов прослушивания.
2	Объектный подход к организации работы с базами данных	Создание подключения к базе данных и технология JDBC. Понятие JDBC-драйвера. Встраиваемая СУБД SQLite. Объектно-реляционное отображение (ORM). Классы-сущности и фабрики сессий.

3	Применение объектно-ориентированного подхода при разработке Android-приложений	Инструментальные средства разработки и структура Android-проекта. ViewGroup и основные контейнеры. Инструмент отладки Logcat. Реализация динамического интерфейса. Адаптеры списков.
---	--	--

4.3 Перечень лабораторных работ

Семестр № 3

№	Наименование лабораторной работы	Кол-во академических часов
1	Разработка консольного приложения	8
2	Разработка классов и их использование	10
3	Разработка приложения работающего с потоками данных	16
4	Разработка библиотеки классов с использованием механизмов наследования, полиморфизма и инкапсуляции	14

Семестр № 4

№	Наименование лабораторной работы	Кол-во академических часов
1	Разработка приложения с графическим пользовательским интерфейсом	12
2	Организация работы с базой данных	8
3	Разработка Android-приложения	12

4.4 Перечень практических занятий

Практических занятий не предусмотрено

4.5 Самостоятельная работа

Семестр № 3

№	Вид СРС	Кол-во академических часов
1	Оформление отчетов по лабораторным и практическим работам	8
2	Подготовка к зачёту	16
3	Подготовка к практическим занятиям (лабораторным работам)	16
4	Подготовка к сдаче и защите отчетов	8
5	Проработка разделов теоретического материала	32

Семестр № 4

№	Вид СРС	Кол-во академических часов
1	Написание курсового проекта (работы)	30
2	Оформление отчетов по лабораторным и	6

	практическим работам	
3	Подготовка к практическим занятиям (лабораторным работам)	6
4	Подготовка к сдаче и защите отчетов	6
5	Проработка разделов теоретического материала	12

В ходе проведения занятий по дисциплине используются следующие интерактивные методы обучения: Видеолекция, Лекция-провокация, Метод проектов

5 Перечень учебно-методического обеспечения дисциплины

5.1 Методические указания для обучающихся по освоению дисциплины

5.1.1 Методические указания для обучающихся по курсовому проектированию/работе:

Аршинский В.Л. Объектно-ориентированное программирование : электронный курс / В.Л. Аршинский, Маланова Т.В. <https://el.istu.edu/course/view.php?id=4237>

5.1.2 Методические указания для обучающихся по лабораторным работам:

1. Аршинский В.Л. Объектно-ориентированное программирование : электронный курс / В.Л. Аршинский, Маланова Т.В. <https://el.istu.edu/course/view.php?id=4237>
2. Аршинский В.Л. Объектно-ориентированное программирование [Электронный ресурс] : методические указания по выполнению лабораторных работ / Иркут. нац. исслед. техн. унт, 2017. - 24 с. Режим доступа: <http://elib.istu.edu/viewer/view.php?file=/files/er-1565.pdf>

5.1.3 Методические указания для обучающихся по самостоятельной работе:

Аршинский В.Л. Объектно-ориентированное программирование : электронный курс / В.Л. Аршинский, Маланова Т.В. <https://el.istu.edu/course/view.php?id=4237>

6 Фонд оценочных средств для контроля текущей успеваемости и проведения промежуточной аттестации по дисциплине

6.1 Оценочные средства для проведения текущего контроля

6.1.1 семестр 3 | Отчет по лабораторной работе

Описание процедуры.

Выполнение каждой лабораторной работы состоит из двух частей: практической части в ходе которой необходимо выполнить индивидуальное задание и подготовка и защита отчета по лабораторной работе. Реализация решения индивидуального задания осуществляется средствами платформы Java. В ходе выполнения лабораторной работы студент должен применить теоретические знания об объектном подходе к программированию, полученные в ходе изучения курса, а также приобрести практические навыки использования объектно-ориентированного языка программирования Java для решения типовых задач. Подготовка отчета по лабораторной работе необходима для закрепления и переосмысления полученных знаний и навыков. Для успешной сдачи лабораторной работы необходимо продемонстрировать работу программы, реализующую индивидуальное задание, устранить выявленные преподавателем недостатки, подготовить и защитить отчет.

Требования к оформлению программного кода, а также содержанию отчета указаны в

соответствующем разделе методических указаний для каждой лабораторной работы.

