Оценочные материалы дополнительной профессиональной программы повышения квалификации
«Мультиагентные системы и LLM»

Пример оценочных материалы для текущего контроля

Примерные вопросы для собеседования

Опишите архитектуру трансформеров и принцип работы механизма внимания (attention).

В чём заключаются основные философские и архитектурные различия между фреймворками LangChain и AutoGen для построения агентов?

Какие роли (например, менеджер, эксперт, критик) и типы коммуникации могут выполнять агенты в мультиагентной системе?

Что такое промпт-инжиниринг? Приведите примеры эффективных продвинутых техник, таких как Chain-of-Thought или ReAct.

Объясните назначение и ключевые преимущества протокола Model Context Protocol (MCP).

Что такое «контекстное окно» (context window) у LLM и какие основные стратегии существуют для работы с длинными контекстами или их ограничениями?

Опишите типичный жизненный цикл и основные компоненты (планировщик, память, инструменты) автономного агента на основе LLM.

Какие основные проблемы координации (например, конфликты, циклы) возникают в мультиагентных системах и как их можно решить?

В чём разница между тонкой настройкой модели (fine-tuning), обучением с подкреплением (RLHF) и промпт-инжинирингом? Когда какой подход предпочтителен?

Что такое «оркестрация» (orchestration) в контексте работы с LLM и каковы её ключевые задачи поверх простого API-вызова?

Как в мультиагентной системе может быть организована общая память (shared memory) или состояние для обмена информацией между агентами?

Какие основные риски безопасности (например, prompt injection, неконтролируемые цепочки действий) присущи мультиагентным системам и как можно их mitigate (снижать)?

Какие существуют практические методы для оценки качества, стабильности и безопасности ответов, генерируемых мультиагентной системой?

Как можно оптимизировать стоимость (cost) и задержки (latency) при работе мультиагентной системы с использованием платных API моделей?

Приведите пример нетривиального реального кейса, где мультиагентный подход имеет decisive (решающее) преимущество перед одиночным агентом или традиционной автоматизацией.

Пример оценочных материалов для итоговой аттестации

Методические указания к подготовке итоговой аттестации

Демонстрация работающей мультиагентной системы для решения конкретной задачи.

Требования к проекту:

Слушатель представляет работающее программное решение (прототип) на базе одного из изученных фреймворков (например, LangChain, AutoGen, CrewAI), в котором минимум два автономных агента на основе LLM взаимодействуют для достижения общей цели. Система должна выполнять законченный, нетривиальный сценарий (например, анализ проблемы, планирование, выполнение шагов, генерацию итогового документа).

Детализированные критерии оценки

Максимальный итоговый балл: 100%. Для получения результата «Зачёт» необходимо набрать не менее 70% от общей суммы баллов.

Критерий 1: Работоспособность системы (до 40 баллов)

Оценивается способность системы выполнить поставленную задачу от начала до конца в режиме демонстрации.

0-25 баллов: Система запускается, но не выполняет задачу до конца. Агенты не взаимодействуют, процесс обрывается на ошибке, конечный результат не формируется или является бессмысленным.

26-35 баллов: Система выполняет ключевые шаги задачи, демонстрирует взаимодействие агентов и выдает осмысленный результат, но с заметными ошибками, неточностями или требует ручного вмешательства на некоторых этапах.

36-40 баллов: Система полностью автономно и стабильно выполняет поставленную задачу. Демонстрация проходит без сбоев, результат является полным, релевантным и соответствует исходному ТЗ. Четко виден последовательный workflow с передачей данных между агентами.

Критерий 2: Правильность архитектуры решения (до 30 баллов)

Оценивается обоснованность проектных решений и соответствие принципам мультиагентных систем.

0-15 баллов: Архитектура неоправданно проста (например, могла бы быть реализована одним агентом) или избыточно сложна. Роли агентов не дифференцированы, логика их взаимодействия не ясна. Выбор фреймворка не обоснован.

16-25 баллов: Архитектура в целом логична, выделены различные роли агентов (например, "аналитик", "исполнитель"). Имеется описание workflow, но могут отсутствовать ключевые компоненты (например, механизм координации, общая память) или их необходимость не объяснена.

26-30 баллов: Архитектура элегантна и оптимальна для задачи. Четко определены и обоснованы роли каждого агента, их зоны ответственности и протоколы взаимодействия. Применены соответствующие паттерны (координатор, черная доска, рынок). Выбор инструментов (фреймворк, LLM, серверы) технически обоснован в презентации или документации.

Критерий 3: Качество кода и документации (до 30 баллов)

Оценивается качество реализации, сопровождающих материалов и возможность проверки работы системы третьим лицом.

Качество кода (до 15 баллов):

Код читаем, имеет понятную структуру и комментарии в ключевых местах.

Отсутствуют грубые ошибки, используется современный и идиоматичный для фреймворка синтаксис.

Конфиденциальные данные (API-ключи) вынесены в переменные окружения или конфигурационные файлы, не закоммичены в репозиторий.

Присутствует базовая обработка ошибок.

Качество документации (до 15 баллов):

README-файл: Содержит четкое описание задачи, инструкцию по установке зависимостей и запуску проекта. Указаны используемые API-ключи и модели LLM.

Архитектурное описание: Схема (блок-схема, диаграмма последовательности) и текстовое описание workflow системы, ролей агентов, форматов их сообщений.

Примеры работы: Приведены примеры входных данных и соответствующих им выходных результатов системы.

Процедура защиты (демонстрации)

Презентация (3-5 минут):

Краткое описание решаемой проблемы.

Демонстрация архитектурной схемы системы и обоснование выбора ролей агентов.

Объяснение выбора инструментов (фреймворк, модели LLM).

Живая демонстрация (5-7 минут):

Запуск системы и выполнение задачи на конкретном примере.

Комментарии о ключевых точках взаимодействия агентов.

Показ конечного результата.

Ответы на вопросы (3-5 минут):

Вопросы по архитектуре, коду, принятым решениям и потенциальным улучшениям от комиссии.

Итоговое решение выносится на основании суммы баллов по всем критериям.

«Зачтено» присваивается, если суммарный балл составляет 70% и более от максимума (70 баллов из 100).