

Государственное автономное профессиональное образовательное учреждение
«Тетюшский государственный колледж гражданской защиты»

СОГЛАСОВАНО


Председатель наблюдательного совета ГАПОУ
«Тетюшский государственный колледж
гражданской защиты»

 С.А. Фокин

30 июня 2025 г.

УТВЕРЖДАЮ

Директор ГАПОУ «Тетюшский
государственный колледж
гражданской защиты»

 Т.Ю. Адашева/
Приказ № 194-б/д от 30 июня 2025 г.



СОГЛАСОВАНО

Начальник «Тетюшский РУЭС»
ПАО «Таттелефом»

 Хуснутдинов Ш.Р.

30 июня 2025 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
по профессиональному модулю
ПМ.01 РАЗРАБОТКА КОДА ДЛЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

код, наименование профессионального модуля
по специальности

09.02.13 Интеграция решений с применением технологий искусственного интеллекта
код и наименование специальности

2025 г.

Фонд оценочных средств разработан на основе:

- федерального государственного образовательного стандарта среднего профессионального образования по специальности: 09.02.13 Интеграция решений с применением технологий искусственного интеллекта (приказ Министерства просвещения РФ от 24 декабря 2024 г. N 1025).

код и наименование специальности

- рабочей программы профессионального модуля ПМ.01 Разработка кода для искусственного интеллекта

наименование профессионального модуля

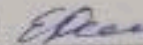
- локальных актов ГАПОУ «Тетюшский государственный колледж гражданской защиты»

Разработчик:

Шакирова Ляйсан Ильгамовна, преподаватель математики и информатики ГАПОУ «Тетюшский государственный колледж гражданской защиты»

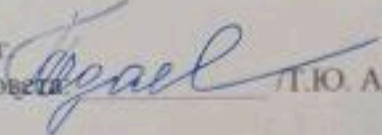
Рассмотрен и одобрен на заседании предметно-цикловой комиссии естественнонаучных дисциплин и математики ГАПОУ «Тетюшский государственный колледж гражданской защиты»

протокол № 4 от 30 июня 2025 г.

председатель ПЦК:  Е.Г. Дороднинова

Рассмотрен педагогическим советом ГАПОУ «Тетюшский государственный колледж гражданской защиты»,

протокол № 9 от 30 июня 2025 г.

председатель педагогического совета:  Т.Ю. Адаева

Общие положения

Результатом освоения профессионального модуля является готовность обучающегося к выполнению вида профессиональной деятельности “Разработка кода для обучения искусственного интеллекта” и составляющих его профессиональных компетенций, а также общие компетенции, формирующиеся в процессе освоения ООП в целом.

Формой аттестации по профессиональному модулю является экзамен (**квалификационный**). Экзамен квалификационный предполагает решение ситуационных задач, теоретических и практических вопросов.

Итогом экзамена является решение: «вид профессиональной деятельности освоен / не освоен» и выставляется оценка.

Комплект ФОС для текущего контроля по МДК ПМ включает контрольно- оценочные материалы для проверки результатов освоения программ теоретического курса МДК., входят в состав учебно-методических комплексов тем МДК, хранятся у преподавателя. Предметом оценки при освоении теоретического курса профессионального модуля являются требования ППСЗ «знания» и «умения», обязательные при реализации программы ПМ. Текущий контроль успеваемости осуществляется в течение семестра, в ходе повседневной учебной работы по индивидуальной инициативе преподавателя.

Промежуточная аттестация проводится с целью установления уровня и качества подготовки обучающихся ФГОС СПО по специальности: 09.02.13 “Интеграция решений с применением технологий искусственного интеллекта” в части требований к результатам освоения программы ПМ ПМ.01 “Разработка кода для искусственного интеллекта”.

1. ФОРМЫ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ

Таблица 1

Элементы модуля, профессиональный модуль	Формы промежуточной аттестации
МДК 01.01 Разработка программных модулей в системах искусственного интеллекта	Экзамен
МДК 01.02. Разработка мобильных приложений с поддержкой искусственного интеллекта	Экзамен
МДК.01.03 Тестирование программных модулей	Дифференцированный зачет
Учебная практика	Дифференцированный зачет
Производственная практика	Дифференцированный зачет
ПМ.01	Экзамен (квалификационный)

2. РЕЗУЛЬТАТЫ ОСВОЕНИЯ МОДУЛЯ, ПОДЛЕЖАЩИЕ ПРОВЕРКЕ

2.1. Общие и профессиональные и компетенции

В результате контроля и оценки по профессиональному модулю осуществляется комплексная проверка следующих общих и профессиональных компетенций:

ОК 01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам

ОК 02 Использовать современные средства поиска, анализа и интерпретации информации и информационные технологии для выполнения задач профессиональной деятельности

ОК 05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста

ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.

- ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием.
 ПК 1.3 Оформлять программный код в соответствии с техническим заданием.
 ПК 1.4 Использовать систему контроля версий программного кода с учетом обеспечения возможности организации групповой разработки.
 ПК 1.5 Выполнять отладку программных модулей с использованием специализированных программных средств.
 ПК 1.6 Выполнять тестирование программного кода.
 ПК 1.7 Составлять тестовые сценарии..

Таблица 2

Профессиональные компетенции	Показатели оценки результата	Тип задания
ПК 1.1	Алгоритм разработан в соответствии с ТЗ, оптимизирован, понятен. Возможны незначительные отклонения. Алгоритм разработан с частичным соответствием ТЗ.	Практическая работа, защита проекта, экзамен (практическое задание)
ПК 1.2	Программный модуль разработан в соответствии с ТЗ, тесты пройдены. Возможны минимальные несоответствия, тесты в целом успешны. Программный модуль разработан с существенными доработками.	Практическая работа, экзамен (практическое задание)
ПК 1.3	Код оформлен в соответствии с требованиями, включая комментарии и стиль кода. Допускаются мелкие недочеты. Код оформлен частично в соответствии с требованиями.	Проверка кода, рецензирование
ПК 1.4	Система контроля версий используется эффективно, изменения фиксируются корректно. Имеются мелкие нарушения порядка фиксации изменений. Система контроля версий используется частично или с ошибками.	Анализ репозитория
ПК 1.5	Отладка выполнена полностью, ошибки устранены, работа модулей оптимизирована. Ошибки устранены, но оптимизация частичная. Ошибки устранены не полностью.	Демонстрация отладки, отчет об ошибках
ПК 1.6	Тестирование выполнено в полном объеме, тесты соответствуют ТЗ, выявленные ошибки исправлены. Тесты соответствуют ТЗ, незначительные ошибки остались. Ошибки выявлены, но не исправлены.	Практическая работа, отчет о тестировании
ПК 1.7	Тестовые сценарии составлены полностью, покрывают все функциональные требования. Покрывают незначительную часть функциональных требований. Покрывают минимальный функционал.	Анализ тестовых сценариев

Таблица 3

Общие компетенции	Показатели оценки результата	Тип задания
ОК 01	Выбор оптимального способа решения задачи, учет контекста.	Наблюдение за работой, оценка решений в практических

		заданиях и проектах
ОК 02	Эффективное использование средств поиска и анализа информации, применение информационных технологий для решения задач.	Оценка использования ресурсов и инструментов, анализ отчетов и презентаций
ОК 05	Грамотное и логичное изложение информации, умение вести дискуссию, соблюдение норм общения.	Оценка устных ответов, оценка письменных работ, оценка презентаций и защиты проектов

Таблица 4

Перечень основных показателей оценки результатов, элементов практического опыта, знаний и умений, подлежащих текущему контролю и промежуточной аттестации

Наименование ПК	Наименование элемента практического опыта	Наименование элемента умений	Наименование элемента знаний
ПК 1.1	Разработка алгоритмов для решения задач машинного обучения.	Анализировать требования, применять методы алгоритмизации, разрабатывать оптимальные алгоритмы.	Методы построения алгоритмов, структуры данных, языки программирования для ИИ.
ПК 1.2	Реализация программных модулей с использованием различных фреймворков.	Реализовывать модули на основе ТЗ, писать чистый код, использовать библиотеки и фреймворки.	Принципы модульного программирования, ЯП для разработки модулей, стандартные фреймворки и библиотеки для работы с ИИ (TensorFlow, PyTorch, Keras).
ПК 1.3	Оформление кода в соответствии со стандартами и требованиями.	Оформлять и документировать код, применять соглашения о наименованиях.	Основные принципы чистого кода, стандарты и практики документирования, инструменты автоматической проверки качества кода.

ПК 1.4	Работа с системой контроля версий (Git, GitLab).	Работать с Git, организовывать совместную работу, разрешать конфликты.	Принципы работы систем контроля версий, основные команды Git, методы разрешения конфликтов.
ПК 1.5	Отладка программных модулей.	Использовать отладчик, идентифицировать и исправлять ошибки, применять логирование.	Принципы работы отладчиков и логирования, способы выявления ошибок, инструменты для отладки кода.
ПК 1.6	Тестирование программного кода.	Проводить тестирование, разрабатывать тестовые сценарии, автоматизировать тестирование.	Принципы тестирования, методы и подходы к написанию тестов, инструменты для тестирования кода.
ПК 1.7	Составление тестовых сценариев.	Определять критические сценарии, разрабатывать пошаговые сценарии, оценивать покрытие тестов.	Основы тест-дизайна, методы разработки тестовых сценариев, принципы проектирования сценариев для функционального и нефункционального тестирования.

2.2. Требования к портфолио

Тип портфолио: портфолио смешанного типа

Документы в портфолио:

Аттестационные лист и характеристика по учебной и производственной практике. (Приложение 1)

Отчет о прохождении учебной и производственной практики. (Приложение 2)

Дневник по учебной и производственной практике. (Приложение 3)

Допуск до промежуточной аттестации по профессиональному модулю (Приложение 4)

Сводная ведомость по итогам профессионального модуля (Приложение 5)

Оценочная ведомость (Приложение 6)

В результате освоения теоретического курса профессионального модуля обучающиеся должны:

иметь практический опыт: Разработки и тестирования программного кода для решения задач в области искусственного интеллекта, работы с системами контроля версий, отладки и профилирования кода.

уметь: Анализировать технические задания, разрабатывать алгоритмы, реализовывать программные модули, оформлять код в соответствии со стандартами, использовать системы контроля версий, отлаживать и тестировать код.

знать: Основные методы и подходы к разработке алгоритмов, принципы модульного программирования, стандарты оформления кода, принципы работы систем контроля версий, методы отладки и тестирования.

3. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ПМ 01 «Разработка кода для искусственного интеллекта»

3.1. ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО МДК 01.01 «Разработка программных модулей в системах искусственного интеллекта»

3.1.1. Оценочное средство для проведения текущего контроля в форме опроса

Вопросы для фронтального опроса

(Раздел 1: Искусственный интеллект и анализ данных)

1. Что такое искусственный интеллект и какие основные направления в нем выделяют? (Тема 1.1)
2. Приведите примеры успешного применения ИИ в реальных задачах. (Тема 1.1)
3. Какие этические вопросы связаны с развитием ИИ? (Тема 1.1)
4. Какие методы сбора данных вы знаете? (Тема 1.2)
5. Почему качество данных важно для ИИ-моделей? (Тема 1.2)
6. Какие методы предобработки данных существуют? (Тема 1.2)
7. Что такое веб-скрапинг и для чего он используется? (Тема 1.2)
8. Зачем нужна нормализация данных? (Тема 1.2)
9. Как обрабатываются пропуски в данных? (Тема 1.2)
10. Что такое кодирование категориальных данных? (Тема 1.2)

(Раздел 2: Алгоритмы и машинное обучение)

11. Какие виды машинного обучения существуют? (Тема 2.1)
12. Назовите основные алгоритмы машинного обучения, относящиеся к обучению с учителем. (Тема 2.1)
13. Что такое линейная регрессия и где она применяется? (Тема 2.1)
14. Что такое логистическая регрессия и чем она отличается от линейной? (Тема 2.1)
15. Объясните принцип работы метода ближайших соседей (kNN). (Тема 2.1)
16. Для чего используются деревья решений? (Тема 2.1)
17. Что такое метод опорных векторов (SVM)? (Тема 2.1)
18. Какие алгоритмы кластеризации вы знаете? (Тема 2.1)
19. В чем суть алгоритма k-means? (Тема 2.1)
20. Как работает агломеративная кластеризация? (Тема 2.1)
21. Как оценивается качество моделей машинного обучения? (Тема 2.2)
22. Что такое точность и полнота в задачах классификации? (Тема 2.2)
23. Что такое F-мера и как она рассчитывается? (Тема 2.2)
24. Что такое ROC-кривая и AUC? (Тема 2.2)
25. Зачем нужна валидация моделей? (Тема 2.2)
26. Какие методы валидации моделей вы знаете? (Тема 2.2)
27. Что такое глубокое обучение и как оно отличается от машинного обучения? (Тема 2.3)
28. Какие архитектуры нейронных сетей вы знаете? (Тема 2.3)
29. Какие процессы обучения нейронных сетей существуют? (Тема 2.3)
30. Какие принципы проектирования архитектуры ИИ-систем вы знаете? (Тема 2.4)

Критерии и нормы оценки устных ответов

«5» за глубокое и полное овладение содержанием учебного материала, в котором обучающиеся легко ориентируются, за умение связывать теорию с практикой, высказывать и

обосновывать свои суждения. Отличная отметка предполагает грамотное, логическое изложение ответа

«4» если обучающийся полно освоил материал, владеет понятийным аппаратом, ориентируется в изученном материале, грамотно излагает ответ, но содержание, форма ответа имеют отдельные недостатки

«3» если обучающийся обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, не умеет доказательно обосновывать свои суждения

«2» если обучающийся имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал

Тестовые задания

Раздел 1: Искусственный интеллект и анализ данных (20 заданий)

Тема 1.1 Введение в искусственный интеллект и его направления (10 заданий)

