МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ)

Программа итоговой (государственной итоговой) аттестации, как компонент образовательной программы
[q1row62]
по направлению подготовки
09.03.01 Информатика и вычислительная техника,
утвержденной первым проректором ФГАОУ ВО РУТ (МИИТ) Тимониным В.С.

ПРОГРАММА ИТОГОВОЙ (ГОСУДАРСТВЕННОЙ ИТОГОВОЙ) АТТЕСТАЦИИ

ПОДГОТОВКА К СДАЧЕ И СДАЧА ГОСУДАРСТВЕННОГО ЭКЗАМЕНА

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность (профиль): IT-сервисы и технологии обработки данных на транспорте

Квалификация выпускника: Бакалавр

Форма обучения: Очная

ФГАОУ ВО РУТ (МИИТ)
ID подписи: 170737
Подписал: директор Паринов Денис Владимирович
Дата: 22.10.2026

Программа итоговой (государственной итоговой) аттестации в виде электронного документа выгружена из единой корпоративной информационной системы управления университетом и соответствует оригиналу

1. Итоговая (государственная итоговая) аттестация по направлению подготовки 09.03.01 Информатика и вычислительная техника и направленности (профилю) IT-сервисы и технологии обработки данных на транспорте в соответствии с учебным планом проводится в форме: Государственного экзамена / итогового экзамена (далее — экзамен).

2. Экзамен.

2.1. Программа экзамена.

Программа государственного экзамена включает в себя задания по дисциплинам, освоение которых имеет определяющее значение для профессиональной компетенции выпускников. Программа экзамена:

1. Основы web-программирования
1.1 HTML. Табличная верста
1.2 HTML. Создание форм
1.3 CSS. Форматирование теста
1.4 CSS. Блочная верстка
2. Системы программирования Python
2.1 Классы в Python
2.2 Основные библиотеки для анализа данных, настройка IDE
2.3 Подключение к базе данных
2.4 Программный и математический аппарат для анализа данных
3. Машинное обучение и анализ данных
3.1 EDA\Feature engineering
3.2 Кластеризация
3.3 Классификация
3.4 Регрессия
3.5 Интерпретация моделей машинного обучения
4. Анализ больших текстовых данных и текстовый поиск
4.1 Seq2Seq модели с вниманием
4.2 Трансформеры
4.3 LLM
5. Алгоритмы и структуры данных
5.1 Алгоритмы
5.2 Структуры данных
5.3 Программный и математический аппарат для анализа данных
6.Основы SQL
6.1 Проектирование БД
6.2 Создание структур в базе данных и написание запросов

6.3 Импорт данных из внешних источников в БД MS SQL

6.4 Основы работы с конструктором таблиц и команд языка SQL

Государственный экзамен проводится в сочетании устной и письменной форм в формате собеседований.

Государственный экзамен проводится в два этапа:

Этап 1. Тестирования и письменное решение задачи.

Этап 2. Собеседование с предоставлением портфолио.

Профессиональная область (дисциплины) для практических заданий определяется по итогам тестирования. Профильным считается область (дисциплина), по которой студент имеет наибольший процент правильных ответов по тесту. В случае равномерного распределения процента правильных ответов студент выбирает профиль на свое усмотрение.

Порядок проведение экзамена:

- начало этапа 1. выдача студенту тестового задания;
- проверка теста и выдача студенту практических заданий по профессиональной области (дисциплине);
- выполнение студентом практическго задания с использование персональных компьютеров;
- время выполнения заданий этапа 1 – не более 1 часа;
- выполненное практическое задание сдается комиссии в виде: скриптов (программных сценариев); кода на бумажном носителе.
- перенос программного сценария на бумажный носитель и передача его комиссии. Окончание этапа 2;
- переход к этапу 2.
- защита реализованного программного продукта и ответы на теоретические вопросы (не более 10 минут на студента).
- комиссией проверяются портфолио студентов и выставляется оценка. Время проверки всех практических заданий экзаменуемых – не более 1 часа, а студенты находятся вне аудитории до окончания проверки;
- оглашение итогового решения комиссии студентам в день проведения экзамена.

Во время экзамена разрешается пользоваться:

- подготовленным файлом requirements.txt;
- подготовленными самописными модулями;
- клонировать репозитории github;
- инструментами импорта и установки библиотек из интернета.

Во время экзамена не разрешается:

- импортировать библиотеки полностью выполняющие задания;
- использовать LLM для решения задачи.

2.2. Перечень вопросов, выносимых на экзамен.

Перечень вопросов и заданий для ГЭК. Перечень вопросов/задач является примерным.

I. Машинное обучение и анализ данных

Тестовые задания (21 вопрос)

1. Какие задачи решают деревья решений?

а) Только классификацию.

б) Только регрессию.

в) Классификацию и регрессию.

г) Только анализ временных рядов.

2. Что такое дерево решений в контексте машинного обучения?

а) Алгоритм для классификации и регрессии, который представляет собой структуру в виде дерева, где каждый узел представляет атрибут, а листья — возможные исходы.

б) Метод анализа данных, основанный на машинном обучении, который используется для прогнозирования будущих событий.

в) Инструмент визуализации данных, позволяющий наглядно представить структуру и связи между различными элементами.

г) Статистический метод для анализа влияния различных факторов на результат.

3. Какой тип визуализации данных используется для демонстрации корреляции между двумя переменными?

а) Диаграмма рассеяния.

б) Гистограмма.

в) Линейный график.

г) Круговая диаграмма.

4. Какой тип визуализации данных лучше всего подходит для отображения распределения числовых данных?