Список тем контрольных вопросов для защиты отчета:

К лабораторной работе №1

1. Создание и запуск Java-программ и состав платформы Java.
2. Точка входа в программу и назначение аргумента метода `main(String[] args)`.
3. Блоки кода и видимость переменных в Java.
4. Объявление переменных и работа с массивами в Java.
5. Типы данных в Java.
6. Операторы цикла Java.
7. Условные операторы Java.
8. Константы в Java, назначение и использование.
9. Глобальные, локальные и временные (temporary) переменные в Java.

К лабораторной работе №2:

1. Интегрированная среда разработки (IDE), её интерфейс и основные возможности.
2. Пакеты java.
3. Классы и объекты.
4. Объявление класса.
5. Члены класса: поля и методы.
6. Модификаторы доступа.
7. Создание объекта.
8. Конструкторы и их вызов.
9. Вызов методов объекта.
10. Передача параметров при вызове методов.
11. Сигнатура метода.

К лабораторной работе №3:

1. Система ввода/вывода.
2. Понятие потока данных.
3. Байтовые и символьные потоки данных.
4. Низкоуровневые и высокоуровневые потоки данных.
5. Цепочки потоков данных.
6. Исключения в java.
7. Проверяемые исключения.
8. Блок `finally`.
9. Возбуждение исключения.
10. Модификатор `static`.
11. Вызов статических методов.
12. Статические поля.

К лабораторной работе №4:

1. Наследование.
2. Инкапсуляция.
3. Полиморфизм.
4. Абстрактный класс, абстрактные методы.
5. Переопределение метода.
6. Реализация интерфейса, множественное наследование.
7. Отношение классов «агрегирование».
8. Основные коллекции java (`ArrayList`, `LinkedList`, `HashTable`).

Критерии оценивания.

В процессе защиты отчета по лабораторной работе, студенту задаются контрольные вопросы теоретического и практического характера, соответствующие теме работы. Для успешной защиты отчета студенту необходимо дать краткое изложение основных результатов полученных в ходе выполнения лабораторной работы, устно ответить на теоретические вопросы по теме лабораторной работы, а также продемонстрировать умение ориентироваться в написанном программном коде. Успешная защита отчета является необходимым условием для выставления оценки «зачтено» по соответствующей лабораторной работе.

6.1.2 семестр 4 | Отчет по лабораторной работе

Описание процедуры.

Выполнение каждой лабораторной работы состоит из двух частей: практической части в ходе которой необходимо выполнить индивидуальное задание и подготовка и защита отчета по лабораторной работе. Реализация решения индивидуального задания осуществляется средствами платформы Java. В ходе выполнения лабораторной работы студент должен применить теоретические знания об объектно-ориентированном подходе к программированию, полученные в ходе изучения курса, а также приобрести практические навыки использования объектно-ориентированного языка программирования Java для решения типовых задач. Подготовка отчета по лабораторной работе необходима для закрепления и переосмысления полученных знаний и навыков. Для успешной сдачи лабораторной работы необходимо продемонстрировать работу программы, реализующую индивидуальное задание, устранить выявленные преподавателем недостатки, подготовить и защитить отчет.

Требования к оформлению программного кода, а также содержанию отчета указаны в соответствующем разделе методических указаний для каждой лабораторной работы.

Список тем контрольных вопросов для защиты отчета:

К лабораторной работе №1(5):

1. Понятие компонента графического пользовательского интерфейса.
2. Назначение контейнеров Swing.
3. Назначение планировщиков раскладки компонентов.
4. Наиболее распространённые планировщики раскладки.
5. Создание своего планировщика раскладки.
6. Обработка событий.
7. Интерфейс ActionListener.
8. Архитектурный паттерн MVC.

К лабораторной работе №2(6):

1. Определение понятий БД и СУБД, какова их роль в программной разработке?
2. Назначение JDBC-драйверов и их виды.
3. Общая схема работы с БД базовыми средствами Java.
4. Классы Statement, Connection и ResultSet.
5. Общие принципы объектно-реляционного отображения.
6. Технология работы с ORM-фреймворками.

К лабораторной работе №3(7):

1. Структура Android-проекта.
2. Разметка экранов приложения.

3. Работа с ресурсами Android-проекта.
4. Жизненный цикл Activity и основные callback-методы.
5. Инструмент Logcat.
6. Какие бывают GroupView и для чего какой использовать?
7. Основные View в Android и работа с ними.
8. RecyclerView и RecyclerViewAdapter.

Критерии оценивания.