1. Что из перечисленного НЕ является направлением искусственного интеллекта?
 - а) Машинное обучение
 - б) Обработка естественного языка
 - в) **Веб-дизайн**
 - д) Компьютерное зрение
2. Кто считается одним из основоположников искусственного интеллекта?
 - а) Билл Гейтс
 - б) **Алан Тьюринг**
 - в) Стив Джобс
 - д) Марк Цукерберг
3. Какая область ИИ занимается созданием систем, способных понимать и генерировать текст?
 - а) Компьютерное зрение
 - б) **Обработка естественного языка**
 - в) Робототехника
 - д) Экспертные системы
4. Что такое “сильный ИИ”?
 - а) ИИ, способный решать узкий круг задач
 - б) **ИИ, превосходящий человеческий интеллект во всех областях**
 - в) ИИ, работающий только в облаке
 - д) ИИ, использующий только машинное обучение
5. Какая из этих задач является примером применения компьютерного зрения?
 - а) Перевод текста с одного языка на другой
 - б) **Распознавание лиц на изображении**
 - в) Прогнозирование погоды
 - д) Управление роботом-пылесосом
6. Что такое машинное обучение?
 - а) Раздел ИИ, занимающийся созданием роботов
 - б) **Раздел ИИ, позволяющий системам обучаться на данных без явного программирования**
 - в) Раздел ИИ, занимающийся обработкой текста
 - д) Раздел ИИ, занимающийся распознаванием изображений
7. В чем разница между “слабым” и “сильным” ИИ?
 - а) **“Слабый” ИИ может только имитировать интеллект, а “сильный” - обладает настоящим интеллектом**
 - б) “Слабый” ИИ работает только на простых задачах, а “сильный” - на сложных
 - в) “Слабый” ИИ использует только машинное обучение, а “сильный” - все методы ИИ

- d) “Слабый” ИИ дешевле в разработке, чем “сильный”
- 8. Какая технология используется в беспилотных автомобилях для распознавания дорожных знаков и пешеходов?
 - a) Обработка естественного языка
 - **b) Компьютерное зрение**
 - c) Экспертные системы
 - d) Робототехника
- 9. Что такое нейронная сеть?
 - a) Компьютерная сеть, соединяющая разные устройства
 - **b) Математическая модель, имитирующая структуру и функции биологических нейронных сетей**
 - c) Программа для обработки текстов
 - d) Алгоритм машинного обучения
- 10. Что такое “обучение с подкреплением”?
 - **a) Вид обучения, при котором система получает награду за правильные действия и штраф за неправильные**
 - b) Вид обучения, при котором система получает размеченные данные
 - c) Вид обучения, при котором система не получает никакой информации о правильности своих действий
 - d) Вид обучения, при котором система постоянно обновляет свои знания из внешних источников

Тема 1.2 Методы сбора и предобработки данных (10 заданий)

- 11. Какой метод сбора данных предполагает автоматическое извлечение информации с веб-сайтов?
 - a) API
 - **b) Веб-скрапинг**
 - c) Краудсорсинг
 - d) Ручной ввод
- 12. Что такое API?
 - a) Графический интерфейс пользователя
 - **b) Программный интерфейс приложения**
 - c) Аппаратное обеспечение компьютера
 - d) Метод машинного обучения
- 13. Какая операция относится к очистке данных?
 - a) Нормализация
 - **b) Удаление дубликатов**
 - c) Кодирование
 - d) Кластеризация
- 14. Зачем нужна нормализация данных?
 - a) Чтобы увеличить объем данных
 - **b) Чтобы привести данные к единому масштабу**
 - c) Чтобы удалить выбросы
 - d) Чтобы закодировать категориальные признаки
- 15. Что такое кодирование категориальных данных?
 - a) Преобразование числовых данных в текстовые
 - **b) Преобразование текстовых данных в числовые**
 - c) Удаление категориальных признаков
 - d) Сжатие данных
- 16. Что такое выбросы (outliers) в данных?
 - a) Пропущенные значения
 - **b) Аномальные значения, сильно отличающиеся от основной массы данных**
 - c) Дублирующиеся записи
 - d) Категориальные признаки

17. Какие методы используются для обработки выбросов?
- **a) Удаление, замена на среднее/медиану, трансформация**
 - b) Нормализация, стандартизация
 - c) Кодирование, векторизация
 - d) Кластеризация, классификация
18. Что такое Feature Engineering (генерация признаков)?
- a) Процесс удаления ненужных признаков из данных
 - **b) Процесс создания новых признаков на основе существующих**
 - c) Процесс нормализации данных
 - d) Процесс кодирования данных
19. Зачем нужно Feature Engineering?
- a) Для уменьшения объема данных
 - **b) Для повышения качества моделей машинного обучения**
 - c) Для упрощения анализа данных
 - d) Для визуализации данных
20. Какие существуют стратегии работы с пропущенными значениями в данных?
- a) Удаление строк с пропущенными значениями
 - b) Замена пропущенных значений на среднее/медиану/наиболее часто встречающееся значение
 - c) Использование алгоритмов, устойчивых к пропущенным значениям
 - **d) Все вышеперечисленное**

Раздел 2: Алгоритмы и машинное обучение (40 заданий)

Тема 2.1 Основы алгоритмов машинного обучения (10 заданий)

21. Какой вид машинного обучения предполагает наличие размеченных данных (с правильными ответами)?
- a) Обучение без учителя
 - b) Обучение с подкреплением
 - **c) Обучение с учителем**
 - d) Глубокое обучение
22. Какой алгоритм относится к обучению без учителя?
- a) Линейная регрессия
 - b) Логистическая регрессия
 - **c) K-means**
 - d) Дерево решений
23. Какая задача решается с помощью линейной регрессии?
- a) Классификация
 - b) Кластеризация
 - **c) Регрессия**
 - d) Уменьшение размерности
24. Какой алгоритм используется для классификации, основываясь на расстоянии до ближайших объектов?
- a) Линейная регрессия
 - b) Логистическая регрессия
 - c) K-means
 - **d) K-ближайших соседей (kNN)**
25. Какой алгоритм строит модель в виде дерева решений?
- a) SVM
 - **b) Random Forest**
 - c) Линейная регрессия
 - d) K-means
26. Что такое “обучение с подкреплением”?
- a) Обучение на размеченных данных
 - b) Обучение на неразмеченных данных

- **с) Обучение путем взаимодействия с окружающей средой и получения вознаграждения за правильные действия**
 - d) Обучение на основе анализа текстов
27. Какой алгоритм машинного обучения используется для уменьшения размерности данных?
- a) Линейная регрессия
 - b) Логистическая регрессия
 - **с) PCA (Principal Component Analysis)**
 - d) K-means
28. Что такое “классификация”?
- a) Задача предсказания числового значения
 - b) Задача разделения данных на группы
 - **с) Задача отнесения объекта к одному из предопределенных классов**
 - d) Задача уменьшения размерности данных
29. Что такое “регрессия”?
- **а) Задача предсказания числового значения**
 - b) Задача разделения данных на группы
 - c) Задача отнесения объекта к одному из предопределенных классов
 - d) Задача уменьшения размерности данных
30. Что такое “кластеризация”?
- a) Задача предсказания числового значения
 - **б) Задача разделения данных на группы**
 - c) Задача отнесения объекта к одному из предопределенных классов
 - d) Задача уменьшения размерности данных

Тема 2.2 Оценка качества моделей и улучшение алгоритмов (10 заданий)

31. Что такое точность (accuracy) в задачах классификации?
- a) Доля правильно предсказанных положительных объектов
 - **б) Доля правильно предсказанных объектов**
 - c) Доля объектов, которые алгоритм отнес к положительному классу
 - d) Доля объектов, которые алгоритм пропустил
32. Что такое полнота (recall) в задачах классификации?
- **а) Доля правильно предсказанных положительных объектов**
 - b) Доля правильно предсказанных объектов
 - c) Доля объектов, которые алгоритм отнес к положительному классу
 - d) Доля объектов, которые алгоритм пропустил
33. Что такое F1-мера?
- a) Среднее арифметическое точности и полноты
 - b) Среднее геометрическое точности и полноты
 - **с) Гармоническое среднее точности и полноты**
 - d) Максимум из точности и полноты
34. Что такое кросс-валидация?
- a) Метод обучения на всех данных сразу
 - b) Метод разбиения данных на тренировочную и тестовую выборки
 - **с) Метод многократного разбиения данных на подвыборки для обучения и проверки**
 - d) Метод уменьшения размерности данных
35. Что такое регуляризация?
- a) Метод увеличения сложности модели
 - **б) Метод упрощения модели для предотвращения переобучения**
 - c) Метод нормализации данных
 - d) Метод кодирования категориальных признаков
36. Какая метрика используется для оценки качества моделей регрессии?
- a) Accuracy

- b) Recall
 - **c) Mean Squared Error (MSE)**
 - d) F1-score
37. Что такое переобучение (overfitting)?
- **a) Ситуация, когда модель слишком хорошо подстраивается под тренировочные данные и плохо работает на новых данных**
 - b) Ситуация, когда модель недостаточно хорошо обучается на тренировочных данных
 - c) Ситуация, когда модель работает слишком быстро
 - d) Ситуация, когда модель использует слишком много ресурсов
38. Что такое недообучение (underfitting)?
- a) Ситуация, когда модель слишком хорошо подстраивается под тренировочные данные
 - **b) Ситуация, когда модель недостаточно хорошо обучается на тренировочных данных**
 - c) Ситуация, когда модель работает слишком медленно
 - d) Ситуация, когда модель использует слишком мало ресурсов
39. Какой метод используется для борьбы с переобучением?
- a) Увеличение количества данных
 - b) Упрощение модели
 - c) Регуляризация
 - **d) Все вышеперечисленное**
40. Что такое “матрица ошибок” (confusion matrix)?
- a) Матрица, показывающая структуру нейронной сети
 - **b) Матрица, показывающая результаты классификации и позволяющая оценить ошибки разных типов**
 - c) Матрица, показывающая корреляцию между признаками
 - d) Матрица, показывающая распределение данных

Тема 2.3 Глубокое обучение и нейронные сети (10 заданий)

41. Что такое нейрон?
- a) Единица хранения данных в компьютере
 - **b) Математическая функция, имитирующая работу биологического нейрона**
 - c) Тип компьютерной сети
 - d) Алгоритм кластеризации
42. Что такое функция активации?
- a) Функция, определяющая скорость обучения нейронной сети
 - **b) Функция, определяющая выходное значение нейрона на основе входных данных**
 - c) Функция, определяющая структуру нейронной сети
 - d) Функция, определяющая размер нейронной сети
43. Что такое обратное распространение ошибки?
- a) Метод обучения без учителя
 - b) Метод обучения с подкреплением
 - **c) Алгоритм обучения нейронных сетей, основанный на распространении ошибки от выхода к входу**
 - d) Метод сбора данных
44. Что такое сверточная нейронная сеть (CNN)?
- a) Тип нейронной сети, предназначенный для обработки последовательностей
 - **b) Тип нейронной сети, предназначенный для обработки изображений**
 - c) Тип нейронной сети, предназначенный для обработки текста
 - d) Тип нейронной сети, предназначенный для обработки звука
45. Что такое рекуррентная нейронная сеть (RNN)?

- а) Тип нейронной сети, предназначенный для обработки последовательностей
- б) Тип нейронной сети, предназначенный для обработки изображений
- с) Тип нейронной сети, предназначенный для обработки текста
- д) Тип нейронной сети, предназначенный для обработки звука

Тема 2.4 Проектирование ИИ-систем (10 заданий)

46. Что такое модульность в проектировании ИИ-систем?
- а) Способность системы работать на разных платформах
 - **б) Разделение системы на отдельные, независимые компоненты**
 - с) Способность системы обрабатывать большие объемы данных
 - д) Способность системы адаптироваться к изменениям
47. Что такое масштабируемость в проектировании ИИ-систем?
- а) Способность системы работать на разных платформах
 - б) Разделение системы на отдельные, независимые компоненты
 - **с) Способность системы обрабатывать большие объемы данных и увеличивать производительность**
 - д) Способность системы адаптироваться к изменениям
48. Что такое контейнеризация?
- а) Метод сжатия данных
 - **б) Метод упаковки приложения и его зависимостей в один образ**
 - с) Метод шифрования данных
 - д) Метод управления базами данных
49. Что такое Docker?
- **а) Система контейнеризации**
 - б) Язык программирования
 - с) Фреймворк для машинного обучения
 - д) Операционная система
50. Что такое Kubernetes?
- а) Система контейнеризации
 - б) Фреймворк для машинного обучения
 - **с) Система оркестрации контейнеров**
 - д) Язык программирования
51. Зачем нужна контейнеризация ИИ-систем?
- **а) Для упрощения развертывания и управления ИИ-системами**
 - б) Для повышения безопасности ИИ-систем
 - с) Для уменьшения затрат на разработку ИИ-систем
 - д) Для повышения производительности ИИ-систем
52. Что такое “оркестрация контейнеров”?
- а) Процесс создания контейнеров
 - **б) Процесс управления и автоматизации развертывания, масштабирования и мониторинга контейнеров**
 - с) Процесс тестирования контейнеров
 - д) Процесс обновления контейнеров
53. Какие инструменты используются для мониторинга ИИ-систем?
- а) Docker
 - б) Kubernetes
 - **с) Prometheus, Grafana**
 - д) PyTorch, TensorFlow
54. Какие меры следует принимать для обеспечения безопасности ИИ-систем?
- а) Защита от атак, направленных на манипулирование данными
 - б) Контроль доступа к системе
 - с) Мониторинг активности системы
 - **д) Все вышеперечисленное**

55. Что такое CI/CD в контексте разработки ИИ-систем?
- a) **Continuous Integration/Continuous Delivery - практика автоматизации сборки, тестирования и развертывания кода**
 - b) Computer Intelligence/Critical Data - методология разработки баз данных
 - c) Complex Implementation/Controlled Deployment - процесс сложного внедрения
 - d) Cybernetic Infrastructure/Distributed Computing - кибернетическая инфраструктура
56. Что такое “инфраструктура как код” (Infrastructure as Code, IaC)?
- a) **Подход к управлению инфраструктурой, при котором описание инфраструктуры хранится в виде кода**
 - b) Подход к разработке ИИ-систем, при котором используется модульный код
 - c) Подход к тестированию ИИ-систем, при котором используются автоматизированные тесты
 - d) Подход к обучению ИИ-систем, при котором используются большие объемы данных
57. Что такое А/В-тестирование применительно к ИИ-системам?
- a) Метод тестирования производительности системы при больших нагрузках
 - b) **Метод сравнения двух версий ИИ-системы для определения, какая из них лучше**
 - c) Метод проверки безопасности системы на проникновение
 - d) Метод оценки стоимости разработки и внедрения системы
58. Что такое “объяснимый ИИ” (Explainable AI, XAI)?
- a) **ИИ, который может объяснить, как он принимает решения**
 - b) ИИ, который может генерировать текст на разных языках
 - c) ИИ, который может распознавать лица
 - d) ИИ, который может играть в шахматы
59. Что такое Federated Learning (федеративное обучение)?
- a) Метод обучения модели на централизованном сервере
 - b) **Метод обучения модели на децентрализованных устройствах, сохраняя конфиденциальность данных**
 - c) Метод обучения модели с использованием открытых данных
 - d) Метод обучения модели с использованием только одного источника данных
60. Что такое “цикл обратной связи” (feedback loop) в ИИ-системах?
- a) Процесс сбора данных
 - b) Процесс обучения модели
 - c) Процесс развертывания модели
 - d) **Процесс, в котором результаты работы системы влияют на ее дальнейшее обучение и поведение**