а) Столбчатая диаграмма.

б) Гистограмма.

в) Линейный график.

г) Круговая диаграмма.

5. Какой метод анализа данных применяется для разбиения данных на группы на основе сходства?

а) Регрессионный анализ.

б) Кластеризация.

в) Дисперсионный анализ.

г) Корреляционный анализ.

6. Какой метод анализа данных используется для определения взаимосвязи между двумя переменными?

а) Регрессионный анализ.

б) Кластеризация.

в) Дисперсионный анализ.

г) Корреляционный анализ.

7. Какое ключевое слово используется в Python для создания конструктора?

а) init.

б) class.

в) main.

г) name.

8. Какой метод в Python отвечает за удаление элемента списка с указанной позицией?

а) pop().

б) remove().

в) delete().

г) clear().

9. Какие типы JOIN существуют в SQL?

а) INNER JOIN, LEFT JOIN, RIGHT JOIN.

б) CROSS JOIN, FULL JOIN, SEMI JOIN.

в) OUTER JOIN, INNER JOIN, CROSS JOIN.

г) Все вышеперечисленные типы JOIN.

10. Что делает команда SELECT в SQL?

а) Создаёт новую таблицу.

б) Удаляет данные из таблицы.

в) Выбирает данные из таблицы.

г) Изменяет структуру таблицы.

11. Какой из видов машинного обучения основывается на взаимодействии обучаемой системы со средой?

а) Обучение с учителем.

б) Глубинное обучение.

в) Обучение без учителя.

г) Обучение с подкреплением.

12. Нейросети хорошо проявляют себя не только в распознавании, но и в генерации изображений. Но кое с чем у них все-таки возникают проблемы. С чем именно?

а) Цвет.

б) Форма.

в) Глубина, количество пикселей.

г) Текстуры.

13. У машинного обучения есть ряд задач. Как называется та, что направлена на предсказание значения той или иной непрерывной числовой величины для входных данных?

а) Переобучение.

б) Классификация.

в) Регрессия.

г) Кластеризация.

14. Искусственные нейронные сети (ИНС) — модели машинного обучения, использующие комбинации распределенных простых операций, зависящих от обучаемых параметров, для обработки входных данных. Какого вида ИНС не существует?

а) Импульсные.

б) Состязательные.

в) Наивные.

г) Рекуррентные.

15. На чем основана модель случайного леса?

а) На ансамбле деревьев.

б) На ансамбле графов.

в) На паре деревьев.

г) На матрице смежности для ансамбля графов.

16. Что является результатом прогноза логистической регрессии?

а) Численное значение целевой переменной.

б) Вероятность возникновения события.

в) Погрешность возникновения некоторого события.

г) Коэффициенты линейной функции, описывающей закон возникновения события.

17. Что из перечисленного относится к технологиям ИИ?

а) CFD-расчёты.

б) СУБД.

в) 3D-печать.

г) Обработка естественного языка.

18. Что из перечисленного является open-source компонентом?

а) SAS.

б) Teradata.

в) Matplotlib.

г) Power BI.

19. Как расшифровывается ELT?

а) Extract-List-Transform.

б) Extract-Load-Translate.

в) Extract-Load-Transform.

г) Excel-Load-Trim.

20. Какой из перечисленных признаков НЕ является интервальным?

а) Рост.

б) Давление.

в) Марка автомобиля.

г) Вес.

21. Что из перечисленного НЕ относится к процедурам улучшения качества данных?

а) "Золотая запись".

б) Идентификационный анализ.

в) Бининг.

г) Извлечение.

Практические задания (5 заданий)

1. EDA и Feature Engineering

Считайте CSV-файл. Проведите разведочный анализ данных. Результаты представьте в виде фрейма данных и текстового описания. Создайте графическое сопровождение (не менее 3 графиков).

2. Кластеризация

Постройте кластеризацию методами k-means (метод elbow для выбора количества кластеров) и DBSCAN. Постройте графики в параллельных координатах и визуализируйте результаты для 2 и 3 переменных.

3. Anomaly Detection

Прочитайте текстовый файл. Проведите анализ аномалий. Обоснуйте выбор инструмента, проведите анализ и разделите данные на выборки, если необходимо.

4. Классификация

Постройте базовую модель классификации с использованием AutoML. Выберите метрику качества (AUC ROC, Accuracy, F1-score). Постройте Confusion Matrix и ROC-Curve. Сравните с моделями на основе деревьев, леса, бустинга и логистической регрессии.

5. Регрессия

Постройте базовую модель регрессии с использованием AutoML. Выберите метрику качества (MAE, MSE, RMSE). Постройте график реальных и предсказанных значений. Сравните с моделями на основе деревьев, леса, бустинга и линейной регрессии.

II. Алгоритмы и структуры данных

Тестовые задания

1. Про сложность и графы: Для ориентированного ациклического графа (DAG) с V вершинами и E рёбрами, какой алгоритм нахождения кратчайших путей из одной вершины во все остальные будет наиболее эффективным и почему? Оцените его сложность.

а) Алгоритм Дейкстры за O(E log V), так как он работает для любых неотрицательных весов.

б) Поиск в ширину (BFS) за O(V+E), так как граф невзвешенный.

в) Топологическая сортировка с последующим релаксацией рёбер за O(V+E), так как позволяет обработать вершины в оптимальном порядке.

г) Алгоритм Форда-Беллмана за O(VE), так как он самый универсальный.

2. Про структуры данных: Вы проектируете систему кэширования для высоконагруженного API. Требуется реализовать политику вытеснения Least Recently Used (LRU) с операциями get и put за константное время O(1) в среднем. Какую комбинацию структур данных вы выберете?