В процессе защиты отчета по лабораторной работе, студенту задаются контрольные вопросы теоретического и практического характера, соответствующие теме работы. Для успешной защиты отчета студенту необходимо дать краткое изложение основных результатов полученных в ходе выполнения лабораторной работы, устно ответить на теоретические вопросы по теме лабораторной работы, а также продемонстрировать умение ориентироваться в написанном программном коде. Успешная защита отчета является необходимым условием для выставления оценки «зачтено» по соответствующей лабораторной работе.

6.2 Оценочные средства для проведения промежуточной аттестации

6.2.1 Критерии и средства (методы) оценивания индикаторов достижения компетенции в рамках промежуточной аттестации

Индикатор достижения компетенции	Критерии оценивания	Средства (методы) оценивания промежуточной аттестации
ПКС-2.1	Уверенно демонстрирует полученные знания, приводит примеры, отвечает на вопросы. Способен с использованием современного объектно-ориентированного языка программирования реализовывать классы на основе результатов объектного анализа задачи	Устное собеседование по теоретическим вопросам и выполнение практического задания
ПКС-2.2	Уверенно демонстрирует полученные знания, приводит примеры, отвечает на вопросы. Способен с использованием современного объектно-ориентированного языка программирования реализовывать фрагмент программного обеспечения на основе результатов объектного анализа предметной области	Устное собеседование по теоретическим вопросам и выполнение практического задания

6.2.2 Типовые оценочные средства промежуточной аттестации

6.2.2.1 Семестр 3, Типовые оценочные средства для проведения зачета по дисциплине

6.2.2.1.1 Описание процедуры

Зачет ставится по результатам выполнения лабораторных работ в 3 семестре, посещения лекций и ответов на вопросы из билета к зачету. Для допуска к зачету должны быть выполнены и защищены все лабораторные работы за 3 семестр. На зачете студент должен устно ответить на вопросы билета, а также быть способен составить программный код по тематике вопроса по требованию преподавателя.

Студент вправе не отвечать на вопросы билета и получить зачет по дисциплине, если он не пропустил ни одной лекции по дисциплине в течение 3 семестра и активно работал в ходе лабораторного практикума. Студент вправе отвечать только на один из вопросов билета (по своему выбору) и получить зачет по дисциплине, если он пропустил не более 2 лекций по дисциплине в 3 семестре. Допускается письменный ответ на вопросы билета на зачете (по решению преподавателя).

Список вопросов к зачёту:

1. Предпосылки возникновения объектно-ориентированного подхода.
2. Сложность разработки программного обеспечения.
3. Способы борьбы со сложностью разработки программного обеспечения.
4. Структурная и объектная декомпозиция.
5. Объектно-ориентированные языки программирования.
6. Объекты и классы: определение этих понятий и их взаимосвязь.
7. Спецификация (определение) класса, его члены.
8. Конструкторы, деструкторы: назначение и использование. Пояснить на примерах.
9. Сигнатура метода и его аргументы. Привести примеры несовместных сигнатур.
10. Поведение и состояние объекта. Раскрыть понятия и проиллюстрировать примером.
11. Механизмы ограничения доступа к членам класса. Пояснить, зачем они нужны.
12. Статические члены класса: определение и использование. Привести примеры.
13. Инкапсуляция в объектно-ориентированном программировании.
14. Взаимодействие объектов, активные и пассивные объекты.
15. Создание объекта и вызов его методов в Java.
16. Использование псевдонима объекта в контексте выполнения его методов (this)
17. Перечислить основные составляющие платформы Java и их назначение.
18. Технология сборки и запуска Java-приложений.
19. Основные конструкции языка Java. Привести примеры.
20. Примитивные и ссылочные типы Java. Примеры объявления переменных.
21. Назначение метода `public static void main()` и его аргументы.
22. Работа с массивами в Java. Привести примеры.
23. Модификатор `final` в Java. Примеры использования.
24. Область видимости переменных (scope) в Java. Привести примеры.
25. Пакеты Java: назначение и использование. Привести примеры стандартных пакетов.
26. Понятие потока ввода/вывода Java. Привести примеры.
27. Низкоуровневые и высокоуровневые потоки ввода/вывода Java.
28. Цепочки потоков ввода/вывода.
29. Классы `Reader` и `Writer`: назначение и отличия от `Input/OutputStream`
30. Обработка исключительных ситуаций (Exceptions). Привести примеры.
31. Делегирование исключительных ситуаций (Exceptions). Привести примеры.
32. Проверяемые (checked) и непроверяемые (runtime) исключения Java.
33. Наследование в объектно-ориентированном программировании.
34. Полиморфизм в объектно-ориентированном программировании.
35. Множественное наследование: реализация интерфейсов Java.