Ключ ответов:

1. c
2. b
3. b
4. b
5. b
6. b
7. a
8. b
9. b
10. a
11. b
12. b
13. b
14. b

- 15. b
- 16. b
- 17. a
- 18. b
- 19. b
- 20. d
- 21. c
- 22. c
- 23. c
- 24. d
- 25. a
- 26. b
- 27. a
- 28. b
- 29. c
- 30. b
- 31. a
- 32. b
- 33. c
- 34. b
- 35. a
- 36. b
- 37. b
- 38. c
- 39. a
- 40. a
- 41. a
- 42. a
- 43. b
- 44. b
- 45. a
- 46. b
- 47. c
- 48. a
- 49. d
- 50. a
- 51. a
- 52. b
- 53. c
- 54. d
- 55. a
- 56. a
- 57. b
- 58. a
- 59. b
- 60. d

Шкала оценки образовательных достижений:		Оценка уровня подготовки Процент результативности (правильных ответов)
Отлично	5	100 – 90 %
Хорошо	4	89 – 80 %
Удовлетворительно	3	79 – 50 %
Неудовлетворительно	2	49 и менее %

3.1.2. Оценочное средство для проведения текущего контроля по результатам практических занятий

Практические задания:

Раздел 1: Искусственный интеллект и анализ данных

Практическая работа №1: Анализ примеров успешных решений на основе ИИ (Тема 1.1)

- Цель: Изучить примеры применения ИИ в различных областях, выявить общие черты и специфические особенности.
- Задачи:
 - Найти и проанализировать 3-5 кейсов успешного внедрения ИИ в различных областях (медицина, финансы, транспорт, образование и т.д.).
 - Определить, какие задачи решаются с помощью ИИ в каждом кейсе.
 - Выделить используемые методы и алгоритмы ИИ.
 - Оценить экономический эффект или социальную значимость внедрения ИИ.
- Результат: Отчет с описанием проанализированных кейсов и выводами о тенденциях и перспективах развития ИИ.

Практическая работа №2: Создание базовой модели ИИ для классификации данных (Тема 1.1)

- Цель: Получить практический опыт создания простой модели машинного обучения для решения задачи классификации.
- Задачи:
 - Выбрать подходящий набор данных для задачи классификации (например, Iris dataset).
 - Разделить данные на тренировочную и тестовую выборки.
 - Выбрать алгоритм классификации (например, логистическая регрессия или kNN).
 - Обучить модель на тренировочной выборке.
 - Оценить качество модели на тестовой выборке (точность, полнота, F1-мера).
- Результат: Рабочий код на Python, реализующий модель классификации, и отчет с результатами оценки качества.

Практическая работа №3: Сбор данных с использованием веб-скрапинга (Тема 1.2)

- Цель: Научиться собирать данные с веб-сайтов с использованием инструментов веб-скрапинга.
- Задачи:
 - Определить веб-сайт для сбора данных (например, интернет-магазин, новостной портал).
 - Изучить структуру веб-сайта и определить элементы, содержащие нужные данные.
 - Написать код на Python с использованием библиотек BeautifulSoup или Scrapy для извлечения данных.
 - Сохранить собранные данные в формате CSV или JSON.
- Результат: Рабочий код на Python, собирающий данные с веб-сайта, и файл с собранными данными.

Практическая работа №4: Предобработка данных для машинного обучения (Тема 1.2)

- Цель: Освоить основные методы предобработки данных для подготовки их к обучению моделей машинного обучения.
- Задачи:
 - Загрузить набор данных (например, из файла CSV).
 - Очистить данные от дубликатов и выбросов.
 - Обработать пропущенные значения.
 - Нормализовать или стандартизировать числовые признаки.

- Закодировать категориальные признаки с использованием one-hot encoding или label encoding.
- Результат: Рабочий код на Python, выполняющий предобработку данных, и описание примененных методов.

Раздел 2: Алгоритмы и машинное обучение

Практическая работа №5: Реализация алгоритма линейной регрессии (Тема 2.1)

- Цель: Реализовать алгоритм линейной регрессии “с нуля” и сравнить его с реализацией из библиотеки scikit-learn.
- Задачи:
 - Реализовать функции для вычисления коэффициентов линейной регрессии (например, методом наименьших квадратов).
 - Реализовать функцию для предсказания значений на основе полученных коэффициентов.
 - Проверить работу реализованного алгоритма на простом наборе данных.
 - Использовать LinearRegression из scikit-learn и сравнить результаты.
- Результат: Рабочий код на Python, реализующий алгоритм линейной регрессии, и отчет с результатами сравнения.

Практическая работа №6: Применение алгоритма k-ближайших соседей (kNN) (Тема 2.1)

- Цель: Изучить и применить алгоритм k-ближайших соседей для задачи классификации.
- Задачи:
 - Выбрать подходящий набор данных для задачи классификации.
 - Реализовать функцию для вычисления расстояния между объектами.
 - Реализовать алгоритм kNN для классификации новых объектов.
 - Оптимизировать параметр k с использованием кросс-валидации.
 - Оценить качество модели на тестовой выборке.
- Результат: Рабочий код на Python, реализующий алгоритм kNN, и отчет с результатами оценки качества.

Практическая работа №7: Обучение и визуализация дерева решений (Тема 2.1)

- Цель: Изучить и применить алгоритм дерева решений для задачи классификации и визуализировать полученное дерево.
- Задачи:
 - Выбрать подходящий набор данных для задачи классификации.
 - Обучить модель дерева решений с использованием DecisionTreeClassifier из scikit-learn.
 - Визуализировать полученное дерево решений с использованием graphviz.
 - Оценить качество модели на тестовой выборке.
 - Проанализировать влияние гиперпараметров дерева на качество модели.
- Результат: Рабочий код на Python, реализующий модель дерева решений, визуализация дерева и отчет с результатами анализа.

Практическая работа №8: Кластеризация данных с использованием k-means (Тема 2.1)

- Цель: Освоить алгоритм кластеризации k-means и применить его для решения задачи сегментации данных.
- Задачи:
 - Выбрать подходящий набор данных для задачи кластеризации.
 - Определить оптимальное количество кластеров с использованием метода локтя (elbow method) или silhouette analysis.
 - Применить алгоритм k-means с использованием KMeans из scikit-learn.
 - Визуализировать результаты кластеризации.
 - Проанализировать полученные кластеры и дать им интерпретацию.
- Результат: Рабочий код на Python, реализующий алгоритм k-means, визуализация кластеров и отчет с анализом результатов.

Практическая работа №9: Оценка качества модели с использованием ROC-кривой и AUC (Тема 2.2)

- Цель: Научиться оценивать качество моделей классификации с использованием ROC-кривой и метрики AUC.
- Задачи:
 - Выбрать подходящий набор данных и обучить модель классификации (например, логистическую регрессию).
 - Вычислить вероятности принадлежности объектов к разным классам.
 - Построить ROC-кривую и вычислить AUC.
 - Проанализировать полученную ROC-кривую и сделать выводы о качестве модели.
- Результат: Рабочий код на Python, строящий ROC-кривую и вычисляющий AUC, и отчет с анализом результатов.

Практическая работа №10: Настройка гиперпараметров модели с использованием GridSearchCV (Тема 2.2)

- Цель: Научиться настраивать гиперпараметры модели с использованием GridSearchCV для повышения ее качества.
- Задачи:
 - Выбрать подходящий набор данных и модель машинного обучения.
 - Определить диапазон значений гиперпараметров для поиска.
 - Использовать GridSearchCV из scikit-learn для поиска оптимальных гиперпараметров.
 - Оценить качество модели с оптимальными гиперпараметрами на тестовой выборке.
 - Сравнить результаты с моделью, обученной с гиперпараметрами по умолчанию.
- Результат: Рабочий код на Python, использующий GridSearchCV для настройки гиперпараметров, и отчет с результатами сравнения.

Практическая работа №11: Создание многослойного персептрона (MLP) с использованием Keras (Тема 2.3)

- Цель: Получить практический опыт создания и обучения многослойного персептрона для решения задачи классификации.
 - Задачи:
 - Выбрать подходящий набор данных для задачи классификации (например, MNIST).
 - Создать модель MLP с использованием Keras.
 - Обучить модель на тренировочной выборке.
 - Оценить качество модели на тестовой выборке.
 - Поэкспериментировать с разными архитектурами сети (количество слоев, количество нейронов в слоях, функции активации) и оптимизаторами.
 - Результат: Рабочий код на Python, реализующий модель MLP, и отчет с результатами экспериментов.

Практическая работа №12: Создание сверточной нейронной сети (CNN) для распознавания изображений (Тема 2.3)

- Цель: Получить практический опыт создания и обучения сверточной нейронной сети для задачи распознавания изображений.
 - Задачи:
 - Выбрать подходящий набор данных для распознавания изображений (например, CIFAR-10).
 - Создать модель CNN с использованием Keras.
 - Обучить модель на тренировочной выборке.
 - Оценить качество модели на тестовой выборке.
 - Поэкспериментировать с разными архитектурами сети (количество слоев, размер фильтров, функции активации, pooling) и оптимизаторами.
 - Результат: Рабочий код на Python, реализующий модель CNN, и отчет с результатами экспериментов.

Практическая работа №13: Использование предобученной модели для классификации изображений (Тема 2.3)

- Цель: Научиться использовать предварительно обученные модели глубокого обучения для решения задач классификации изображений.
 - Задачи:
 - Выбрать предварительно обученную модель (например, VGG16, ResNet50) из Keras.
 - Загрузить веса предварительно обученной модели.
 - Подготовить данные для подачи на вход модели.
 - Получить предсказания модели для новых изображений.
 - Оценить качество работы модели.
 - Результат: Рабочий код на Python, использующий предварительно обученную модель для классификации изображений, и отчет с результатами оценки.

Практическая работа №14: Контейнеризация ИИ-модели с использованием Docker (Тема 2.4)

- Цель: Научиться упаковывать ИИ-модель и ее зависимости в контейнер Docker.
 - Задачи:
 - Создать Dockerfile с инструкциями по установке зависимостей и запуску ИИ-модели.
 - Собрать Docker-образ на основе Dockerfile.
 - Запустить Docker-контейнер на основе созданного образа.
 - Проверить работоспособность ИИ-модели внутри контейнера.
 - Результат: Dockerfile, Docker-образ и инструкция по запуску контейнера с ИИ-моделью.

Практическая работа №15: Развертывание ИИ-системы с использованием Docker Compose (Тема 2.4)

- Цель: Научиться развертывать комплексную ИИ-систему, состоящую из нескольких контейнеров, с использованием Docker Compose.
 - Задачи:
 - Определить компоненты ИИ-системы (например, веб-сервис, база данных, модель машинного обучения).
 - Создать Dockerfile для каждого компонента.
 - Описать структуру системы в файле docker-compose.yml.
 - Запустить систему с использованием команды docker-compose up.
 - Проверить работоспособность системы.
 - Результат: Файлы Dockerfile и docker-compose.yml, и инструкция по развертыванию ИИ-системы.

Бинарная шкала (уровень освоения)	Показатели	Критерии
Зачтено	Полнота выполнения практического задания; Своевременность выполнения задания;	Студентом задание выполнено самостоятельно. При этом составлен правильный алгоритм выполнения задания.
Не зачтено	Последовательность и рациональность выполнения задания;	Студентом задание не выполнено или имеются существенные ошибки в его выполнении.

Оценивание выполнения практических заданий

Отметка "5" Практическая работа выполнена в полном объеме с соблюдением необходимой последовательности. Обучающиеся работали полностью самостоятельно:

подобрали необходимые для выполнения предлагаемых работ источники знаний, показали необходимые для проведения практических и самостоятельных работ теоретические знания, практические умения и навыки. Работа оформлена аккуратно, в оптимальной для фиксации результатов форме.

Отметка "4" Практическая или самостоятельная работа выполнена студентами в полном объеме и самостоятельно. Допускается отклонение от необходимой последовательности выполнения, не влияющее на правильность конечного результата (перестановка пунктов типового плана, последовательность выполняемых заданий, ответы на вопросы). Используются указанные источники знаний. Работа показала знание основного теоретического материала и овладение умениями, необходимыми для самостоятельного выполнения работы. Допускаются неточности и небрежность в оформлении результатов работы.

Отметка "3" Практическая работа выполнена и оформлена с помощью преподавателя. На выполнение работы затрачено много времени (дана возможность доделать работу дома). Студент показал знания теоретического материала, но испытывали затруднения при самостоятельной работе с практическими материалами

Отметка "2" Выставляется в том случае, когда студент оказался не подготовленным к выполнению этой работы. Полученные результаты не позволяют сделать правильных выводов и полностью расходятся с поставленной целью. Обнаружено плохое знание теоретического материала и отсутствие необходимых умений.

3.1.3. Оценочное средство для проведения промежуточной аттестации по МДК 01.01 «Разработка программных модулей в системах искусственного интеллекта»

Теоретические вопросы к экзамену

(Раздел 1: Искусственный интеллект и анализ данных)

1. История и эволюция искусственного интеллекта.
2. Основные направления ИИ: машинное обучение, глубокое обучение, нейронные сети.
3. Методы сбора и предобработки данных.
4. Виды обучения в машинном обучении.
5. Основные алгоритмы машинного обучения.
6. Методы оценки качества моделей.
7. Глубокое обучение и нейронные сети.
8. Принципы проектирования ИИ-систем.
9. Определите понятие искусственного интеллекта. Какие основные подходы и направления в ИИ вы знаете?
10. Опишите историю развития искусственного интеллекта, выделите основные этапы и достижения.
11. Какие существуют методы сбора данных для задач машинного обучения? Опишите их преимущества и недостатки.
12. Какие этапы включает процесс предобработки данных? Зачем нужна предобработка?