а) Очередь (queue) и хэш-таблица (hash map).

б) Связный список (linked list) и хэш-таблица (hash map).

в) Двусвязный список (doubly linked list) и словарь, отображающий ключ на узел списка.

г) Приоритетная очередь (heap) и хэш-таблица.

3. Про рекурсию и динамику: Решите рекуррентное соотношение, описывающее время работы наивного рекурсивного алгоритма для вычисления чисел Фибоначчи: T(n) = T(n-1) + T(n-2) + O(1). Какова его асимптотическая сложность?

а) O(n)

б) O(2^n)

в) O(?^n), где ? – золотое сечение (~1.618)

г) O(n log n)

Устные / Аналитические задания

1. Анализ алгоритма: Дан алгоритм, якобы находящий второе по величине число в массиве за один проход. Найдите логическую ошибку в его рассуждении или псевдокоде и предложите корректное решение.

2. Сравнение и выбор: Когда для решения задачи поиска будет предпочтительнее использовать красно-черное дерево, а когда – хэш-таблицу? Сравните с точки зрения временной сложности основных операций, гарантий производительности, использования памяти и функциональности (например, необходимость обхода в отсортированном порядке).

Письменные задания

1. Реализация с оптимизацией: Реализуйте итеративную (не рекурсивную) версию алгоритма быстрой сортировки (quicksort), использующую стек для эмуляции рекурсии. Ваша реализация должна быть устойчивой к уже отсортированным данным и использовать рандомизированный выбор опорного элемента (pivot). Проанализируйте сложность по памяти.

2. Задача на проектирование: Реализуйте класс Trie (префиксное дерево) для эффективного хранения и поиска строк. Поддержите операции insert(word), search(word) (точное совпадение) и startsWith(prefix). Оптимизируйте использование памяти.

3. Алгоритмическая задача: Напишите функцию, которая находит в бинарном дереве наибольший бинарный поисковый поддерево (Largest BST Subtree). Функция должна возвращать размер (количество узлов) этого поддерева. Алгоритм должен работать за O(n) времени.

III. Основы SQL

Тестовые задания

1. Сложный запрос: В таблице employee (id, name, salary, department_id) нужно найти сотрудников, чья зарплата выше средней по их отделу, но не является максимальной в этом отделе. Выберите корректный вариант запроса, использующий оконные функции.

а) SELECT * FROM employee WHERE salary > AVG(salary) OVER (PARTITION BY department_id) AND salary < MAX(salary) OVER (PARTITION BY department_id);

б) WITH dept_stats AS (...) ... (предложить варианты с CTE)

в) Запрос невозможно выполнить одним SELECT без вложенных подзапросов.

г) Ваш вариант, демонстрирующий понимание ограничений оконных функций в WHERE.

2. Производительность и индексы: У вас есть запрос с JOIN трех больших таблиц по полям с внешними ключами и фильтрацией по дате в одной из них. Запрос выполняется медленно. Какой набор действий для исследования и оптимизации будет наиболее полным?

а) Добавить индексы на все поля, участвующие в JOIN и WHERE.

б) Проанализировать план выполнения (EXPLAIN ANALYZE), проверить наличие и использование индексов, рассмотреть денормализацию или материализованные представления.

в) Увеличить размер оперативной памяти СУБД.

г) Переписать запрос, используя подзапросы вместо JOIN.

Устные / Аналитические задания

1. Объяснение различий: Детально объясните разницу между INNER JOIN, LEFT JOIN и FULL OUTER JOIN не только на примере результата, но и с точки зрения этапов выполнения запроса процессором СУБД. В каком случае LEFT JOIN может дать такой же результат, как INNER JOIN?

2. Проблема уникальности: В чем фундаментальная проблема использования SELECT DISTINCT для "исправления" дубликатов, возникающих из-за некорректного JOIN? Как правильно устранить причину дубликатов?

Письменные задания

1. Аналитический запрос: Напишите SQL-запрос для таблицы user_sessions (session_id, user_id, start_time, end_time), который вычисляет максимальное количество одновременных активных сессий за весь период. Решите без использования самосоединения (self-join) для всех записей, если возможно.

2. Операции с данными: Напишите транзакционный SQL-скрипт, который безопасно обновляет статус заказов в таблице orders на "выполнен" для всех заказов старше 30 дней, но только если по ним нет открытых споров в таблице disputes. Скрипт должен логировать изменения и обеспечивать целостность данных.

3. Рекурсивный запрос: Используя рекурсивное общее табличное выражение (RECURSIVE CTE), напишите запрос к таблице folder_structure (folder_id, parent_folder_id, name), который выводит полный иерархический путь для каждой папки в формате "Корень/Документы/Проекты/Отчеты".

IV. Анализ больших текстовых данных и текстовый поиск

Тестовые задания (6 вопросов)

1. Какое правильное определение модели Sequential?

а) Модель разбиения текста на отдельные токены/блоки.

б) Полносвязная модель нейронной сети прямого распространения.

в) Рекуррентная нейронная сеть.

2. Что на вход получает encoder модели text2text?

а) Векторное представление вопроса.

б) Векторное представление ответа.

в) Вектор контекста.

г) Векторное представление ответа и контекста.

3. Какие функции активации обычно есть в слое LSTM?

а) Сигмоид.

б) Tanh.

в) ReLU.

г) Units.

4. Какой результат соответствует преобразованию текста методом texts_to_matrix?

а) [0, 0, 0, 1, 0, 0, 1, 0, 0, 1].