Пример задания:

Типовой билет к зачёту:

Вопрос 1:

Объекты и классы: определение этих понятий и их взаимосвязь.

Вопрос 2:

Обработка исключительных ситуаций (Exceptions). Продемонстрировать на примере.

6.2.2.1.2 Критерии оценивания

Зачтено	Не зачтено
демонстрирует понимание основ объектно-ориентированного программирования и знания основных понятий объектно-ориентированного подхода, владеет понятийным аппаратом и методами объектной декомпозиции для решения типовых задач в рамках предметной области, владеет объектно-ориентированным языком программирования на базовом уровне: способен описывать классы, создавать объекты и вызывать их методы для решения типовых задач.	наличие грубых ошибок в ответах на вопросы, непонимание сущности излагаемого вопроса, отсутствие навыков использования объектно-ориентированного программирования для решения типовых задач

6.2.2.2 Семестр 4, Типовые оценочные средства для проведения экзамена по дисциплине

6.2.2.2.1 Описание процедуры

Экзамен проводится в устной форме, для допуска до экзамена необходимо сдать все лабораторные работы. Оформляются 15 билетов, в каждом билете 2 теоретических вопроса и один практический. Оценка выставляется по результатам собеседования по вопросам.

Вопросы к экзамену:

1. Исторические предпосылки возникновения объектно-ориентированного подхода.
2. Сложность разработки программного обеспечения: из чего складывается, чем обуславливается?
3. Способы борьбы со сложностью разработки программного обеспечения.
4. Структурная и объектная декомпозиция. В чем разница? Примеры.
5. Определение объектно-ориентированного проектирования и программирования и их взаимосвязь.
6. Объектно-ориентированные языки программирования: признаки таких языков, примеры.
7. Объекты и классы: дать определение этих понятий и раскрыть их взаимосвязь. Проиллюстрировать на примерах.
8. Взаимодействие объектов, активные и пассивные объекты.
9. Механизмы ограничения доступа к членам класса в Java. Пояснить, зачем они нужны.
10. Конструкторы: назначение и использование. Привести примеры объявления и вызова Java.
11. Поведение и состояние объекта (формальные определения и пример).
12. Понятие инкапсуляции в объектно-ориентированном программировании. Почему это

один из основополагающих принципов ООП?

13. Наследование в объектно-ориентированном программировании. Почему это один из основополагающих принципов ООП? Как реализуется в Java?
14. Полиморфизм в объектно-ориентированном программировании. Почему это один из основополагающих принципов ООП? Статический и динамический полиморфизм в Java
15. Статические члены класса в Java. Назначение и применение.
16. Объявление класса в Java, какие могут быть члены у класса? Привести примеры.
17. Абстрактные классы: назначение, использование.
18. Множественное наследование, «проблема ромба».
19. Вызов методов суперкласса в Java. Привести примеры.
20. Множественное наследование в Java: реализация интерфейсов.
21. Переопределение унаследованных методов. Привести примеры.
22. Какие компоненты входят в состав платформы Java и их назначение?
23. Создание объекта и вызов его методов в Java.
24. Назначение метода `public static void main()` и его аргументы.
25. Пакеты Java: назначение и использование.
26. Объявление и особенности работы с массивами в Java, примеры.
27. Основные конструкции языка Java (примеры).
28. Сигнатура метода и его аргументы.
29. Область видимости переменных в Java: какие возможны варианты и для чего?
30. Использование модификатора `final` в Java: где можно применять и какой эффект оказывает?
31. Байтовые потоки ввода/вывода Java. Примеры.
32. Символьные потоки ввода/вывода Java. Примеры.
33. Низкоуровневые и высокоуровневые потоки ввода/вывода Java. Примеры.
34. Цепочки (chains) потоков ввода/вывода. Примеры.
35. Обработка исключительных ситуаций (Exceptions) в Java.
36. Проверяемые (checked) и непроверяемые (runtime) исключения Java.
37. Делегирование обработка исключительных ситуаций (throws exceptions) в Java.
38. Примитивные и ссылочные типы Java. В чем между ними разница? Примеры.
39. Что нужно для создания и/или запуска Java-приложений? Продемонстрировать технологию запуска java-приложения из консоли.
40. Использование псевдонима объекта (this) в контексте выполнения его методов в Java.
41. Анонимные классы в Java – определение и примеры использования.
42. Шаблон проектирования MVC.
43. Компоненты и контейнеры Swing.
44. Планировщики раскладки компонентов. Примеры.
45. Обработка событий: шаблон Listener.
46. Работа с компонентом Swing JTable.
47. Коллекции Java: ArrayList и LinkedList (на примере).
48. Общая схема работы с БД базовыми средствами Java.
49. Понятие ORM и общая схема работы в Java.
50. UML: диаграмма классов, виды отношений между классами.

