

Автономная некоммерческая организация профессионального образования
«ПЕРМСКИЙ ГУМАНИТАРНО-ТЕХНОЛОГИЧЕСКИЙ КОЛЛЕДЖ»
(АНО ПО «ПГТК»)

**МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ
ЭКЗАМЕНА КВАЛИФИКАЦИОННОГО
ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ
ПМ.01. РАЗРАБОТКА КОДА ДЛЯ ИСКУССТВЕННОГО
ИНТЕЛЛЕКТА**

для специальности
**09.02.13 Интеграция решений с применением технологий
искусственного интеллекта**
(код и наименование специальности)

Квалификация выпускника
Специалист по работе с искусственным интеллектом

Форма обучения
Очная

Пермь, 2026 г

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ по профессиональному модулю ПМ.01. РАЗРАБОТКА КОДА ДЛЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА» составлена в соответствии с требованиями Федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.13 Интеграция применением

ФОС предназначен для преподавателей АНО ПО «ПГТК».

Автор – составитель: Могильникова Н.С., старший преподаватель.

Оглавление

1.ПАСПОРТ ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ЭКЗАМЕНА КВАЛИФИКАЦИОННОГО ПО ПРОФЕССИОНАЛЬНОМУ МОДУЛЮ.....	Ошибка! Закладка не определена.
2.ЗАДАНИЯ ДЛЯ ЭКЗАМЕНУЮЩЕГОСЯ	8
3. КРИТЕРИИ ОЦЕНКИ ВЫПОЛНЕНИЯ ПРАКТИЧЕСКИХ ЗАДАНИЙ	10

Методические рекомендации по выполнению практических работ предназначен для оценивания достижений запланированных результатов

Методические рекомендации по выполнению практических работ представляет собой комплект материалов для проведения практических занятий (в форме практической подготовке) и осуществления контроля за выполнением работ.

Методические рекомендации по выполнению практических работ позволяет оценивать:

Код ОК, ПК	Уметь	Знать	Владеть навыками
ПК 1.1 Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.	Анализировать технические задания и выявлять требования к алгоритмам. Применять методы алгоритмизации для решения задач программирования. Разрабатывать оптимальные алгоритмы для решения задач в области ИИ.	Основные методы и подходы к построению алгоритмов (типовые поисковые алгоритмы, жадные алгоритмы, динамическое программирование, рекурсивные подходы). Принципы эффективной обработки данных. Языки программирования, применяемые для разработки алгоритмов.	Разработки, оптимизации и оценки сложности алгоритмов для ИИ-программ. Использования библиотек и инструментов для работы с алгоритмами и данными (например: Pandas, NumPy, Scikit-learn). Применения структур данных (деревья, графы, списки) для реализации алгоритмов.
ПК 1.2 Разрабатывать программные модули в соответствии с техническим заданием.	Реализовывать программные модули на основе требований технического задания. Соблюдать при разработке принципы «чистого кода». Использовать стандартные библиотеки и фреймворки для ускорения разработки.	Принципы модульного программирования. Языки программирования для разработки модулей. Стандартные фреймворки и библиотеки для работы с ИИ.	Разработки модульных ИИ-систем, соответствующих требованиям производительности и безопасности. Внедрения разработанных ИИ-модулей в комплексные программные системы. Оптимизации кода и работы с интерфейсами для взаимодействия между модулями.
ПК 1.3 Оформлять программный код в соответствии с техническим заданием.	Оформлять код в соответствии с принятыми стандартами и требованиями. Документировать разработанный программный код. Соблюдать соглашения о наименованиях переменных, функций и классов (например, PEP8 для Python).	Основные принципы чистого кода (Clean Code). Стандарты и практики документирования программного обеспечения. Инструменты для автоматической проверки качества кода (например, PyLint, ESLint).	Оформления, документирования и структурирования кода для последующей поддержки. Использования инструментов статического анализа кода для выявления ошибок и улучшения качества. Работы с системами документирования кода (например, Doxygen, Sphinx).
ПК 1.4 Использовать	Работать с системами контроля версий для	Принципы работы распределенных систем	Управления проектами с использованием