б) [416, 10, 3].

в) Словарь частот.

5. Какая задача относится к задачам ML без учителя?

а) NER.

б) Topic Model.

в) Text2Text.

г) Суммаризация.

6. Функция активации ReLU — это...

а) Линейный выпрямитель.

б) Моделирование векторного пространства.

в) Функция, отключающая нейроны по схеме Бернулли.

Устные задания (3 вопроса)

1. Что такое большая лингвистическая модель?

2. Каковы компоненты архитектуры LLM?

3. Какие типы LLM вы знаете?

Письменные задания (5 заданий)

1. Разработать API-сервис, принимающий вопросы пользователей и выполняющий обращения к языковым моделям с применением подхода RAG.

2. Прочитайте текстовый файл, проведите анализ тональности текста и анализ именных сущностей.

3. Прочитайте текстовый файл, напишите программу, прогнозирующую количество реакций на публикацию текста в социальной сети.

4. Прочитайте текстовый файл. Проведите векторизацию текста, используя различные методы.

5. Прочитайте текстовый файл. Используйте его как контекст для создания диалоговой системы.

V. Основы веб-программирования

Тестовые задания

1. Архитектура и безопасность: Какой механизм является наиболее эффективным для предотвращения межсайтового скриптинга (XSS) в современном веб-приложении, рендерящем пользовательский контент?

а) Валидация всех входящих данных на сервере с помощью регулярных выражений.

б) Экранирование (escaping) HTML-сущностей на стороне сервера перед выводом.

в) Использование Content Security Policy (CSP) заголовков для ограничения источников исполняемых скриптов.

г) Проверка наличия тегов <script> в пользовательском вводе.

2. Протоколы и производительность: В чем ключевое архитектурное отличие HTTP/2 от HTTP/1.1, позволяющее значительно увеличить производительность загрузки страниц со множеством ресурсов?

а) Бинарный фреймовый протокол вместо текстового.

б) Обязательное использование TLS/SSL шифрования.

в) Мультиплексирование нескольких запросов в рамках одного TCP-соединения.

г) Серверные push-уведомления.

Устные / Аналитические задания

1. Жизненный цикл запроса: Детально опишите полный жизненный цикл асинхронного AJAX/Fetch запроса из браузера на сервер и обратно, начиная с вызова fetch() и заканчивая срабатыванием then(). Включите в описание: Event Loop, макрозадачи/микрозадачи, рендеринг, взаимодействие с сетевым стеком.

2. Проблема состояния (state): В чем заключаются фундаментальные проблемы управления состоянием (state management) в большом одностраничном приложении (SPA)? Сравните подходы: подъем состояния (lifting state up), Context API (React), специализированные хранилища (Redux, MobX).

Письменные задания

1. Реализация механизма: Напишите на чистом JavaScript (без фреймворков) реализацию простого роутера (router) для SPA. Он должен: обрабатывать изменения URL в адресной строке (через History API), сопоставлять пути с "компонентами" (функциями рендеринга), отображать соответствующий контент и поддерживать параметры в URL.

2. Работа с API и асинхронностью: Напишите функцию-обертку для выполнения HTTP-запросов с поддержкой: повторных попыток (retry logic) при временных ошибках (5xx), таймаута, отмены запроса через AbortController, базовой обработки ошибок (сетевых, HTTP-статусов) и логирования.

3. Оптимизация производительности: Дан список из 10 000 элементов, которые необходимо отобразить в виде прокручиваемой таблицы с возможностью фильтрации и сортировки. Опишите архитектурные и технические подходы для обеспечения плавного пользовательского интерфейса (например, виртуализация, дебаунсинг, мемоизация).

VI. Программирование и алгоритмы

Тестовые задания

1. Рекурсия и стек: При глубокой рекурсии в Python может возникнуть RecursionError. Объясните, как механизм sys.setrecursionlimit() взаимодействует с ограничениями стека вызовов операционной системы. Почему простое увеличение лимита — опасное решение, и какие архитектурные альтернативы (итерация, явный стек, хвостовая рекурсия) следует рассматривать?

2. Итераторы и генераторы: В чем принципиальное различие между __iter__()/__next__() и ключевым словом yield с точки зрения управления состоянием и потребления памяти при обработке потоков данных бесконечной или очень большой длины?

3. Асимптотический анализ: Дан алгоритм со сложностью O(n log n). При тестировании на данных размером 10^6 элементов время работы составило 2 секунды. Оцените время работы на данных размером 10^8 элементов, предположив, что константа пропорциональности не изменится. Как изменится оценка, если выяснится, что алгоритм имеет скрытые накладные расходы по памяти, приводящие к своппингу на больших объемах?

Практическое задание: Реализуйте LRUCache с использованием collections.OrderedDict. Затем перепишите его, используя двусвязный список и словарь, и проведите сравнительный анализ производительности двух реализаций при интенсивных операциях get и put.

VII. Операционные системы и Linux

Тестовые задания

1. Виртуальная память: Объясните, как работает механизм своппинга (swapping) и в какой ситуации его активное использование для процесса обработки данных в Pandas укажет на критическую проблему в алгоритме, а не просто на нехватку RAM.

2. Файловые системы: Почему операция fsync() критически важна для баз данных? Что может произойти с данными в PostgreSQL при аварийном отключении сервера, если транзакция была зафиксирована (COMMIT), но соответствующие fsync() вызовы не завершились?