13. Что такое очистка данных? Какие методы используются для выявления и удаления ошибок и аномалий?

(Раздел 2: Алгоритмы и машинное обучение)

6. Опишите основные виды машинного обучения: обучение с учителем, обучение без учителя, обучение с подкреплением.
7. Что такое линейная регрессия? Как она используется для решения задач регрессии?
8. Что такое логистическая регрессия? Как она используется для решения задач классификации?
9. Опишите алгоритм k-ближайших соседей (kNN). Как выбирается оптимальное значение k?
10. Что такое дерево решений? Как строится дерево решений?
11. Что такое метод опорных векторов (SVM)? В чем его преимущества и недостатки?
12. Опишите алгоритм кластеризации k-means. Как определить оптимальное количество кластеров?
13. Как оценивается качество моделей классификации? Опишите метрики точность, полнота, F1-мера, ROC-кривая, AUC.
14. Что такое кросс-валидация? Какие методы кросс-валидации вы знаете?
15. Что такое регуляризация? Зачем она нужна? Опишите L1 и L2 регуляризацию.
16. Что такое переобучение (overfitting) и недообучение (underfitting)? Как с ними бороться?

(Раздел 3: Глубокое обучение и проектирование ИИ-систем)

17. Что такое глубокое обучение? Чем оно отличается от традиционного машинного обучения?
18. Что такое нейронная сеть? Опишите структуру и принципы работы нейронной сети.
19. Что такое функция активации? Приведите примеры различных функций активации и опишите их свойства.
20. Что такое обратное распространение ошибки? Как происходит обучение нейронной сети?
21. Опишите основные архитектуры нейронных сетей: многослойный перцептрон (MLP), сверточные нейронные сети (CNN), рекуррентные нейронные сети (RNN).
22. Какие задачи решают CNN? Как работают сверточные слои и слои пулинга?
23. Какие задачи решают RNN? Как работают рекуррентные слои?
24. Какие принципы следует учитывать при проектировании ИИ-систем? (модульность, масштабируемость, эффективность, безопасность)
25. Что такое контейнеризация и оркестрация? Зачем они нужны при развертывании ИИ-систем?

Практические задания к экзамену

(Раздел 1: Искусственный интеллект и анализ данных)

1. Реализовать алгоритм линейной регрессии на заданном наборе данных.
2. Применить алгоритм кластеризации K-means для сегментации данных.
3. Создать и обучить многослойный перцептрон для задачи классификации.
4. Развернуть контейнер с ИИ-моделью с использованием Docker.
5. Загрузить данные из CSV-файла, удалить дубликаты и вывести первые 10 строк.

6. Обработать пропущенные значения в заданном наборе данных, заполнив их средним значением по столбцу.
7. Нормализовать числовые признаки в заданном наборе данных, используя `MinMaxScaler`.
8. Закодировать категориальные признаки в заданном наборе данных, используя `OneHotEncoder`.
9. Разделить набор данных на тренировочную и тестовую выборки в соотношении 80/20.

(Раздел 2: Алгоритмы и машинное обучение)

6. Реализовать модель линейной регрессии для предсказания цены дома на основе его площади.
7. Реализовать модель логистической регрессии для классификации клиентов на “склонных к оттоку” и “не склонных к оттоку”.
8. Обучить модель `kNN` для классификации изображений цифр из набора данных `MNIST`.
9. Построить дерево решений для классификации ирисов из набора данных `Iris`.
10. Обучить модель `SVM` для классификации клиентов на “высокодоходных” и “низкодоходных”.
11. Применить алгоритм `k-means` для кластеризации клиентов на основе их покупательского поведения.
12. Оценить качество модели классификации (логистической регрессии) с использованием матрицы ошибок.
13. Вычислить точность, полноту и `F1`-меру для заданной модели классификации.
14. Построить `ROC`-кривую и вычислить `AUC` для заданной модели классификации.
15. Применить кросс-валидацию для оценки качества модели линейной регрессии.

(Раздел 3: Глубокое обучение и проектирование ИИ-систем)

16. Создать модель многослойного персептрона (`MLP`) для классификации изображений с использованием `Keras`.
17. Обучить созданную модель `MLP` на наборе данных `MNIST`.
18. Оценить качество обученной модели `MLP` на тестовой выборке.
19. Создать модель сверточной нейронной сети (`CNN`) для классификации изображений с использованием `Keras`.
20. Обучить созданную модель `CNN` на наборе данных `CIFAR-10`.
21. Оценить качество обученной модели `CNN` на тестовой выборке.
22. Загрузить предварительно обученную модель (например, `VGG16`) из `Keras` и использовать ее для классификации новых изображений.
23. Создать `Dockerfile` для контейнеризации модели машинного обучения.
24. Запустить `Docker`-контейнер с моделью машинного обучения.
25. Развернуть простую ИИ-систему, состоящую из веб-сервиса и модели машинного обучения, с использованием `Docker Compose`.

3.2. ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО МДК 01.02 «Разработка мобильных приложений с поддержкой искусственного интеллекта»

3.2.1. Оценочное средство для проведения текущего контроля в форме опроса

Вопросы для фронтального опроса

(Раздел 1: Основы мобильной разработки)

1. Какие основные платформы для разработки мобильных приложений вы знаете? (Android, iOS)
2. Какие языки программирования используются для разработки Android-приложений? (Kotlin, Java)
3. Что такое Android Studio и для чего она нужна?
4. Какие основные компоненты пользовательского интерфейса Android вы знаете? (TextView, EditText, Button, ImageView, RecyclerView)
5. Что такое TensorFlow Lite и зачем он используется в мобильных приложениях?
6. Какие предобученные модели ИИ можно использовать в мобильных приложениях? (Распознавание изображений, текста, речи)
7. Как оптимизировать модели ИИ для работы на мобильных устройствах?
8. Как разрабатывать интуитивно понятный пользовательский интерфейс для мобильного приложения?
9. Как взаимодействовать с сенсорами устройства для получения данных?
10. Какие инструменты используются для тестирования мобильных приложений?

(Раздел 2: Тестирование и развертывание мобильных ИИ-приложений)

11. Что такое системы контроля версий и зачем они нужны? (Git, GitLab)
12. Как автоматизировать тестирование мобильных приложений? (Espresso, Appium)
13. Как развертывать приложения в Play Market и App Store?
14. Какие этапы включает процесс развертывания мобильного приложения?
15. Какие особенности необходимо учитывать при тестировании мобильных приложений с ИИ?
16. Как подготовить релизную версию приложения?
17. Как осуществляется поддержка выпущенного приложения?
18. Что такое Google Play Console и App Store Connect?
19. Какие метрики используются для оценки эффективности мобильного приложения?
20. Какие методы продвижения мобильных приложений вы знаете?

Критерии и нормы оценки устных ответов

«5» за глубокое и полное овладение содержанием учебного материала, в котором обучающиеся легко ориентируются, за умение связывать теорию с практикой, высказывать и обосновывать свои суждения. Отличная отметка предполагает грамотное, логическое изложение ответа

«4» если обучающийся полно освоил материал, владеет понятийным аппаратом, ориентируется в изученном материале, грамотно излагает ответ, но содержание, форма ответа имеют отдельные недостатки

«3» если обучающийся обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, не умеет доказательно обосновывать свои суждения

«2» если обучающийся имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал

Тестовые задания

1. Какой язык программирования является предпочтительным для разработки Android-приложений?
 - a) Swift
 - b) Objective-C
 - c) **Kotlin**
 - d) C#
2. Что такое Android SDK?
 - a) **Набор инструментов для разработки Android-приложений**
 - b) Операционная система для мобильных устройств
 - c) Магазин приложений для Android
 - d) Язык программирования для Android
3. Какая библиотека используется для работы с моделями машинного обучения в Android?
 - a) OpenCV
 - b) **TensorFlow Lite**
 - c) NumPy
 - d) Pandas
4. Какой компонент Android отвечает за отображение текста на экране?
 - a) EditText
 - b) Button
 - c) **TextView**
 - d) ImageView
5. Что такое “Intent” в Android?
 - a) Компонент пользовательского интерфейса
 - b) **Механизм для запуска Activity, Service, Broadcast Receiver**
 - c) Файл конфигурации приложения
 - d) Класс для работы с базами данных
6. Что такое “Activity” в Android?
 - a) Компонент пользовательского интерфейса
 - b) **Отдельный экран приложения**
 - c) Сервис, работающий в фоновом режиме
 - d) Файл ресурсов
7. Какой формат используется для хранения моделей машинного обучения в TensorFlow Lite?
 - a) .h5
 - b) .pb
 - c) **.tflite**
 - d) .onnx
8. Что такое “эмпирическое тестирование” мобильных приложений?
 - a) Тестирование кода разработчиком
 - b) Автоматическое тестирование с использованием специальных инструментов
 - c) **Тестирование пользователями (бета-тестирование)**
 - d) Статический анализ кода
9. Что такое “UI/UX” в контексте разработки мобильных приложений?
 - a) Набор инструментов для разработки пользовательского интерфейса
 - b) Процесс оптимизации производительности приложения
 - c) **User Interface/User Experience - удобство и привлекательность пользовательского интерфейса**
 - d) Способ подключения к сети Интернет
10. Какой инструмент используется для автоматизации UI-тестов в Android?
 - a) JUnit

- b) Mockito
- c) Espresso
- d) Robolectric

Шкала оценки образовательных достижений:		Оценка уровня подготовки Процент результативности (правильных ответов)
Отлично	5	100 – 90 %
Хорошо	4	89 – 80 %
Удовлетворительно	3	79 – 50 %
Неудовлетворительно	2	49 и менее %

3.2.2. Оценочное средство для проведения текущего контроля по результатам практических занятий

Практические задания:

Практическая работа №1: Создание простого Android-приложения с использованием Android Studio. (Раздел 1, Тема 1.1)

- **Цель:** Освоить базовые навыки работы с Android Studio, создать проект, добавить элементы UI.
- **Задачи:**
 1. Создать новый проект в Android Studio с использованием шаблона “Empty Activity”.
 2. Изменить layout-файл (activity_main.xml) и добавить текстовое поле (TextView) и кнопку (Button).
 3. Изменить текст в TextView на “Hello, World!”.
 4. Настроить кнопку так, чтобы при нажатии выводилось сообщение (Toast) на экран.
 5. Запустить приложение на эмуляторе или реальном устройстве.
- **Ожидаемый результат:** Рабочее Android-приложение с текстовым полем и кнопкой, выводящее сообщение при нажатии на кнопку.

Практическая работа №2: Разработка пользовательского интерфейса для мобильного приложения с использованием различных компонентов. (Раздел 1, Тема 1.1)

- **Цель:** Научиться создавать различные layout-ы, добавлять и настраивать компоненты UI.
- **Задачи:**
 1. Создать новый проект в Android Studio.
 2. Использовать LinearLayout, RelativeLayout и ConstraintLayout для создания различных layout-ов.
 3. Добавить в layout-ы следующие компоненты UI: TextView, EditText, Button, ImageView, CheckBox, RadioButton, Spinner.
 4. Настроить свойства компонентов UI (текст, размер, цвет, расположение и т.д.).
 5. Реализовать обработку событий (например, нажатие на кнопку, изменение текста в EditText).
- **Ожидаемый результат:** Рабочее Android-приложение с различными layout-ами и компонентами UI, реализующее базовые функции.

Практическая работа №3: Интеграция TensorFlow Lite модели в Android-приложение для классификации изображений. (Раздел 1, Тема 1.2)

- **Цель:** Научиться интегрировать TensorFlow Lite в Android-приложение, использовать предобученную модель для классификации.
- **Задачи:**

1. Скачать TensorFlow Lite модель для классификации изображений (например, MobileNet).
 2. Добавить модель в папку assets проекта.
 3. Импортировать библиотеку TensorFlow Lite в проект.
 4. Реализовать код для загрузки модели и подготовки изображения для классификации.
 5. Реализовать код для запуска модели и получения результатов классификации.
 6. Отобразить результаты классификации на экране приложения.
- **Ожидаемый результат:** Рабочее Android-приложение, классифицирующее изображения с использованием TensorFlow Lite.

Практическая работа №4: Оптимизация TensorFlow Lite модели для работы на мобильном устройстве. (Раздел 1, Тема 1.2)

- **Цель:** Научиться оптимизировать модель для уменьшения размера и повышения производительности.
- **Задачи:**
 1. Использовать quantization (квантование) для уменьшения размера модели.
 2. Использовать GPU delegate для ускорения вычислений на GPU.
 3. Сравнить производительность модели до и после оптимизации (время классификации, использование памяти).
 4. Оценить влияние оптимизации на точность классификации.
- **Ожидаемый результат:** Оптимизированная TensorFlow Lite модель и отчет с результатами сравнения производительности и точности.

Практическая работа №5: Разработка мобильного приложения для распознавания объектов в реальном времени с использованием камеры устройства. (Раздел 1, Тема 1.3)

- **Цель:** Научиться взаимодействовать с камерой устройства, использовать модель машинного обучения для распознавания объектов в реальном времени.
- **Задачи:**
 1. Получить доступ к камере устройства.
 2. Реализовать предварительный просмотр изображения с камеры на экране приложения.
 3. Интегрировать TensorFlow Lite модель для распознавания объектов.
 4. Обрабатывать изображения с камеры и передавать их на вход модели.
 5. Отображать результаты распознавания (bounding boxes, labels, confidence scores) на экране.
- **Ожидаемый результат:** Рабочее Android-приложение, распознающее объекты в реальном времени с использованием камеры.

Практическая работа №6: Создание приложения, использующего API для получения данных и отображения их в интерфейсе. (Раздел 1, Тема 1.3)

- **Цель:** Научиться работать с асинхронными запросами, получать данные из внешних источников и отображать их в UI.
- **Задачи:**
 1. Выбрать API для получения данных (например, API для получения погоды, новостей, курса валют).
 2. Использовать библиотеку Retrofit или Volley для выполнения асинхронных запросов к API.
 3. Обработать полученные данные (JSON или XML).
 4. Отобразить данные в интерфейсе приложения (например, в ListView или RecyclerView).
 5. Реализовать обработку ошибок при выполнении запросов.
- **Ожидаемый результат:** Рабочее Android-приложение, отображающее данные из API.