систему контроля версий программного кода с учетом обеспечения возможности организации групповой разработки.	управления проектами. Организовывать совместную работу над проектом через ветки разработки и слияние изменений. Разрешать конфликты при слиянии кода.	контроля версий. Основные команды и операции в системах контроля версий (например: commit, pull, push, merge). Методы разрешения конфликтов в ходе групповой разработки.	систем контроля версий для организации командной работы. Разрешения конфликтов при слиянии веток и использования pull request для рецензирования кода. Настройки процессов CI/CD для автоматического тестирования и развертывания кода.
ПК 1.5 Выполнять отладку программных модулей с использованием специализированных программных средств.	Использовать инструменты для отладки программного кода. Идентифицировать и исправлять ошибки в программе. Применять методы логирования для анализа выполнения программ.	Принципы работы отладчиков и логирования. Способы выявления ошибок в программе (отладка по шагам, точки останова). Инструменты для отладки кода (например, PyCharm, Visual Studio Debugger).	Отладки программных модулей с использованием пошаговой проверки. Применения методов логирования и профилирования производительности. Использования специальных средств для отладки многопоточных программ.
ПК 1.6 Выполнять тестирование программного кода.	Проводить различные виды тестирования (юнит-тестирование, интеграционное тестирование). Выполнять настройки окружения и подготовку тестовых данных. Фиксировать результаты выполнения тестов и подготавливать отчеты о результатах тестов. Определять уровень критичности дефектов. Разрабатывать автоматизированные тесты для тестирования модулей и/или отдельных функций. Восстанавливать окружение и тесты после сбоя	Техники выполнения тестовых прогонов. Инструменты и среды выполнения тестирования. Языки разработки автоматизированных тестов. Инструменты для тестирования программного кода. Правила выполнения отчетов о тестировании	Выполнения статического тестирования программного кода на предмет выявления ошибок/дефектов алгоритмов, в том числе – на наличие обработки исключений. Выполнения тестирования программных модулей в соответствии с тест-планом. Генерирования тестовых данных. Выполнения интеграционного тестирования в соответствии с заданием. Выполнения регрессионного тестирования в соответствии с заданием. Работы с CI/CD пайплайнами для автоматизации

			тестирования.
ПК 1.7 Составлять тестовые сценарии.	Проектировать тестовые сценарии на основе тестовых планов. Разрабатывать тестовые пакеты и задания на выполнение тестирования. Использовать шаблоны для написания тест-кейсов. Оценивать риски при отборе тестов для регрессионного тестирования. Оценивать тесты на соответствие целям тестирования.	Цели, задачи и виды тестирования. Понятие стратегии тестирования. Жизненный цикл дефекта. Основы тест-дизайна: тестовый сценарий, тестовый пакет, чек-лист, основные шаблоны. Основные инструменты проектирования тестов. Методы и подходы к написанию тестов (Test-Driven Development, Behavior-Driven Development).	Разработки тестовых сценариев в соответствии с тестовым планом (тестирование производительности, надежности, UI-тестирование), в том числе с применением средств автоматизации проектирования. Разработки тестовых пакетов и заданий на выполнение тестирования. Оценки тестовых данных на предмет покрытия строк и покрытия ветвей, выполнения валидации данных. Автоматизации создания и выполнения тестовых сценариев.
ОК 01 Выбирать способы решения задач профессиональной деятельности применительно к различным контекстам	распознавать задачу и/или проблему в профессиональном и/или социальном контексте, анализировать и выделять её составные части определять этапы решения задачи, составлять план действия, реализовывать составленный план, определять необходимые ресурсы выявлять и эффективно искать информацию, необходимую для решения задачи и/или проблемы владеть актуальными методами работы в профессиональной и смежных сферах оценивать результат и последствия своих действий (самостоятельно или с помощью наставника)	актуальный профессиональный и социальный контекст, в котором приходится работать и жить структура плана для решения задач, алгоритмы выполнения работ в профессиональной и смежных областях основные источники информации и ресурсы для решения задач и/или проблем в профессиональном и/или социальном контексте методы работы в профессиональной и смежных сферах порядок оценки результатов решения задач профессиональной деятельности	
ОК 02 Использовать современные средства поиска, анализа и интерпретации	определять задачи для поиска информации, планировать процесс поиска, выбирать необходимые источники информации	номенклатура информационных источников, применяемых в профессиональной деятельности приемы структурирования	