3. Процессы и потоки: В Python из-за GIL (Global Interpreter Lock) многопоточность не даёт выигрыша на CPU-bound задачах. Опишите стратегию распараллеливания тяжелых вычислений (например, перемножение матриц) с использованием модулей multiprocessing или concurrent.futures. Какие накладные расходы при этом возникают?

Практическое задание: Напишите скрипт, который мониторит потребление памяти заданным процессом (по PID) и при превышении

порогового значения снимает и анализирует его дамп памяти (core dump) с помощью gcore, после чего безопасно завершает процесс.

VIII. Сети и сетевые протоколы

Тестовые задания

1. TCP vs UDP: Вы разрабатываете систему для передачи потоковых данных с датчиков (IoT). В каких сценариях выбор UDP с собственной логикой контроля перегрузок и повторной отправки будет оправдан, несмотря на его ненадёжность по сравнению с TCP?

2. TLS/SSL: Опишите полную последовательность шагов в рукопожатии TLS 1.3 (TLS handshake). Какие криптографические алгоритмы (шифры, алгоритмы обмена ключами) сегодня считаются безопасными, а какие (например, RSA-1024, SHA-1) — устаревшими и почему?

3. HTTP/2 и HTTP/3: В чем фундаментальное архитектурное отличие HTTP/3 от HTTP/2? Как замена транспортного протокола с TCP на QUIC решает проблему "head-of-line blocking"?

Практическое задание: Используя scapy или socket, напишите утилиту, которая определяет, проходит ли трафик до определённого хоста через NAT (преобразование сетевых адресов), и оценивает количество хостов за этим NAT.

IX. Архитектура и проектирование ПО (

Тестовые задания

1. Монолит vs Микросервисы: Опишите "закон Конвея" (Conway's Law) и его влияние на выбор архитектуры. Приведите пример, когда попытка разбить монолит на микросервисы без изменения организационной структуры команды приведет к провалу.

2. CAP-теорема: Для распределённой системы хранения конфигураций (например, etcd, Consul) объясните, какой компромисс CAP (Consistency, Availability, Partition Tolerance) она выбирает по умолчанию и как это влияет на её поведение при сетевом разделении (network partition).

3. Шаблоны: Объясните разницу между паттернами "Стратегия" (Strategy), "Состояние" (State) и "Шаблонный метод" (Template Method). Приведите конкретный пример из области Data Engineering, где каждый из них был бы применим.

Практическое задание: Спроектируйте схему взаимодействия сервисов для системы онлайн-платежей с гарантированной идемпотентностью операций, учитывая возможные отказы сети и дублирование запросов.

X. Контейнеризация и оркестрация

Тестовые задания

1. Docker: Что такое "multi-stage build" в Docker и какую проблему безопасности и оптимизации размера образа он решает? Приведите пример для образа, содержащего Python-приложение с native-зависимостями.

2. Kubernetes: Опишите, как работают Liveness, Readiness и Startup пробы в Kubernetes. В каком сценарии pod может быть "живым" (live), но не "готовым" (ready), и к каким последствиям это приведет, если он включен в Service?

3. Безопасность: Что такое "security context" в Kubernetes и как с его помощью ограничить возможности контейнера (например, запретить запуск от root, сделать файловую систему read-only, отключить kernel capabilities)?

Практическое задание: Напишите манифест Deployment для stateless ML-сервиса на Python с автоматическим горизонтальным масштабированием (HPA) по CPU, использованием ConfigMap для настроек, Secret для API-ключа, ограничением ресурсов и пробой здоровья на /health.

XI. Системы контроля версий (Git)

Тестовые задания

1. Стратегии слияния: Объясните разницу между merge commit, fast-forward merge и squash merge. Какой метод предпочтителен для поддержания чистой линейной истории (git log --oneline) в проекте с активными pull request, и какие подводные камни у этого подхода?

2. Переписывание истории: Каковы риски выполнения git rebase для ветки, которая уже была отправлена в удалённый репозиторий и, возможно, склонирована другими разработчиками? Как безопасно исправить коммит в уже опубликованной ветке?

Практическое задание: Вам в main случайно попал коммит с чувствительными данными (пароль в конфиге). Опишите последовательность команд Git для его полного удаления из истории (включая все теги и reflog), минимизируя ущерб для коллаборации.

XII. Машинное обучение и Data Science

Тестовые задания

1. Метрики: В задаче обнаружения мошеннических транзакций (1% положительных) метрика Accuracy показала 99%. Почему эта метрика бесполедна, и какие метрики (Precision, Recall, F1, PR-AUC, ROC-AUC) следует использовать, объяснив бизнес-смысл каждого выбора?

2. Feature Engineering: Что такое "проклятие размерности" (curse of dimensionality) и как оно влияет на производительность модели при добавлении сотен сгенерированных признаков? Какие методы отбора признаков (feature selection) и уменьшения размерности (dimensionality reduction) можно применить?

3. Атаки: Что такое "состязательные атаки" (adversarial attacks) на модели компьютерного зрения? Приведите пример минимальной perturbation, которая заставит модель классифицировать панду как гиббона.

Практическое задание: Проанализируйте датасет с пропусками (missing values), аномалиями и несбалансированными классами. Разработайте и обоснуйте пайплайн предобработки, включая стратегии импутации, кодирования категориальных признаков и борьбы с дисбалансом.

XIII. Распределенные системы и инфраструктура

Тестовые задания

1. Service Mesh: Для чего нужен сервис-меш (service mesh, например, Istio, Linkerd) в микросервисной архитектуре, если у вас уже есть балансировщик нагрузки (load balancer) и API-шлюз (API gateway)?