Практическая работа №7: Использование Git для управления версиями проекта. (Раздел 2, Тема 2.1)

- **Цель:** Освоить базовые команды Git, научиться создавать репозиторий, фиксировать изменения, создавать ветки.
- **Задачи:**
 1. Создать локальный Git-репозиторий для проекта.
 2. Добавить файлы проекта в репозиторий (git add).
 3. Зафиксировать изменения (git commit) с информативным сообщением.
 4. Создать новую ветку (git branch).
 5. Переключиться на новую ветку (git checkout).
 6. Внести изменения в код в новой ветке.
 7. Зафиксировать изменения в новой ветке.
 8. Слить изменения из новой ветки в основную ветку (git merge).
 9. Разрешить возможные конфликты при слиянии.
 10. Отправить репозиторий на GitHub или GitLab.
- **Ожидаемый результат:** Рабочий Git-репозиторий с историей изменений, ветками и разрешенными конфликтами, размещенный на GitHub или GitLab.

Практическая работа №8: Автоматизация тестирования мобильного приложения с использованием Espresso. (Раздел 2, Тема 2.1)

- **Цель:** Научиться писать автоматизированные тесты для проверки UI и функциональности приложения.
- **Задачи:**
 1. Добавить зависимость Espresso в проект.
 2. Написать тесты для проверки UI (например, наличие элементов, текст элементов, нажатие на кнопки).
 3. Написать тесты для проверки функциональности приложения (например, правильность вычислений, корректность работы с API).
 4. Запустить тесты и убедиться, что они проходят успешно.
- **Ожидаемый результат:** Набор автоматизированных тестов, покрывающий основные функции приложения, и отчет о прохождении тестов.

Бинарная шкала (уровень освоения)	Показатели	Критерии
Зачтено	Полнота выполнения практического задания; Своевременность выполнения задания;	Студентом задание выполнено самостоятельно. При этом составлен правильный алгоритм выполнения задания.
Не зачтено	Последовательность и рациональность выполнения задания;	Студентом задание не выполнено или имеются существенные ошибки в его выполнении.

Оценивание выполнения практических заданий

Отметка "5" Практическая работа выполнена в полном объеме с соблюдением необходимой последовательности. Обучающиеся работали полностью самостоятельно: подобрали необходимые для выполнения предлагаемых работ источники знаний, показали необходимые для проведения практических и самостоятельных работ теоретические знания,

практические умения и навыки. Работа оформлена аккуратно, в оптимальной для фиксации результатов форме.

Отметка "4" Практическая или самостоятельная работа выполнена студентами в полном объеме и самостоятельно. Допускается отклонение от необходимой последовательности выполнения, не влияющее на правильность конечного результата (перестановка пунктов типового плана, последовательность выполняемых заданий, ответы на вопросы). Используются указанные источники знаний. Работа показала знание основного теоретического материала и овладение умениями, необходимыми для самостоятельного выполнения работы. Допускаются неточности и небрежность в оформлении результатов работы.

Отметка "3" Практическая работа выполнена и оформлена с помощью преподавателя. На выполнение работы затрачено много времени (дана возможность доделать работу дома). Студент показал знания теоретического материала, но испытывали затруднения при самостоятельной работе с практическими материалами

Отметка "2" Выставляется в том случае, когда студент оказался не подготовленным к выполнению этой работы. Полученные результаты не позволяют сделать правильных выводов и полностью расходятся с поставленной целью. Обнаружено плохое знание теоретического материала и отсутствие необходимых умений.

3.2.3. Оценочное средство для проведения промежуточной аттестации по МДК 01.02 «Разработка мобильных приложений с поддержкой искусственного интеллекта»

Теоретические вопросы к экзамену

1. Опишите архитектуру Android-приложения.
2. Что такое жизненный цикл Activity и как им управлять?
3. Как реализовать многопоточность в Android?
4. Как использовать SQLite для хранения данных в Android-приложении?
5. Как использовать камеры и датчики мобильного устройства в Android-приложении?
6. Объясните процесс интеграции TensorFlow Lite в Android-приложение.
7. Какие методы оптимизации моделей машинного обучения для мобильных устройств вы знаете?
8. Как обеспечить безопасность мобильного приложения?
9. Какие существуют подходы к тестированию мобильных приложений с использованием ИИ?
10. Опишите процесс развертывания мобильного приложения в Google Play Store.
11. Что такое “разрешения” в Android и как с ними работать?
12. Как создать адаптивный пользовательский интерфейс в Android?
13. Как обрабатывать исключения и ошибки в Android-приложениях?
14. Что такое “сервисы” (Services) в Android и для чего они нужны?
15. Как работать с многопоточностью в Android?
16. Что такое “Broadcast Receivers” в Android и как их использовать?
17. Как хранить данные в Android-приложении? (SharedPreferences, SQLite)
18. Как реализовать взаимодействие с пользователем через уведомления (Notifications)?

19. Какие особенности необходимо учитывать при разработке приложений для разных версий Android?
20. Как проводить отладку (debugging) Android-приложения?
21. Что такое “адаптивный дизайн” и как его реализовать в Android?
22. Как локализовать Android-приложение для разных языков?
23. Как использовать Firebase в Android-приложении? (Authentication, Realtime Database, Cloud Firestore)
24. Какие инструменты и библиотеки используются для работы с моделями машинного обучения в Android?
25. Как тестировать производительность Android-приложения?

Практические задания к экзамену

1. Создайте Android-приложение, отображающее список данных из API.
2. Создайте Android-приложение, реализующее классификацию изображений с использованием TensorFlow Lite.
3. Создайте Android-приложение, использующее камеру для распознавания объектов в реальном времени.
4. Оптимизируйте заданную TensorFlow Lite модель для уменьшения размера и повышения производительности.
5. Реализуйте систему автоматизированного тестирования для Android-приложения с использованием Espresso.
6. Добавьте возможность локализации в существующее Android-приложение.
7. Интегрируйте Firebase Authentication в Android-приложение.
8. Создайте Android-приложение, которое использует Firebase Realtime Database для хранения и синхронизации данных.
9. Реализуйте систему уведомлений в Android-приложении.
10. Создайте пользовательский компонент UI в Android.
11. Создайте анимацию в Android.
12. Разработайте UI для входа в приложение используя Material Design.
13. Добавьте поддержку нескольких языков.
14. Создайте систему сохранения настроек пользователя используя SharedPreferences.
15. Реализуйте чтение данных с датчика (например, акселерометра).
16. Реализуйте функциональность работы с геолокацией.
17. Создайте многопоточную загрузку данных из сети.
18. Сделайте программу, которая сохраняет фотографии в память телефона.
19. Подключите в проект библиотеку для работы с графиками.
20. Реализуйте push-уведомления.
21. Разверните Android-приложение на эмуляторе.
22. Разверните Android-приложение на реальном устройстве.
23. Выполните профилировку и найдите проблемные места.
24. Сделайте unit-тесты для важного функционала.
25. Напишите документацию к проекту.

3.3. ОЦЕНОЧНЫЕ СРЕДСТВА ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО

МДК 01.03 «Тестирование программных модулей»

3.3.1. Оценочное средство для проведения текущего контроля в форме опроса

Вопросы для фронтального опроса

1. Что такое тестирование программного обеспечения?
2. Какие цели преследует тестирование ПО?
3. Какие виды тестирования вы знаете? (юнит-тестирование, интеграционное тестирование, системное тестирование, приемочное тестирование, регрессионное тестирование, тестирование производительности, тестирование безопасности и др.)

4. Что такое юнит-тест?
5. Что такое интеграционное тестирование?
6. Что такое системное тестирование?
7. Что такое приемочное тестирование?
8. Что такое регрессионное тестирование?
9. Что такое тестирование производительности?
10. Что такое тестирование безопасности?
11. Что такое “черный ящик” и “белый ящик” тестирование?
12. Что такое тестовый сценарий (test scenario)?
13. Что такое тестовый случай (test case)?
14. Что такое тестовый набор (test suite)?
15. Что такое тестовые данные?
16. Что такое баг-репорт (bug report)?
17. Что такое жизненный цикл бага (bug lifecycle)?
18. Что такое тестовая документация?
19. Какие инструменты используются для автоматизации тестирования? (Selenium, JUnit, TestNG, pytest, Appium, Jenkins, GitLab CI и др.)
20. Что такое CI/CD и как тестирование встраивается в этот процесс?
21. В чем особенности тестирования ИИ-модулей?
22. Что такое метрики качества моделей ИИ? (точность, полнота, F-мера, ROC-кривые)
23. Какие методы используются для оценки качества моделей ИИ?
24. Как автоматизировать тестирование ИИ-систем?
25. Что такое тестирование мобильных ИИ-приложений и какие особенности оно имеет?

Критерии и нормы оценки устных ответов

«5» за глубокое и полное овладение содержанием учебного материала, в котором обучающиеся легко ориентируются, за умение связывать теорию с практикой, высказывать и обосновывать свои суждения. Отличная отметка предполагает грамотное, логическое изложение ответа

«4» если обучающийся полно освоил материал, владеет понятийным аппаратом, ориентируется в изученном материале, грамотно излагает ответ, но содержание, форма ответа имеют отдельные недостатки

«3» если обучающийся обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, не умеет доказательно обосновывать свои суждения

«2» если обучающийся имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал

Тестовые задания

1. Какой вид тестирования проверяет отдельные компоненты (модули) ПО?
 - a) Интеграционное тестирование
 - b) Системное тестирование
 - **с) Юнит-тестирование**
 - d) Приемочное тестирование
2. Какой вид тестирования проверяет взаимодействие между несколькими компонентами ПО?
 - a) Юнит-тестирование
 - b) Системное тестирование
 - **с) Интеграционное тестирование**
 - d) Приемочное тестирование
3. Какой вид тестирования проверяет ПО как единое целое, в соответствии с требованиями?

- a) Юнит-тестирование
 - b) Интеграционное тестирование
 - c) Приемочное тестирование
 - **d) Системное тестирование**
4. Какой вид тестирования проводится для проверки, что изменения в коде не привели к новым ошибкам?
- a) Юнит-тестирование
 - b) Интеграционное тестирование
 - c) Системное тестирование
 - **d) Регрессионное тестирование**
5. Что такое “черный ящик” тестирование?
- a) Тестирование кода разработчиком
 - b) Тестирование с использованием отладчика
 - **c) Тестирование без знания внутренней структуры ПО**
 - d) Тестирование с полным знанием внутренней структуры ПО
6. Что такое тестовый сценарий (test scenario)?
- a) Набор тестовых случаев
 - **b) Последовательность действий для проверки определенной функциональности**
 - c) Файл с тестовыми данными
 - d) Отчет об ошибке
7. Что такое тестовый случай (test case)?
- a) Общее описание проверяемой функциональности
 - **b) Конкретный набор входных данных, ожидаемый результат и шаги для проверки**
 - c) Отчет об ошибке
 - d) Набор тестовых сценариев
8. Что такое баг-репорт (bug report)?
- a) Отчет об успешном прохождении тестов
 - **b) Описание обнаруженной ошибки в ПО**
 - c) Тестовый сценарий
 - d) Тестовые данные
9. Какой инструмент используется для автоматизации тестирования веб-приложений?
- a) JUnit
 - b) TestNG
 - **c) Selenium**
 - d) pytest
10. Что такое CI/CD?
- a) Система управления базами данных
 - **b) Процесс автоматизации сборки, тестирования и развертывания ПО**
 - c) Язык программирования
 - d) Система контроля версий
11. Какая метрика используется для оценки качества модели машинного обучения?
- a) Code Coverage
 - b) Branch Coverage
 - **c) Accuracy**
 - d) Cyclomatic Complexity
12. Что такое “юнит-тестирование” в машинном обучении?
- a) Тестирование работы приложения на мобильном устройстве
 - **b) Тестирование отдельных функций и методов модели**
 - c) Тестирование интеграции модели с другими системами
 - d) Тестирование пользовательского интерфейса
13. Что такое “адверсарные атаки” (adversarial attacks) в машинном обучении?

- а) Атаки на серверы, где хранятся данные
 - **б) Специально разработанные входные данные, приводящие к неправильной работе модели**
 - с) Вирусы, заражающие код модели
 - д) Атаки на конфиденциальность данных пользователей
14. Что такое “тест на устойчивость” (robustness testing) для ИИ-моделей?
- а) Тестирование скорости работы модели
 - **б) Тестирование модели на данных, отличающихся от тренировочных**
 - с) Тестирование потребления памяти моделью
 - д) Тестирование модели на разных платформах
15. Что такое “тестирование справедливости” (fairness testing) в ИИ?
- а) Проверка соответствия модели требованиям законодательства
 - **б) Проверка отсутствия предвзятости в работе модели по отношению к различным группам пользователей**
 - с) Проверка точности модели на разных наборах данных
 - д) Проверка стабильности работы модели во времени

Шкала оценки образовательных достижений:		Оценка уровня подготовки Процент результативности (правильных ответов)
Отлично	5	100 – 90 %
Хорошо	4	89 – 80 %
Удовлетворительно	3	79 – 50 %
Неудовлетворительно	2	49 и менее %

3.3.2. Оценочное средство для проведения текущего контроля по результатам практических занятий

Практические задания:

1. **Практическая работа №1:** Написание юнит-тестов для функции на Python с использованием pytest. (Раздел 1, Тема 1.1)
 - Цель: Освоить основы юнит-тестирования с использованием pytest.
 - **Задачи:**
 - a. Написать функцию на Python, выполняющую простое вычисление (например, сложение, вычитание, умножение, деление).
 - b. Создать тестовый файл с использованием pytest.
 - c. Написать несколько юнит-тестов для проверки различных сценариев работы функции (положительные числа, отрицательные числа, ноль, граничные значения).
 - d. Запустить тесты с использованием pytest и убедиться, что все тесты проходят успешно.
2. **Практическая работа №2:** Написание юнит-тестов для класса на Java с использованием JUnit. (Раздел 1, Тема 1.1)
 - Цель: Освоить основы юнит-тестирования с использованием JUnit.
 - **Задачи:**
 - a. Создать класс на Java с несколькими методами.
 - b. Создать тестовый класс с использованием JUnit.
 - c. Написать несколько юнит-тестов для проверки различных сценариев работы методов класса (положительные значения, отрицательные значения, исключения).
 - d. Запустить тесты с использованием JUnit и убедиться, что все тесты проходят успешно.