информации и информационные технологии для выполнения задач профессиональной деятельности	выделять наиболее значимое в перечне информации, структурировать получаемую информацию, оформлять результаты поиска оценивать практическую значимость результатов поиска применять средства информационных технологий для решения профессиональных задач использовать современное программное обеспечение в профессиональной деятельности использовать различные цифровые средства для решения профессиональных задач	информации формат оформления результатов поиска информации современные средства и устройства информатизации, порядок их применения программное обеспечение в профессиональной деятельности, в том числе цифровые средства психологические основы деятельности коллектива	
ОК 05 Осуществлять устную и письменную коммуникацию на государственном языке Российской Федерации с учетом особенностей социального и культурного контекста	грамотно излагать свои мысли и оформлять документы по профессиональной тематике на государственном языке проявлять толерантность в рабочем коллективе	правила оформления документов правила построения устных сообщений особенности социального и культурного контекста	

ЗАДАНИЯ ДЛЯ ЭКЗАМЕНУЮЩЕГОСЯ

Задания для экзамена квалификационного ориентированы на проверку освоения вида деятельности (всего модуля) *ПМ.01. РАЗРАБОТКА КОДА ДЛЯ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА*» в целом. Типовые задания носят компетентностно-ориентированный, комплексный характер.

Типовое задание для экзаменуемого

ИНСТРУКЦИЯ

1. Внимательно прочитайте задание.
2. Результаты работы необходимо загрузить на СДО ПГТК в курс Экзамен (квалификационный) по ПМ.01
3. Время выполнения задания 70 минут.

ТЕКСТ ЗАДАНИЯ

Теоретические вопросы:

Необходимо ответить письменно на два вопроса:

1. Опишите общую последовательность этапов обучения нейронной сети.
2. Что такое регуляризация и зачем она применяется при обучении моделей ИИ?
3. Какие виды активационных функций используются в искусственных нейронных сетях и каковы их особенности?
4. Объясните смысл термина "перенасыщенность модели" ("overfitting") и предложите способы его устранения.
5. Какая роль отводится предварительному процессу подготовки данных при обучении моделей ИИ?

Практическое задание:

Необходимо разработать программу на Python для обучения нейронной сети, решающей задачу бинарной классификации на синтетическом наборе данных. Исходные данные представляют собой двухклассовый набор точек на двумерной плоскости (x , y), классы разделяются нелинейно (параболой).

Требуется выполнить следующее:

1. Загрузить и визуализировать тренировочный набор данных.
2. Разбить данные на обучающую и тестовую выборки.
3. Подобрать подходящую архитектуру нейронной сети (количество слоёв, число нейронов, активационные функции).
4. Произвести обучение сети с использованием выбранного оптимизатора и функции потерь.
5. Оценить качество полученной модели на тестовой выборке.
6. Вывести итоговую точность модели.

Требования к решению:

- Код должен быть аккуратно написанным, читаемым и содержать адекватные комментарии.
- Используемые пакеты Python: TensorFlow/Keras, NumPy, Matplotlib/Pandas/SciKit-Learn (или аналогичные библиотеки).
- Необходимо продемонстрировать понимание основ обучения нейронных сетей и умения оценивать результаты работы моделей.