2. Распределенные транзакции: Почему классические двухфазные коммиты (2PC) плохо подходят для микросервисов, и какие паттерны (Saga, Outbox) используются вместо них для обеспечения согласованности данных в конечном счете (eventual consistency)?

3. Масштабирование: В чем разница между горизонтальным (scaling out) и вертикальным (scaling up) масштабированием? Как выбор между ними влияет на отказоустойчивость (high availability) и стоимость инфраструктуры?

Практическое задание: Спроектируйте схему репликации и шардирования для базы данных PostgreSQL, которая должна обслуживать глобальное приложение с запросами из разных регионов, минимизируя latency и обеспечивая отказоустойчивость.

XIV. Управление ИТ-услугами (ITSM/ITIL)

Тестовые задания

1. SLA/SLO/SLI: Объясните иерархию и взаимосвязь между Service-Level Indicator (SLI), Service-Level Objective (SLO) и Service-Level Agreement (SLA). Приведите пример для сервиса прогнозирования, где SLI — доля успешных запросов с latency <100ms, SLO — 95-й процентиль, а SLA — финансовые санкции при нарушении.

2. Инцидент vs Проблема: Сформулируйте принципиальное различие между управлением инцидентами (Incident Management) и управлением проблемами (Problem Management) в ITIL. К какому процессу отнесёте анализ root cause падения ML-модели в продакшене?

Практическое задание: Разработайте шаблон постмортема (postmortem) для инцидента, связанного с утечкой памяти в сервисе векторизации, который привел к его полной недоступности на 2 часа. Включите разделы: timeline, root cause, impact, действия по исправлению и предотвращению.

XV. Облачные технологии (Cloud)

Тестовые задания

1. Модели обслуживания: Объясните разницу между ответственностью клиента и провайдера в моделях IaaS (например, виртуальная машина), PaaS (например, App Engine) и SaaS (например, Google Workspace). Где проходит граница ответственности за безопасность ОС и патчи?

2. Serverless: В чем преимущества и ограничения serverless-архитектуры (например, AWS Lambda) для запуска инференса ML-моделей по сравнению с выделенными инстансами или Kubernetes?

Практическое задание: Сравните TCO (Total Cost of Ownership) для развертывания одного и того же ML-пайплайна на: 1) собственном железе в дата-центре, 2) управляемом Kubernetes в облаке (GKE/EKS), 3) полностью serverless-архитектуре (Cloud Functions + BigQuery). Учтите капитальные и операционные расходы.

2.3. Рекомендации обучающимся по подготовке к экзамену.

Подготовка к экзамену способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к государственному экзамену, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания. На государственном экзамене студент демонстрирует то, что он приобрел в процессе обучения. В период подготовки к государственному экзамену студенты вновь обращаются к учебно-методическому материалу и закрепляют знания. Подготовка студента к государственному экзамену включает в себя три этапа: самостоятельная работа в течение всего периода обучения; непосредственная подготовка в дни, предшествующие государственному экзамену по темам разделам и темам учебных дисциплин, выносимых на государственную аттестацию. При подготовке к государственному экзамену студентам целесообразно использовать материалы лекций, методические материалы образовательной программы, рекомендованные правовые акты, основную и дополнительную литературу.

В процессе экзаменационного ответа преподавателем оценивается не только знание того или иного вопроса, но и ряд других, не лежащих на поверхности факторов к числу которых, в первую очередь, относится культура гуманитарного знания, профессиональное оперирование терминологией, культура речи студента. Во время ответа на поставленные вопросы надо быть готовым к дополнительным или уточняющим вопросам. Дополнительные вопросы задаются членами государственной комиссии в рамках билета и

связаны, как правило, с неполным ответом. Уточняющие вопросы задаются, чтобы либо конкретизировать мысли студента, либо чтобы студент подкрепил те или иные теоретические положения практикой, либо привлек знания смежных учебных дисциплин. Полный ответ на уточняющие вопросы лишь усиливает эффект общего ответа студента. Итоговая оценка знаний предполагает дифференцированный подход к студенту, учет его индивидуальных способностей, степень усвоения и систематизации основных теоретических положений, понятий и категорий. Оценивается так же культура речи, грамотное комментирование, приведение примеров, умение связывать теорию с практикой, творчески применять знания к неординарным ситуациям, излагать материал доказательно, подкреплять теоретические положения знанием нормативных актов, полемизировать там, где это необходимо.

2.4. Перечень рекомендуемой литературы

№ п/п	Библиографическое описание	Место доступа
1	Рашка, С. Python и машинное обучение: крайне необходимое пособие по новейшей предсказательной аналитике, обязательное для более глубокого понимания методологии машинного обучения : руководство / С. Рашка ; перевод с английского А. В.	https://e.lanbook.com/book/100905?category_pk=1556&publisher__fk=1028&ysclid=lw60n980di451663507

	Логунова. — Москва : ДМК Пресс, 2017. — 418 с. — ISBN 978-5-97060-409-0		
2	Йылдырым, С. Осваиваем архитектуру Transformer / С. Йылдырым, М. Асгари-Ченаглу ; перевод с английского В. С. Яценкова. — Москва : ДМК Пресс, 2022. — 320 с. — ISBN 978-5-93700-106-1	https://e.lanbook.com/book/241148	
3	Алетдинова, А. А. Интеллектуальный анализ больших данных : учебное пособие / А. А. Алетдинова, М. Ш. Муртазина. — Новосибирск : НГТУ, 2023. — 66 с. — ISBN	https://e.lanbook.com/book/404567	