3. **Практическая работа №3:** Анализ покрытия кода юнит-тестами с использованием Coverage.py. (Раздел 1, Тема 1.1)
 - Цель: Научиться оценивать покрытие кода тестами и улучшать качество тестирования.
 - **Задачи:**
 - a. Выбрать модуль Python с написанными юнит-тестами.
 - b. Запустить тесты с использованием Coverage.py и получить отчет о покрытии кода.
 - c. Проанализировать отчет и выявить участки кода, не покрытые тестами.
 - d. Написать дополнительные тесты для увеличения покрытия кода.
 - e. Повторно запустить Coverage.py и убедиться, что покрытие кода увеличилось.
4. **Практическая работа №4:** Разработка интеграционных тестов для проверки взаимодействия между двумя модулями Python. (Раздел 1, Тема 1.2)
 - Цель: Освоить интеграционное тестирование и проверку взаимодействия между компонентами ПО.
 - **Задачи:**
 - a. Разработать два модуля Python, взаимодействующих друг с другом (например, один модуль генерирует данные, другой - обрабатывает).
 - b. Написать интеграционные тесты для проверки правильности передачи данных между модулями.
 - c. Использовать Mock-объекты для имитации поведения зависимостей.
 - d. Запустить тесты и убедиться, что все тесты проходят успешно.
5. **Практическая работа №5:** Разработка системных тестов для проверки веб-приложения с использованием Selenium. (Раздел 1, Тема 1.2)
 - Цель: Освоить автоматизированное тестирование веб-приложений с использованием Selenium.
 - **Задачи:**
 - a. Выбрать веб-приложение для тестирования.
 - b. Написать тесты с использованием Selenium для проверки основных функций приложения (например, авторизация, добавление товара в корзину, оформление заказа).
 - c. Использовать различные локаторы для поиска элементов на странице (ID, CSS-селекторы, XPath).
 - d. Запустить тесты и убедиться, что все тесты проходят успешно.
6. **Практическая работа №6:** Написание баг-репорта (bug report) на обнаруженную ошибку в ПО. (Раздел 1, Тема 1.3)
 - Цель: Научиться правильно описывать обнаруженные ошибки в ПО.
 - **Задачи:**
 - a. Выбрать ПО для тестирования.
 - b. Протестировать ПО и обнаружить ошибку.
 - c. Написать баг-репорт, содержащий следующую информацию:
 - Заголовок (краткое описание проблемы)
 - Описание (подробное описание проблемы, шаги для воспроизведения)
 - Ожидаемый результат
 - Фактический результат
 - Версия ПО
 - Операционная система
 - Скриншоты или видео (при необходимости)
7. **Практическая работа №7:** Интеграция юнит-тестов в процесс CI/CD с использованием Jenkins. (Раздел 2, Тема 2.1)
 - Цель: Автоматизировать запуск юнит-тестов при каждом изменении кода.

- **Задачи:**
 - a. Настроить Jenkins для автоматической сборки проекта при каждом изменении кода в репозитории.
 - b. Добавить шаг в Jenkins-билд для запуска юнит-тестов.
 - c. Настроить отправку уведомлений о результатах тестирования.
 - d. Проверить, что юнит-тесты запускаются автоматически и результаты отображаются в Jenkins.
- 8. **Практическая работа №8:** Тестирование API с использованием Postman. (Раздел 2, Тема 2.1)
 - Цель: Научиться тестировать API и проверять правильность ответов сервера.
 - **Задачи:**
 - a. Выбрать API для тестирования.
 - b. Импортировать API в Postman (например, из OpenAPI/Swagger спецификации).
 - c. Создать наборы запросов для тестирования различных endpoints API.
 - d. Проверить корректность ответов сервера (код ответа, структура данных, значения полей).
 - e. Использовать Postman для автоматизации тестирования API (например, написание скриптов для проверки данных).
- 9. **Практическая работа №9:** Проведение нагрузочного тестирования веб-приложения с использованием JMeter. (Раздел 2, Тема 2.1)
 - Цель: Научиться оценивать производительность веб-приложения при высоких нагрузках.
 - **Задачи:**
 - a. Разработать план нагрузочного тестирования в JMeter.
 - b. Настроить количество пользователей, время выполнения теста, параметры запросов.
 - c. Запустить нагрузочное тестирование и собрать статистику (время ответа, количество запросов в секунду, ошибки).
 - d. Проанализировать результаты тестирования и выявить узкие места в приложении.
- 10. **Практическая работа №10:** Анализ результатов тестирования производительности и выявление узких мест. (Раздел 2, Тема 2.1)
 - Цель: Развить навыки анализа результатов нагрузочного тестирования.
 - **Задачи:**
 - a. Использовать данные, полученные в результате нагрузочного тестирования (время ответа, количество запросов в секунду, ошибки).
 - b. Определить, какие компоненты системы являются “узким местом”, ограничивающим производительность.
 - c. Предложить рекомендации по оптимизации производительности (например, кэширование, оптимизация запросов к базе данных, увеличение ресурсов сервера).
- 11. **Практическая работа №11:** Разработка тестовых сценариев для функционального тестирования мобильного приложения. (Раздел 2, Тема 2.1)
 - Цель: Развить навыки разработки тестовых сценариев для мобильных приложений
 - **Задачи:**
 - a. Выбрать мобильное приложение для тестирования
 - b. Определить основные функции приложения, которые необходимо протестировать.
 - c. Для каждой функции разработать тестовые сценарии, включающие:
 - Описание теста
 - Предусловия

- Шаги
- Ожидаемый результат

12. Практическая работа №12: Создание тестовых данных для тестирования ИИ-модели. (Раздел 3, Тема 3.1)

- Цель: Научиться создавать наборы данных для тестирования ИИ-моделей
- **Задачи:**
 - a. Определить тип входных данных, которые принимает ИИ-модель.
 - b. Создать набор тестовых данных, включающий:
 - Типичные данные, которые модель должна обрабатывать правильно.
 - Граничные значения.
 - Некорректные или поврежденные данные.
 - Данные, которые могут вызвать предвзятость (bias).
 - c. Автоматизировать процесс создания тестовых данных, если это возможно.

13. Практическая работа №13: Оценка качества модели машинного обучения с использованием различных метрик. (Раздел 3, Тема 3.1)

- Цель: Научиться оценивать качество моделей машинного обучения
- **Задачи:**
 - a. Выбрать модель машинного обучения, которую необходимо протестировать.
 - b. Подготовить тестовый набор данных с известными правильными ответами.
 - c. Использовать модель для получения предсказаний на тестовом наборе данных.
 - d. Рассчитать следующие метрики:
 - Точность (Accuracy)
 - Полнота (Precision)
 - Отзыв (Recall)
 - F1-мера
 - ROC AUC (если применимо)
 - e. Проанализировать полученные метрики и сделать выводы о качестве модели.

14. Практическая работа №14: Тестирование модели на устойчивость к “адверсарным атакам”. (Раздел 3, Тема 3.2)

- Цель: Научиться проверять устойчивость модели к “адверсарным атакам”
- **Задачи:**
 - a. Выбрать модель машинного обучения, которую необходимо протестировать.
 - b. Изучить методы создания “адверсарных атак” (например, FGSM, PGD).
 - c. Создать “адверсарные примеры” для тестовых данных.
 - d. Подать “адверсарные примеры” на вход модели и оценить, насколько изменились результаты предсказаний.
 - e. Предложить методы защиты от “адверсарных атак”.

15. Практическая работа №15: Проведение тестирования справедливости (fairness testing) ИИ-модели. (Раздел 3, Тема 3.2)

- Цель: Научиться проверять отсутствие предвзятости в работе модели
- **Задачи:**
 - a. Определить, какие группы пользователей могут быть затронуты предвзятостью.
 - b. Собрать данные, позволяющие оценить результаты работы модели для каждой группы.

- c. Рассчитать метрики справедливости (например, demographic parity, equal opportunity).
- d. Проанализировать результаты и выявить наличие предвзятости, если она есть.
- e. Предложить методы снижения предвзятости.

Бинарная шкала (уровень освоения)	Показатели	Критерии
Зачтено	Полнота выполнения практического задания; Своевременность выполнения задания;	Студентом задание выполнено самостоятельно. При этом составлен правильный алгоритм выполнения задания.
Не зачтено	Последовательность и рациональность выполнения задания;	Студентом задание не выполнено или имеются существенные ошибки в его выполнении.

Оценивание выполнения практических заданий

Отметка "5" Практическая работа выполнена в полном объеме с соблюдением необходимой последовательности. Обучающиеся работали полностью самостоятельно: подобрали необходимые для выполнения предлагаемых работ источники знаний, показали необходимые для проведения практических и самостоятельных работ теоретические знания, практические умения и навыки. Работа оформлена аккуратно, в оптимальной для фиксации результатов форме.

Отметка "4" Практическая или самостоятельная работа выполнена студентами в полном объеме и самостоятельно. Допускается отклонение от необходимой последовательности выполнения, не влияющее на правильность конечного результата (перестановка пунктов типового плана, последовательность выполняемых заданий, ответы на вопросы). Используются указанные источники знаний. Работа показала знание основного теоретического материала и овладение умениями, необходимыми для самостоятельного выполнения работы. Допускаются неточности и небрежность в оформлении результатов работы.

Отметка "3" Практическая работа выполнена и оформлена с помощью преподавателя. На выполнение работы затрачено много времени (дана возможность доделать работу дома). Студент показал знания теоретического материала, но испытывали затруднения при самостоятельной работе с практическими материалами

Отметка "2" Выставляется в том случае, когда студент оказался не подготовленным к выполнению этой работы. Полученные результаты не позволяют сделать правильных выводов и полностью расходятся с поставленной целью. Обнаружено плохое знание теоретического материала и отсутствие необходимых умений.

3.3.3. Оценочное средство для проведения промежуточной аттестации по МДК 01.03 «Тестирование программных модулей»

Теоретические вопросы к дифференцированному зачету

1. Что такое тестирование программного обеспечения и каковы его цели?
2. Опишите основные виды тестирования (юнит, интеграционное, системное, приемочное, регрессионное, производительности, безопасности).
3. Что такое тестовый сценарий и тестовый случай? Как они связаны?
4. Опишите процесс составления баг-репорта. Какие поля он должен содержать?
5. Что такое жизненный цикл бага?
6. Какие инструменты используются для автоматизации тестирования веб-приложений?
7. Что такое CI/CD и как тестирование встраивается в этот процесс?
8. В чем особенности тестирования ИИ-модулей?
9. Какие метрики используются для оценки качества моделей ИИ?
10. Как проводится тестирование мобильных ИИ-приложений?
11. Что такое тестирование “белого ящика” и при каких условиях оно применяется?
12. Что такое тестирование “черного ящика” и при каких условиях оно применяется?
13. Расскажите про виды тестовой документации.
14. Как оценить покрытие кода тестами?
15. Что такое smoke testing?
16. Что такое sanity testing?
17. В чем разница между тестированием “ad-hoc” и исследовательским тестированием?
18. Что такое негативное тестирование?
19. Какие виды тестирования относятся к тестированию производительности?
20. Какие виды тестирования относятся к тестированию безопасности?
21. Что такое fuzz testing?
22. Как планировать тестирование ПО?
23. Как управлять тестовой средой?
24. Что такое “адверсарные атаки” применительно к ИИ?
25. Как проводить тестирование на справедливость (fairness testing) в ИИ?

Практические задания к дифференцированному зачету

1. Написать юнит-тесты для заданной функции на Python с использованием pytest.
2. Разработать интеграционные тесты для проверки взаимодействия между двумя классами Java с использованием JUnit.
3. Проанализировать покрытие кода заданного модуля юнит-тестами с использованием Coverage.py и предложить способы улучшения покрытия.
4. Разработать тестовые сценарии для функционального тестирования заданного веб-приложения.
5. Написать баг-репорт на обнаруженную ошибку в заданном ПО.
6. Настроить автоматический запуск юнит-тестов при каждом изменении кода с использованием Jenkins.
7. Провести тестирование API с использованием Postman и составить отчет о результатах.
8. Провести нагрузочное тестирование веб-приложения с использованием JMeter и проанализировать результаты.
9. Разработать план тестирования для заданного мобильного приложения.
10. Создать тестовые данные для тестирования модели классификации изображений.
11. Оценить качество заданной модели машинного обучения с использованием метрик точности, полноты и F1-меры.
12. Спроектировать набор тестов для проверки веб-сервиса, предоставляющего API для работы с данными.

13. Выбрать стратегию тестирования (например, waterfall, agile) в зависимости от особенностей проекта.
14. Составить план регрессионного тестирования после внесения изменений в код.
15. Спроектировать сценарии для тестирования пользовательского интерфейса веб-приложения.
16. Написать тесты, проверяющие обработку исключений в заданном коде.
17. Реализовать систему логирования для облегчения отладки и тестирования ПО.
18. Сравнить различные инструменты для автоматизации тестирования и выбрать наиболее подходящий для заданного проекта.
19. Провести code review для заданного кода и выявить потенциальные проблемы.
20. Описать процесс управления тестовыми данными.
21. Разработать программу для автоматической генерации тестовых данных.
22. Провести тестирование безопасности веб-приложения на наличие распространенных уязвимостей.
23. Реализовать систему мониторинга производительности приложения.
24. Составить отчет о результатах тестирования и предложить рекомендации по улучшению качества ПО.
25. Провести тестирование на “справедливость” (fairness) модели машинного обучения.