Образец оформления отчета:

Отчет оформляется в электронном виде (.docx/.pdf). Должен включать:

- Постановку задачи.
- Код программы с комментариями.
- Результаты работы программы с выводами.
- Визуализацию обучающих данных и результатов классификации.

Устная защита:

Во время устной защиты кандидат обязан представить свою работу комиссии, объяснить принятые решения и ответить на дополнительные вопросы членов комиссии.

3. КРИТЕРИИ ОЦЕНКИ ВЫПОЛНЕНИЯ ПРАКТИЧЕСКИХ ЗАДАНИЙ

Результаты сформированности общих и профессиональных компетенций выражаются в уровнях: высоком, среднем, низком.

Высокий уровень – студент уверенно демонстрирует готовность и способность к самостоятельной профессиональной деятельности не только в стандартных, но и во внештатных ситуациях.

Средний уровень – студент выполняет все виды профессиональной деятельности в стандартных ситуациях уверенно, добросовестно, эффективно.

Низкий уровень – студент выполняет все виды профессиональной деятельности, допуская ошибки и неточности.

При оценке выполненных практических заданий учитываются:

- рациональное распределение времени на выполнение задания;
- наличие развернутого ответа на поставленные вопросы;
- самостоятельность выполнения задания;
- своевременность выполнения заданий в соответствии с установленным лимитом времени.

Критерии оценки практического задания:

1. Оформление отчета (максимум 10 баллов):

- Четко изложенная постановка задачи (2 балла);
- Аккуратное оформление отчета (структура, заголовки, подписи к таблицам и рисункам) (3 балла);
- Адекватное наличие комментариев в коде (5 баллов).

2. Правильное выполнение задачи (максимум 30 баллов):

- Грамотная загрузка и предварительный анализ данных (5 баллов);
- Верная визуализация тренировочной выборки (5 баллов);
- Правильно выполненное разделение данных на обучающую и тестовую выборки (5 баллов);
- Подбор подходящей архитектуры нейронной сети (слои, функции активации, начальные веса) (10 баллов);
- Грамматически правильное использование оптимизатора и функции потерь (5 баллов).

3. Качество обученной модели (максимум 20 баллов):

- Высокая точность классификации на тестовой выборке (лучше 90%) (10 баллов);
- Отсутствие эффекта переобучения (good generalization) (5 баллов);
- Оценка результата с указанием потери и точности (5 баллов).

4. Уровень владения инструментом (Python/Tensorflow/Keras) (максимум 20 баллов):

- Исправность и аккуратность написанного кода (5 баллов);
- Логичность структуры программы (модульность, отсутствие избыточности) (5 баллов);
- Владение необходимыми функциями и методами библиотек (10 баллов).

5. Уровень самостоятельности и творчества (максимум 10 баллов):

- Наличие оригинальных идей и улучшений относительно стандартных процедур (например, собственные попытки улучшить качество модели или подбор лучших гиперпараметров) (5 баллов);
- Демонстрация уверенного владения материалом, стремление глубже разобраться в поставленных задачах (5 баллов).

6. Устная защита (максимум 10 баллов):

- Последовательно изложено обоснование принятых решений (5 баллов);
- Способность ясно и точно отвечать на вопросы комиссии (5 баллов).

Итоговая оценка:

Суммарный балл складывается из всех перечисленных пунктов и выставляется в диапазоне от 0 до 100 баллов.

- До 60 баллов включительно — неудовлетворительно;
- От 61 до 75 баллов — удовлетворительно;
- От 76 до 90 баллов — хорошо;
- От 91 до 100 баллов — отлично.

Эти критерии обеспечивают объективную оценку знаний и навыков студентов, позволяя определить степень готовности студента к профессиональной деятельности в области разработки кода для обучения искусственного интеллекта.