Пример задания:

Типовой билет к экзамену:

1. Поведение и состояние объекта (формальные определения и пример).
2. Понятие ORM и общая схема работы в Java.
3. Обработка событий: шаблон Listener.

6.2.2.2 Критерии оценивания

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
<p>глубокое понимание теоретических основ объектно-ориентированного программирования, отличные знания основных понятий объектно-ориентированного подхода к разработке программного обеспечения, уверенное владение понятийным аппаратом и техниками объектно-ориентированного подхода при решении типовых задач в рамках предметной области, грамотное и логически стройное изложение материала при ответах на все вопросы из билета, знание дополнительно рекомендованной литературы</p>	<p>хорошее понимание теоретических основ объектно-ориентированного программирования и знания основных понятий объектно-ориентированного подхода, владеет понятийным аппаратом и методами объектной декомпозиции для решения типовых задач в рамках предметной области, незначительные ошибки при ответах на заданные вопросы, демонстрирует хорошее владение объектно-ориентированным языком программирования и программно-инструментальными средствами</p>	<p>понимание теоретических основ объектно-ориентированного программирования с незначительными пробелами, знает основные понятия и методы объектно-ориентированного подхода, изложение ответов с ошибками, уверенно исправляемыми после дополнительных вопросов, необходимость наводящих вопросов для получения верного ответа от студента</p>	<p>наличие грубых ошибок в ответе, непонимание сущности излагаемого вопроса, неуверенность и неточность ответов на дополнительные и наводящие вопросы, отсутствие навыков владения объектно-ориентированным языком программирования.</p>

6.2.2.3 Семестр 4, Типовые оценочные средства для курсовой работы/курсового проектирования по дисциплине

6.2.2.3.1 Описание процедуры

Промежуточная аттестация в виде защиты курсового проекта проводится в последнюю неделю 4-го семестра обучения. По результатам аттестации выставляются оценки: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно». Целью курсового проектирования является закрепление теоретических знаний полученных в процессе изучения курса «Объектно-ориентированное программирование» и развитие практических

навыков применения объектного подхода в ходе решения конкретной практической задачи – разработки приложения с использованием объектно-ориентированного подхода, а также развитие способностей к обоснованному принятию самостоятельных решений в ходе проектной деятельности и получение базового опыта разработки программного обеспечения с использованием объектно-ориентированного подхода.

Задание на курсовое проектирование: студенты должны в соответствии с индивидуальным заданием спроектировать и реализовать приложение (или библиотеку классов) применяя объектно-ориентированный подход к разработке программного обеспечения. Реализованное программное обеспечение обязательно должно быть протестировано.

Тема курсового проекта формулируется руководителем курсового проектирования, но может быть предложена студентом самостоятельно.

По мере выполнения курсового проекта, студент показывают руководителю для предварительной проверки законченные главы пояснительной записки, а также программный код или его фрагменты. Готовые курсовые проекты защищаются студентами. В процессе защиты студент должен ответить на вопросы преподавателя по содержанию разделов пояснительной записки и быть готов пояснить проектные решения и дать комментарии по программному коду.

На защиту студент должен предоставить пояснительную записку по курсовому проекту, содержащую исчерпывающее описание всех разделов курсового проекта и корректное обоснование проектных решений, а также ссылку на репозиторий с исходным кодом проекта, оформленную в соответствии с требованиями СТО ИРНИТУ.

Примерные темы курсовых проектов:

1. Разработка приложения для работы с базой объектов недвижимости.
2. Моделирование работы цеха производящего изделия из набора деталей.
3. Моделирование замкнутой биологической системы (корм, травоядное, хищник).
4. Разработка библиотеки классов для реализации моделей систем массового обслуживания различных типов.
5. Библиотека классов для реализации различных структур данных (например, «Обобщённый массив» позволяющий эффективно хранить данные произвольных типов).
6. Библиотека классов для реализации журнала успеваемости студентов по предмету.
7. Приложение для ведения клиентской базы организации.
8. Разработка библиотеки классов для работы с различными геометрическими объектами.
9. Разработка записной книжки с различными видами записей (контакты, пароли, напоминания и т.п.).
10. Разработка библиотеки классов для реализации искусственной нейронной сети прямого распространения.
11. Приложение для учета товаров на складе.
12. Библиотека классов для автоматического разбора и анализа текста.
13. Приложение для учета кадров в организации.
14. Разработка графической оболочки для экспертной системы на основе CLIPS/Drools.
15. Разработка игрового приложения.
16. Разработка мобильного приложения работающего с удаленными сервисами.