3.4. Оценочные материалы для проведения экзамена по профессиональному модулю

Оценочные средства используются с целью установления уровня и качества подготовки обучающихся ФГОС СПО по специальности **09.02.13 “Интеграция решений с применением технологий искусственного интеллекта”** в части требований к результатам освоения программы ПМ ПМ.01 **“Разработка кода для искусственного интеллекта”** и определяет:

- сформированность профессиональных компетенций, динамику формирования общих компетенций и обеспечивающих их умений.
- сформированность умения применять теоретические знания, приобретенный практический опыт при решении практических задач в условиях, приближенных к будущей профессиональной деятельности;
- соответствие усвоенных алгоритмов практической деятельности заданному эталону деятельности;
- усвоение объёма профессионально значимой информации, необходимого для формирования компетенций вида профессиональной деятельности. Задания экзамена по модулю носят компетентностно-ориентированную, комплексную готовность применять знания, умения и навыки в ситуациях, нетождественных тем, в которых они формировались. Итогом экзамена по модулю является решение: «вид профессиональной деятельности освоен /не освоен». Экзамен по модулю проводится в форме выполнения оценочных заданий.

Задания для промежуточной аттестации по ПМ 01 «Разработка кода для искусственного интеллекта»

Теоретические вопросы к экзамену квалификационному

Раздел 1: Общие вопросы по Искусственному Интеллекту и разработке

1. Что такое искусственный интеллект? Опишите основные направления и области применения ИИ.

2. Опишите этапы разработки программного обеспечения с использованием методологии Agile.
3. Что такое машинное обучение? В чем разница между обучением с учителем, без учителя и с подкреплением?
4. Какие существуют этические проблемы, связанные с развитием и применением ИИ?
5. Что такое нейронная сеть? Объясните основные принципы работы и архитектуры нейронных сетей.
6. Опишите основные методы сбора и предобработки данных для задач машинного обучения.
7. Что такое “большие данные”? Какие особенности нужно учитывать при работе с большими данными?
8. Какие языки программирования наиболее часто используются для разработки ИИ-приложений? Почему?
9. Что такое DevOps и как эта методология применяется в разработке ИИ-систем?
10. Что такое контейнеризация (Docker) и оркестрация контейнеров (Kubernetes)? Зачем они нужны при разработке и развертывании ИИ-систем?

Раздел 2: Разработка кода для ИИ (МДК 01.01)

11. Опишите основные алгоритмы машинного обучения для задач классификации (логистическая регрессия, SVM, деревья решений, kNN).
12. Опишите основные алгоритмы машинного обучения для задач регрессии (линейная регрессия, полиномиальная регрессия).
13. Как оценивать качество моделей машинного обучения? Опишите основные метрики (точность, полнота, F1-мера, ROC AUC, MSE, RMSE).
14. Что такое переобучение (overfitting) и недообучение (underfitting)? Как с ними бороться?
15. Опишите основные методы регуляризации (L1, L2). Зачем нужна регуляризация?
16. Что такое кросс-валидация? Зачем она нужна? Какие методы кросс-валидации вы знаете?
17. Что такое глубокое обучение? Чем оно отличается от традиционного машинного обучения?
18. Какие основные слои используются в сверточных нейронных сетях (CNN)? Как они работают?
19. Какие основные слои используются в рекуррентных нейронных сетях (RNN)? Как они работают?
20. Что такое “эмбединги” (embeddings) и как они используются для представления текста в задачах обработки естественного языка (NLP)?

Раздел 3: Разработка мобильных приложений с поддержкой ИИ (МДК 01.02)

21. Опишите архитектуру Android-приложения. Какие основные компоненты вы знаете?
22. Что такое жизненный цикл Activity в Android?
23. Как интегрировать TensorFlow Lite модель в Android-приложение?
24. Какие методы оптимизации TensorFlow Lite моделей для мобильных устройств вы знаете?
25. Как использовать камеру и другие сенсоры мобильного устройства в Android-приложении?
26. Какие библиотеки и инструменты используются для разработки пользовательского интерфейса в Android?
27. Как реализовать асинхронные операции в Android?
28. Как хранить данные в Android-приложении? (SharedPreferences, SQLite, Room)
29. Как использовать Firebase в Android-приложении? (Authentication, Realtime Database, Cloud Firestore)
30. Что такое “разрешения” (permissions) в Android и как с ними работать?
31. Как автоматизировать тестирование UI в Android-приложениях (Espresso, UI Automator)?

32. Как опубликовать Android-приложение в Google Play Store?

Раздел 4: Тестирование программных модулей (МДК 01.03)

33. Что такое тестирование программного обеспечения? Какие цели преследует тестирование?
34. Опишите основные виды тестирования (юнит-тестирование, интеграционное тестирование, системное тестирование, приемочное тестирование, регрессионное тестирование, тестирование производительности, тестирование безопасности).
35. Что такое “черный ящик” и “белый ящик” тестирование? В чем разница?
36. Что такое тестовый сценарий и тестовый случай?
37. Опишите процесс составления баг-репорта.
38. Что такое жизненный цикл бага?
39. Какие инструменты используются для автоматизации тестирования веб-приложений? (Selenium, Cypress)
40. Что такое CI/CD? Как тестирование встраивается в этот процесс?
41. Как оценивать покрытие кода тестами?
42. Что такое нагрузочное тестирование? Какие инструменты используются для нагрузочного тестирования?
43. В чем особенности тестирования ИИ-моделей?
44. Как оценивать качество моделей ИИ с точки зрения тестирования?
45. Что такое “адверсарные атаки” (adversarial attacks) и как тестировать модели на устойчивость к ним?
46. Что такое “тестирование справедливости” (fairness testing) в ИИ?
47. Как проводить тестирование безопасности веб-приложений?
48. Что такое статический анализ кода?
49. Что такое динамический анализ кода?
50. Какие метрики используются для оценки качества процесса тестирования?

Практические задания к экзамену квалификационному

1. Разработать алгоритм классификации изображений с использованием сверточной нейронной сети.
2. Создать чат-бота на основе модели обработки естественного языка.
3. Реализовать систему рекомендаций на основе анализа пользовательских данных.
4. Развернуть ИИ-модель в мобильном приложении с использованием TensorFlow Lite.
5. Автоматизировать тестирование программного кода с использованием PyTest.
6. Разрешить конфликты при слиянии веток в Git.
7. Провести отладку программного модуля с использованием инструментов профилирования.
8. Разработать тестовые сценарии для проверки функциональности веб-приложения с ИИ.

Задание 1: Разработка и тестирование модели классификации (МДК 01.01, 01.03)

- **Описание:** Разработайте модель машинного обучения для классификации текстов на заданную тему (например, классификация отзывов на положительные и отрицательные). Используйте библиотеки Python (scikit-learn, TensorFlow, PyTorch).
- **Требования:**
 1. Соберите набор данных для обучения и тестирования модели (минимум 1000 примеров).
 2. Выполните предобработку текста (токенизация, удаление стоп-слов, лемматизация или стемминг).
 3. Разделите данные на тренировочную и тестовую выборки.
 4. Выберите и обучите модель классификации (например, логистическая регрессия, SVM, Naive Bayes).
 5. Оцените качество модели на тестовой выборке (точность, полнота, F1-мера).

6. Напишите юнит-тесты для основных функций, используемых в процессе разработки модели.

Задание 2: Интеграция модели классификации в мобильное приложение (МДК 01.01, 01.02)

- **Описание:** Интегрируйте разработанную в Задании 1 модель классификации в Android-приложение.
- **Требования:**
 1. Преобразуйте модель в формат TensorFlow Lite.
 2. Создайте Android-приложение, которое позволяет пользователю вводить текст.
 3. Интегрируйте TensorFlow Lite модель в приложение.
 4. Приложение должно классифицировать введенный текст и отображать результат на экране.
 5. Оптимизируйте модель для работы на мобильном устройстве.
 6. Протестируйте производительность приложения на реальном устройстве.

Задание 3: Тестирование мобильного приложения с ИИ (МДК 01.02, 01.03)

- **Описание:** Проведите тестирование разработанного в Задании 2 мобильного приложения.
- **Требования:**
 1. Разработайте тестовые сценарии для проверки различных функций приложения (правильность классификации, производительность, UI).
 2. Напишите автоматизированные UI-тесты с использованием Espresso или UI Automator.
 3. Проведите тестирование производительности приложения на разных устройствах.
 4. Проверьте устойчивость приложения к различным типам входных данных.
 5. Составьте баг-репорт на обнаруженные ошибки.

Задание 4: Разработка веб-сервиса для машинного обучения и тестирование (МДК 01.01, 01.03)

- **Описание:** Разработайте простой веб-сервис (REST API) для предоставления доступа к модели машинного обучения.
- **Требования:**
 1. Используйте Python framework (Flask, FastAPI) для создания API.
 2. Реализуйте endpoint, принимающий входные данные и возвращающий результат предсказания модели.
 3. Напишите юнит-тесты для API endpoint.
 4. Проведите тестирование API с использованием Postman или аналогичного инструмента.
 5. Задokumentируйте API с использованием Swagger или OpenAPI.

Задание 5: Анализ и визуализация данных (МДК 01.01, 01.03)

- **Описание:** Выберите набор данных и проведите его анализ.
- **Требования:**
 1. Загрузите набор данных и проведите предварительную обработку.
 2. Выполните разведочный анализ данных (exploratory data analysis, EDA).
 3. Постройте графики и диаграммы для визуализации данных.
 4. Сформулируйте выводы на основе результатов анализа.
 5. Предложите возможные направления использования этих данных для построения моделей машинного обучения.

Задание 6: Тестирование на справедливость (fairness) модели машинного обучения (МДК 01.01, 01.03)

- **Описание:** Проверьте модель машинного обучения на наличие предвзятости (bias).
- **Требования:**
 1. Выберите модель машинного обучения, которая может быть предвзятой по отношению к определенным группам пользователей.

2. Соберите данные, позволяющие оценить результаты работы модели для каждой группы.
3. Рассчитайте метрики справедливости (например, demographic parity, equal opportunity).
4. Проанализируйте результаты и выявите наличие предвзятости, если она есть.
5. Предложите методы снижения предвзятости.

Задание 7: Развертывание ИИ-системы с использованием Docker и Kubernetes (МДК 01.01)

- **Описание:** Контейнеризируйте разработанную модель машинного обучения и разверните ее с использованием Kubernetes.
- **Требования:**
 1. Создайте Dockerfile для упаковки модели и всех зависимостей.
 2. Соберите Docker-образ.
 3. Создайте Kubernetes deployment и service для развертывания модели.
 4. Проверьте работоспособность развернутой системы.

Задание 8: Автоматизация тестирования с использованием CI/CD (МДК 01.03)

- **Описание:** Настройте автоматический запуск тестов при каждом изменении кода с использованием CI/CD.
- **Требования:**
 1. Выберите CI/CD систему (Jenkins, GitLab CI, GitHub Actions).
 2. Настройте автоматическую сборку проекта при каждом изменении кода в репозитории.
 3. Добавьте шаг в билд для запуска юнит-тестов и интеграционных тестов.
 4. Настройте отправку уведомлений о результатах тестирования.

Критерии и нормы оценки устных ответов

«5» за глубокое и полное овладение содержанием учебного материала, в котором обучающиеся легко ориентируются, за умение связывать теорию с практикой, высказывать и обосновывать свои суждения. Отличная отметка предполагает грамотное, логическое изложение ответа

«4» если обучающийся полно освоил материал, владеет понятийным аппаратом, ориентируется в изученном материале, грамотно излагает ответ, но содержание, форма ответа имеют отдельные недостатки

«3» если обучающийся обнаруживает знание и понимание основных положений учебного материала, но излагает его неполно, непоследовательно, допускает неточности в определении понятий, не умеет доказательно обосновывать свои суждения

«2» если обучающийся имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажающие их смысл, беспорядочно и неуверенно излагает материал

Критерии и нормы оценки выполнения практических заданий

Бинарная шкала (уровень освоения)	Показатели	Критерии
Не зачтено	Полнота выполнения практического задания; Своевременность выполнения задания; Последовательность и рациональность выполнения задания;	Студентом задание выполнено самостоятельно. При этом составлен правильный алгоритм выполнения задания. Студентом задание не выполнено или имеются существенные ошибки в его выполнении.

Оценивание выполнения практических заданий

Отметка "5" Практическая работа выполнена в полном объеме с соблюдением необходимой последовательности. Обучающиеся работали полностью самостоятельно: подобрали необходимые для выполнения предлагаемых работ источники знаний, показали необходимые для проведения практических и самостоятельных работ теоретические знания, практические умения и навыки. Работа оформлена аккуратно, в оптимальной для фиксации результатов форме.

Отметка "4" Практическая или самостоятельная работа выполнена студентами в полном объеме и самостоятельно. Допускается отклонение от необходимой последовательности выполнения, не влияющее на правильность конечного результата (перестановка пунктов типового плана, последовательность выполняемых заданий, ответы на вопросы). Используются указанные источники знаний. Работа показала знание основного теоретического материала и овладение умениями, необходимыми для самостоятельного выполнения работы. Допускаются неточности и небрежность в оформлении результатов работы

Отметка "3" Практическая работа выполнена и оформлена с помощью преподавателя. На выполнение работы затрачено много времени (дана возможность доделать работу дома). Студент показал знания теоретического материала, но испытывали затруднения при самостоятельной работе с практическими материалами.

Отметка "2" Выставляется в том случае, когда студент оказался не подготовленным к выполнению этой работы. Полученные результаты не позволяют сделать правильных выводов и полностью расходятся с поставленной целью. Обнаружено плохое знание теоретического материала и отсутствие необходимых умений.

При возникновении разногласий и выставлении итоговой оценки учитываются особые мнения членов аттестационной комиссии, и спор решается в пользу обучающегося.

ОСОБЕННОСТИ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ДЛЯ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

В ходе текущего контроля осуществляется индивидуальное общение преподавателя с обучающимся. При наличии трудностей и (или) ошибок у обучающегося преподаватель в ходе текущего контроля дублирует объяснение нового материала с учетом особенностей восприятия обучающимся содержания материала. При проведении текущего контроля и промежуточной аттестации обеспечивается соблюдение следующих требований:

- для обучающихся из числа лиц с ограниченными возможностями здоровья текущий контроль и промежуточная аттестация проводится с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся (далее - индивидуальные особенности);

- проведение мероприятий по текущему контролю и промежуточной аттестации для лиц с ограниченными возможностями здоровья в одной аудитории совместно с обучающимся, не имеющими ограниченных возможностей здоровья, допускается, если это не создает трудностей для обучающихся;

- присутствие в аудитории ассистента, оказывающего обучающимся необходимую техническую помощь с учетом их индивидуальных особенностей (занять рабочее место, понять и оформить задание, общаться с преподавателем);

- предоставление обучающимся при необходимости услуги с использованием русского жестового языка, включая обеспечение допуска на объект сурдопереводчика, тифлопереводчика (в организации должен быть такой специалист в штате (если это востребованная услуга) или договор с организациями системы социальной защиты по предоставлению таких услуг в случае необходимости);

- предоставление обучающимся права выбора последовательности выполнения задания и увеличение времени выполнения задания (по согласованию с преподавателем); по желанию обучающегося устный ответ при контроле знаний может проводиться в письменной форме или наоборот, письменный ответ заменен устным.