	978-5-7782-4899-1		
4	Павлов, Л. А. Структуры и алгоритмы обработки данных : учебник для вузов / Л. А. Павлов, Н. В. Первова. — 3-е изд., стер. — Санкт-Петербург : Лань, 2021. — 256 с. — ISBN 978-5-8114-7259-8	https://e.lanbook.com/book/156929?category=1540&ysclid=lw4yv0hohi198485137	
5	Пантелеев, Е. Р. Алгоритмы и структуры данных : учебное пособие / Е. Р. Пантелеев, А. Л. Алыкова. — Иваново : ИГЭУ, 2018. — 142 с.	https://e.lanbook.com/book/154576	
6	Таблицы интегралов и другие математические формулы Г.Б. Двайт; Ред. К.А. Семендяев; Пер. с англ. Н.В. Леви	НТБ (уч.6); НТБ (фб.); НТБ (чз.2)	

	Однотомное издание "Лань" , 2005		
7	Ростовцев, В. С. Искусственные нейронные сети / В. С. Ростовцев. — 4-е изд., стер. — Санкт-Петербург : Лань, 2024. — 216 с. — ISBN 978-5-507-47362-5	https://e.lanbook.com/book/364517?ysclid=m1rz4vi0fi416423386	
8	Соробин, А. Б. Сверточные нейронные сети: примеры реализаций : учебно-методическое пособие / А. Б. Соробин. — Москва : РТУ МИРЭА, 2020. — 159 с.	https://e.lanbook.com/book/163853?category=1547&ysclid=m1rz5u3q5x476258334	
9	Гудфеллоу, Я. Глубокое обучение / Я. Гудфеллоу, И. Бенджио, А. Курвилль ; перевод с английского	https://e.lanbook.com/book/107901	

	А. А. Слинкина. — 2-е изд. — Москва : ДМК Пресс, 2018. — 652 с. — ISBN 978-5-97060-618-6		
10	Флах, П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных / П. Флах. — Москва : ДМК Пресс, 2015. — 400 с. — ISBN 978-5-97060-273-7	https://e.lanbook.com/book/69955	
11	Голицына, О. Л. Базы данных : учебное пособие / О.Л. Голицына, Н.В. Максимов, И.И. Попов. — 4-е изд., перераб. и доп. — Москва : ФОРУМ : ИНФРА-М, 2020. — 400 с. —	https://znanium.ru/catalog/document?id=362825&ysclid=lweo1o32q4992160167#bib	

	(Среднее профессиональное образование). - ISBN 978-5-00091-601-8		
12	Карпова, Т. С. Базы данных: модели, разработка, реализация : учебное пособие / Т. С. Карпова. — 2-е изд. — Москва : ИНТУИТ, 2016. — 403 с.	https://e.lanbook.com/book/100575?category_pk=1556&ysclid=lweo4bpupg644993495	
13	Габдуллин, Н. М. Развитие человеческого капитала и цифровой экономики в регионах России: факторный и кластерный анализ : монография / Н. М. Габдуллин. — Казань : КФУ, 2019. — 268 с. — ISBN 978-5-00130-291-9	https://e.lanbook.com/book/173018	
14	Гласснер, Э. Глубокое обучение	https://e.lanbook.com/book/131710	

	без математики. Том 2. Практика : руководство / Э. Гласснер ; перевод с английского В. А. Яроцкого. — Москва : ДМК Пресс, 2020. — 610 с. — ISBN 978-5-97060-767-1		
15	Гультяева, Т. А. Методы статистического обучения в задачах регрессии и классификации : монография / Т. А. Гультяева, А. А. Попов, А. С. Саутин. — Новосибирск : НГТУ, 2016. — 323 с. — ISBN 978-5-7782-2817-7	https://e.lanbook.com/book/118291	
16	Кук, Д. Машинное обучение с использованием библиотеки	https://e.lanbook.com/book/97353	

	H2O / Д. Кук ; перевод с английского А. Б. Огурцова. — Москва : ДМК Пресс, 2018. — 250 с. — ISBN 978-5-97060-508-0		
17	Флах, П. Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных / П. Флах. — Москва : ДМК Пресс, 2015. — 400 с. — ISBN 978-5-97060-273-7	https://e.lanbook.com/book/69955	
18	Шалев-Шварц, Ш. Идеи машинного обучения : учебное пособие / Ш. Шалев-Шварц, Бен-ДавидШ. ; перевод с английского А. А. Слинкина. — Москва :	https://e.lanbook.com/book/131686	

	ДМК Пресс, 2019. — 436 с. — ISBN 978-5-97060-673-5		
19	Гудфеллоу, Я. Глубокое обучение / Я. Гудфеллоу, И. Бенджио, А. Курвилль ; перевод с английского А. А. Слинкина. — 2-е изд. — Москва : ДМК Пресс, 2018. — 652 с. — ISBN 978-5-97060-618-6	https://e.lanbook.com/book/107901	
20	Болотова, Ю. А. Методы и алгоритмы интеллектуальной обработки цифровых изображений : учебное пособие / Ю. А. Болотова, А. А. Друки, В. Г. Спицын. — Томск : ТПУ, 2016. — 208 с. — ISBN 978-5-4387-0710-3	https://e.lanbook.com/book/107751	
21	Шапиро, Л. Компьютер	https://e.lanbook.com/book/135496	