6.2.2.3.2 Критерии оценивания

Отлично	Хорошо	Удовлетворительно	Неудовлетворительно
Продемонстриров	Продемонстриров	Демонстрирует	Неспособность дать

<p>ано глубокое понимание теоретических основ объектно-ориентированного подхода к разработке программного обеспечения. Программная архитектура разделена на слои для обеспечения гибкости и простоты дальнейшего развития функциональности. Бизнес-логика приложения не содержит зависимостей от инфраструктурных слоев и фреймворков. Поставленная задача решена полностью, в процессе решения были применены самостоятельно изученные программные библиотеки и инструментальные средства, использованы дополнительные информационные источники и на них имеются ссылки по тексту пояснительной записки.</p>	<p>ано хорошее понимание теоретических основ объектно-ориентированного подхода к разработке программного обеспечения. Бизнес-логика приложения не содержит зависимостей от классов осуществляющих ввод/вывод данных, а также от классов реализующих пользовательский интерфейс. Поставленная задача решена полностью с применением изученных в рамках курса программных библиотек и инструментальных средств.</p>	<p>понимание теоретических основ объектно-ориентированного подхода к разработке программного обеспечения с незначительными пробелами. Бизнес-логика приложения может содержать зависимости от классов осуществляющих ввод/вывод данных, или от классов реализующих пользовательский интерфейс. Поставленная задача решена частично.</p>	<p>пояснения по пояснительной записке и программному коду, отсутствуют выводы; несоответствие оглавления главам и разделам работы; несогласованность темы работы и её содержания; отсутствие или фальсификация ссылок на литературные источники; грубые ошибки в работе, непонимание сущности излагаемого вопроса, неуверенность и неточность ответов на дополнительные или наводящие вопросы. Поставленная задача не решена.</p>
---	---	---	---

7 Основная учебная литература

1. Васильев А. Н. Java. Объектно-ориентированное программирование : для магистров и бакалавров: базовый курс по объектно-ориентированному программированию / А. Н. Васильев, 2012. - 395.

2. Барков И. А. Объектно-ориентированное программирование : учебник для вузов / И. А. Барков, 2023. - 700.
3. Объектно-ориентированное программирование : методические указания по выполнению лабораторных работ / Иркут. нац. исслед. техн. ун-т, 2017. - 24.

8 Дополнительная учебная литература и справочная

1. Рамбо Дж. UML 2.0. Объектно-ориентированное моделирование и разработка / Дж. Рамбо, М. Блаха, 2007. - 540.
2. Горбунова Т. Н. Объектно-ориентированное программирование и проектирование : учебное пособие / Т. Н. Горбунова, 2011. - 97.
3. Эккель Б. Философия Java : монография / Б. Эккель, 2015. - 1165.
4. Хабибуллин И. Ш. Java 7 : [наиболее полное руководство]: для программистов / И. Ш. Хабибуллин, 2014. - 768.
5. Паттерны проектирования / Э. Фримен, Э. Фримен при участии К. Сьерра и Б. Бейтса, 2015. - 645.

9 Ресурсы сети Интернет

1. <http://library.istu.edu/>
2. <https://e.lanbook.com/>

10 Профессиональные базы данных

Не используются

11 Перечень информационных технологий, лицензионных и свободно распространяемых специализированных программных средств, информационных справочных систем

1. Свободно распространяемое программное обеспечение Свободно распространяемое программное обеспечение Java Development Kit (версия JDK8 или выше)
2. Свободно распространяемое программное обеспечение Свободно распространяемое программное обеспечение JetBrains AndroidStudio (последняя версия)
3. Свободно распространяемое программное обеспечение Свободно распространяемое программное обеспечение JetBrains IntelliJ IDEA Community Edition (последняя версия)

12 Материально-техническое обеспечение дисциплины

1. Проектор Epson EB-460i LCD или аналогичный по техническим характеристикам
2. Компьютер "i5-4440(3.1)/8Gb/500Gb/VGA/23"" или аналогичный по техническим характеристикам: не менее 16 шт.