Образец экзаменационных билетов

Рассмотрено на заседании ПЦК _____ протокол № _____, от _____ 20____ г. Председатель ПЦК _____/_____/	Экзаменационный билет № 1 По дисциплине 1 курс	Утверждаю _____ 20____ г. Зам. директора по УР _____/Д.А. Владимиров/
--	--	--

Вопрос 1. _____

Вопрос 2. _____

Преподаватель: _____

3.5. Оценочные средства для проведения промежуточной аттестации по учебной и производственной практике.

Оценочные средства предназначены для проверки результатов освоения учебной и производственной практик программы подготовки специалистов среднего звена (далее ППССЗ) по специальности ФГОС СПО **09.02.13 «Интеграция решений с применением технологий искусственного интеллекта»**, относящейся к профессиональному модулю ПМ.0 ПМ.01 **«Разработка кода для искусственного интеллекта»**

Текущий контроль проводят с целью оценки систематичности работы обучающегося, включает в себя ряд контрольных мероприятий, реализуемых в рамках аудиторной и внеаудиторной самостоятельной работы обучающегося. Промежуточная аттестация проводится с целью установления уровня и качества подготовки, обучающихся ФГОС СПО по специальности **09.02.13 «Интеграция решений с применением технологий искусственного интеллекта»**, в части требований к результатам освоения программы учебной практики и определяет:

1. полноту и прочность знаний;
2. сформированность умений применять теоретические знания при решении практических задач.

Формой аттестации учебной и производственной практик является дифференцированный зачет, который проводится в соответствии с графиком учебного процесса и определяется согласно учебного плана, за счет времени отводимого на освоение практики. Дифференцированный зачет проводится в виде защиты отчетов. Оценочные средства составлены на основе рабочей программы учебной практики и охватывают наиболее актуальные разделы и темы рабочей программы. Перечень вопросов, выносимых на проведение дифференцированного зачета, разработаны руководителем учебной практики.

Оценочные материалы для текущего контроля и промежуточной аттестации

Текущий контроль проводится в форме устных ответов на вопросы в процессе выполнения задания по учебной и производственной практикам.

Перечень вопросов для проведения текущего контроля

МДК 01.01: “Разработка кода для искусственного интеллекта”

Раздел 1: Основы ИИ и машинного обучения

1. Что такое искусственный интеллект? Каковы основные цели и направления исследований в области ИИ?
2. Объясните разницу между машинным обучением, глубоким обучением и искусственным интеллектом.
3. Какие типы задач машинного обучения вы знаете? (классификация, регрессия, кластеризация, уменьшение размерности, и т.д.)
4. Что такое обучение с учителем, обучение без учителя и обучение с подкреплением? Приведите примеры.
5. Какие основные библиотеки Python используются для машинного обучения? (scikit-learn, TensorFlow, PyTorch)

Раздел 2: Сбор и предобработка данных

6. Какие инструменты и библиотеки используются для сбора и предобработки данных? (pandas, NumPy, BeautifulSoup, Scrapy)
7. Опишите процесс сбора данных из различных источников (CSV-файлы, базы данных, API, веб-сайты).
8. Какие методы используются для очистки данных? (удаление дубликатов, обработка пропущенных значений, удаление выбросов)
9. Что такое нормализация и стандартизация данных? Зачем они нужны?
10. Какие методы используются для кодирования категориальных признаков? (one-hot encoding, label encoding)
11. Что такое Feature Engineering и зачем он нужен?

Раздел 3: Алгоритмы машинного обучения

12. Опишите принцип работы алгоритма линейной регрессии.
13. Опишите принцип работы алгоритма логистической регрессии.
14. Как работает алгоритм k-ближайших соседей (kNN)?
15. Как строится дерево решений?
16. Опишите основные параметры алгоритма SVM (Support Vector Machine).
17. Как работает алгоритм кластеризации k-means?
18. Что такое градиентный спуск? Зачем он нужен?
19. Какие оптимизаторы используются для обучения моделей машинного обучения? (SGD, Adam, RMSprop)
20. Что такое функция потерь (loss function)? Приведите примеры.

Раздел 4: Оценка качества моделей и улучшение алгоритмов

21. Как оценивать качество моделей классификации? (точность, полнота, F1-мера, ROC AUC)
22. Как оценивать качество моделей регрессии? (MSE, RMSE, MAE)
23. Что такое переобучение (overfitting) и недообучение (underfitting)? Как с ними бороться?
24. Что такое кросс-валидация? Зачем она нужна?
25. Что такое регуляризация? Какие типы регуляризации вы знаете? (L1, L2)

Раздел 5: Глубокое обучение

26. Что такое нейронная сеть? Объясните основные компоненты (нейроны, слои, веса, функции активации).
27. Какие функции активации вы знаете? (sigmoid, ReLU, tanh)
28. Что такое обратное распространение ошибки?
29. Опишите архитектуру сверточной нейронной сети (CNN). Какие слои используются?
30. Опишите архитектуру рекуррентной нейронной сети (RNN). Какие типы RNN вы знаете? (LSTM, GRU)

МДК 01.02: “Разработка мобильных приложений с поддержкой искусственного интеллекта”

Раздел 1: Основы разработки мобильных приложений

1. Какие основные платформы для разработки мобильных приложений вы знаете? (Android, iOS, кросс-платформенные решения)
2. Какие языки программирования используются для разработки Android-приложений? (Kotlin, Java)
3. Что такое Android SDK?
4. Что такое Android Studio?
5. Какие основные компоненты Android-приложения вы знаете? (Activity, Service, Broadcast Receiver, Content Provider)
6. Что такое жизненный цикл Activity?
7. Как создать пользовательский интерфейс в Android? (XML layouts, ConstraintLayout, Material Design)
8. Как обрабатывать события в Android? (onClick, onTouch, onKeyDown)
9. Как работать с потоками в Android? (AsyncTask, Handler, Executor)
10. Как хранить данные в Android-приложении? (SharedPreferences, SQLite, Room Persistence Library)

Раздел 2: ИИ в мобильных приложениях

11. Что такое TensorFlow Lite? Зачем он нужен?
12. Как интегрировать TensorFlow Lite модель в Android-приложение?

13. Какие методы оптимизации моделей машинного обучения для мобильных устройств вы знаете? (квантование, обрезка)
14. Как использовать камеру и другие сенсоры мобильного устройства в Android-приложении?
15. Как реализовать распознавание изображений в реальном времени с использованием камеры?
16. Как использовать API для получения данных и отображения их в интерфейсе?
17. Какие библиотеки используются для работы с сетью в Android? (Retrofit, Volley)
18. Как обеспечить безопасность мобильного приложения с поддержкой ИИ?
19. Что такое этические аспекты разработки ИИ в мобильных приложениях? (конфиденциальность данных, предвзятость)
20. Как реализовать локализацию мобильного приложения?

МДК 01.03: “Тестирование программных модулей”

Раздел 1: Основы тестирования ПО

1. Что такое тестирование программного обеспечения? Каковы основные цели тестирования?
2. Какие принципы тестирования вы знаете? (например, принцип Парето, принцип исчерпывающего тестирования)
3. Какие уровни тестирования вы знаете? (юнит-тестирование, интеграционное тестирование, системное тестирование, приемочное тестирование)
4. Какие виды тестирования вы знаете? (функциональное, нефункциональное, тестирование черного ящика, тестирование белого ящика)
5. Что такое тестовый сценарий и тестовый случай?
6. Что такое позитивное и негативное тестирование?
7. Что такое тестирование производительности?
8. Что такое тестирование безопасности?
9. Что такое регрессионное тестирование?
10. Что такое автоматизированное тестирование? Какие преимущества автоматизации тестирования вы знаете?

Раздел 2: Инструменты и методы тестирования

11. Какие инструменты используются для юнит-тестирования в Python? (pytest, unittest)
12. Какие инструменты используются для юнит-тестирования в Java? (JUnit, TestNG)
13. Что такое Mock-объекты? Зачем они нужны?
14. Как оценить покрытие кода тестами? (coverage.py)

15. Какие инструменты используются для автоматизации тестирования веб-приложений? (Selenium, Cypress)
16. Какие инструменты используются для тестирования API? (Postman, Swagger)
17. Какие инструменты используются для нагрузочного тестирования? (JMeter, Gatling)
18. Что такое CI/CD? Как тестирование встраивается в этот процесс?
19. Какие системы CI/CD вы знаете? (Jenkins, GitLab CI, GitHub Actions)
20. Как создать баг-репорт? Какие поля он должен содержать?

Раздел 3: Тестирование ИИ-систем

21. В чем особенности тестирования ИИ-моделей?
22. Как оценивать качество моделей ИИ с точки зрения тестирования?
23. Что такое “адверсарные атаки” (adversarial attacks)? Как тестировать модели на устойчивость к ним?
24. Что такое “тестирование справедливости” (fairness testing) в ИИ?
25. Как проводить тестирование мобильных ИИ-приложений?

Критерии оценки:

Оценка «5» Обучающийся освоил 90-100% оцениваемой компетенции, умеет связывать теорию с практикой, применять полученный практический опыт, анализировать, делать выводы, принимать самостоятельные решения в конкретной ситуации, высказывать и обосновывать свои суждения. Демонстрирует умение вести беседы, консультировать граждан, выходить из конфликтных ситуаций. Владеет навыками работы с нормативными документами. Владеет письменной и устной коммуникацией, логическим изложением ответа.

Оценка «4» Обучающийся освоил 70-80% оцениваемой компетенции, умеет применять теоретические знания и полученный практический опыт в решении практической ситуации. Умело работает с нормативными документами. Умеет аргументировать свои выводы и принимать самостоятельные решения, но допускает отдельные неточности, как по содержанию, так и по умениям, навыкам работы с нормативно-правовой документацией.

Оценка «3» Обучающийся освоил 60-69% оцениваемой компетенции, показывает удовлетворительные знания основных вопросов программного материала, умения анализировать, делать выводы в условиях конкретной ситуационной задачи. Излагает решение проблемы недостаточно полно, непоследовательно, допускает неточности. Затрудняется доказательно обосновывать свои суждения.

Оценка «2» Обучающийся не овладел оцениваемой компетенцией, не раскрывает сущность поставленной проблемы. Не умеет применять теоретические знания в решении практической ситуации. Допускает ошибки в принимаемом решении, в работе с нормативными

в организации _____

_____ *наименование организации, юридический адрес*

Виды и качество выполнения работ

Виды и объем работ, выполненных обучающимся во время практики	Качество выполнения работ в соответствии с технологией и (или) требованиями организации, в которой проходила практика

Характеристика учебной и профессиональной деятельности обучающегося во время учебной/производственной практики

Дата «__»_____.20__Г.

Подпись руководителя практики

_____/ ФИО, должность

Подпись ответственного лица организации (базы практики)

_____/ ФИО, должность

Государственное автономное профессиональное образовательное учреждение
«Тетюшский государственный колледж гражданской защиты»

ОТЧЕТ
о прохождении
ПРОИЗВОДСТВЕННОЙ (УЧЕБНОЙ ПРАКТИКИ)

по профессиональному модулю ПМ. _____

Студента _____

Специальность _____

Группа _____

Срок прохождения практики

с _____

по _____

Руководитель практики

Оценка _____

Тетюши, 20____ г.

День 1 «___» _____ 20__ года

День 2 «___» _____ 20__ года

Государственное автономное профессиональное образовательное учреждение
«Тетюшский государственный колледж гражданской защиты»

ДНЕВНИК (ПРОИЗВОДСТВЕННОЙ) УЧЕБНОЙ ПРАКТИКИ

по профессиональному модулю ПМ.____

Студента _____

Специальность _____

Группа _____

Срок прохождения практики

с _____

по _____

Руководитель практики _____

Тетюши, 20 ____ г.

Дата (число, месяц, год)	Краткое содержание выполненной работы (по заданиям руководителя практики от колледжа и/или заданиям руководителя практики от предприятия)	Подпись руководителя практики
1	2	3
1.		
2.		
3.		
4.		
5.		

Руководитель практики

(подпись)

(расшифровка подписи)

День 1 «___» _____ 20__ года

День 2 «___» _____ 20__ года

Приложение 4

Допуск к сдаче квалификационного экзамена по профессиональному модулю

ПМ _____
Я, _____ обучающийся(аяся) _____ курса специальности СПО
_____, освоил(а) программу
профессионального модуля _____
в объеме _____ часов с _____ 20__ г. по _____ 20__ г., прошу
допустить меня к сдаче квалификационного экзамена.

дата

подпись

ОЦЕНОЧНАЯ ВЕДОМОСТЬ ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ

ПМ. _____

ФИО _____,

обучающ(ая)ийся на _____ курсе по специальности СПО _____, освоил(а) программу профессионального модуля ПМ _____

в объеме _____ час. с «__» ____ .20__ г. по «__» ____ .20__ г.

Результаты промежуточной аттестации по элементам профессионального модуля

Элементы модуля (наименование МДК, практик)	Формы промежуточной аттестации	Оценка
МДК _____	Экзамен	
МДК _____	Дифференцированный зачет	
УП. _____	Дифференцированный зачет	
ПП. _____	Дифференцированный зачет	

Коды проверяемых компетенций	Показатели оценки результата	Освоен\ не освоен	Оценка в баллах
ОК _____		Освоен/не освоен	
ОК _____		Освоен/не освоен	
ПК _____		Освоен/не освоен	
ПК _____	–	Освоен/не освоен	
ПК _____	–	Освоен/не освоен	
Итоговая оценка			

3. Перечень материалов, оборудования и информационных источников, используемых в фонде оценочных средств

-
-
-

4. Информационное обеспечение: перечень рекомендуемых учебных изданий, Интернет-ресурсов, дополнительной литературы.

Основная литература

Дополнительная литература

Интернет-ресурсы