	ное зрение : учебное пособие / Л. Шапиро, Д. Стокман ; под редакцией С. М. Соколова ; перевод с английского А. А. Богуславского. — 4-е изд. — Москва : Лаборатория знаний, 2020. — 763 с. — ISBN 978-5-00101-696-0		
22	Нуньес-Иглесиас, Х. Элегантный SciPy / Х. Нуньес-Иглесиас, в. д. Уолт, Х. Дэшноу. — Москва : ДМК Пресс, 2018. — 266 с. — ISBN 978-5-97060-600-1	https://e.lanbook.com/book/116124	
23	Волосова, А. В. Технологии искусственного интеллекта в ULS-системах / А. В.	https://e.lanbook.com/book/370217	

	Волосова. — 2-е изд., испр. — Санкт-Петербург : Лань, 2024. — 308 с. — ISBN 978-5-507-45885-1		
24	Ненашев, В. А. Компьютерное зрение. Анализ, обработка и моделирование : учебное пособие / В. А. Ненашев. — Санкт-Петербург : ГУАП, 2022. — 78 с. — ISBN 978-5-8088-1806-4	https://e.lanbook.com/book/341057	

3. Перечень компетенций, которые должны быть сформированы у обучающихся в результате освоения образовательной программы.

ОПК-1 - Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности;

ОПК-2 - Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;

ОПК-3 - Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;

ОПК-4 - Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью;

ОПК-5 - Способен инсталлировать программное и аппаратное обеспечение для информационных и автоматизированных систем;

ОПК-6 - Способен разрабатывать бизнес-планы и технические задания на оснащение отделов, лабораторий, офисов компьютерным и сетевым оборудованием;

ОПК-7 - Способен участвовать в настройке и наладке программно-аппаратных комплексов;

ОПК-8 - Способен разрабатывать алгоритмы и программы, пригодные для практического применения;

ОПК-9 - Способен осваивать методики использования программных средств для решения практических задач;

ПК-1 - Способен анализировать большие данные с использованием существующей в организации методологической и технологической инфраструктуры;

ПК-2 - Способен осуществлять концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности ;

ПК-3 - Способен осуществлять разработку требований и проектирование программного обеспечения;

ПК-4 - Способен выполнять работы и управлять работами по созданию (модификации) и сопровождению ИС, автоматизирующих задачи организационного управления и бизнес-процессы;

ПК-5 - Способен управлять проектами в области ИТ на основе полученных планов проектов в условиях, когда проект не выходит за пределы утвержденных параметров;

ПК-6 - Способен осуществлять непосредственное руководство процессами разработки программного обеспечения ;

ПК-7 - Способен к организации процессов разработки программного обеспечения ;

ПК-8 - Способен обеспечивать защиту информации в автоматизированных системах в процессе их эксплуатации;

ПК-9 - Способен обеспечивать информационную безопасность на уровне БД;

УК-1 - Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач;

УК-2 - Способен определять круг задач в рамках поставленной цели и выбирать оптимальные способы их решения, исходя из действующих правовых норм, имеющихся ресурсов и ограничений;

УК-3 - Способен осуществлять социальное взаимодействие и реализовывать свою роль в команде;

УК-4 - Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(ых) языке(ах);

УК-5 - Способен воспринимать межкультурное разнообразие общества в социально-историческом, этическом и философском контекстах;

УК-6 - Способен управлять своим временем, выстраивать и реализовывать траекторию саморазвития на основе принципов образования в течение всей жизни;

УК-7 - Способен поддерживать должный уровень физической подготовленности для обеспечения полноценной социальной и профессиональной деятельности;

УК-8 - Способен создавать и поддерживать в повседневной жизни и в профессиональной деятельности безопасные условия жизнедеятельности для сохранения природной среды, обеспечения устойчивого развития общества, в том числе при угрозе и возникновении чрезвычайных ситуаций и военных конфликтов;

УК-9 - Способен использовать базовые дефектологические знания в социальной и профессиональной сферах;

УК-10 - Способен принимать обоснованные экономические решения в различных областях жизнедеятельности;

УК-11 - Способен формировать нетерпимое отношение к проявлениям экстремизма, терроризма, коррупционному поведению и противодействовать им в профессиональной деятельности.

4. Критерии оценки результатов итоговой (государственной итоговой) аттестации.

4.1. Критерии оценки результатов сдачи экзамена.

Шкала оценивания	Критерии
Оценивание результатов	Правильно выполнено 90% и более заданий (отсутствие ответа приравнивается к неверно выполненному заданию). На очной части экзамена: - Задание выполнено полностью, - ответ обоснованно, - выводы и предложения аргументированы и оформлены должным образом Выпускник демонстрирует навыки публичной дискуссии, защиты собственных научных и/или практических идей, предложений и рекомендаций, культура общения с аудиторией на высоком уровне.
Оценивание результатов	Приведено 80-89% правильных ответов из тестовых заданий (отсутствие ответа приравнивается к неверно выполненному заданию). На очной части экзамена: - задание выполнено полностью, но допущено незначительные неточности в расчетах или оформлении; - или при условии надлежащего оформления, задания, выполнено не меньше, как на 80%
Оценивание результатов	Приведено свыше 70-79% правильных ответов из тестовых заданий (отсутствие ответа приравнивается к неверно выполненному заданию). На очной части экзамена: - задание выполнено не менее как на 70% при условии надлежащего оформления; или не более как на 80% при условии незначительных ошибок в расчетах
Оценивание результатов	Приведено менее 55% правильных ответов из тестовых заданий (отсутствие ответа приравнивается к неверно выполненному заданию). На очной части экзамена: - задание выполнено менее как на 50%, допущены принципиальные ошибки в расчетах

Авторы:

доцент, к.н. Академии "Высшая инженерная школа" О.Б. Проневич

Согласовано:

Директор Д.В. Паринов

Председатель учебно-методической комиссии Д.В. Паринов