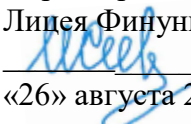


Федеральное государственное образовательное бюджетное
учреждение высшего образования
«Финансовый университет при Правительстве Российской Федерации»

УТВЕРЖДАЮ

Директор

Лицея Финуниверситета

 И.В. Сивцова

«26» августа 2026 г.

РАБОЧАЯ ПРОГРАММА КУРСА ПО ВЫБОРУ
«ВАЙБКОДИНГ И РАЗРАБОТКА ИИ-АГЕНТОВ»

Среднее общее образование

Рабочая программа согласована
на заседании педагогического совета
Протокол № 1 от «26» августа 2026 г.

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Программа по предмету «Вайбкодинг и разработка ИИ-агентов» на уровне среднего общего образования разработана на основе требований к результатам освоения основной образовательной программы среднего общего образования, а также федеральной рабочей программы воспитания.

Программа по предмету «Вайбкодинг и разработка ИИ-агентов» дает представление о целях, общей стратегии обучения, воспитания и развития обучающихся средствами учебного предмета «Вайбкодинг и разработка ИИ-агентов» на углублённом уровне, устанавливает обязательное предметное содержание, предусматривает его структурирование по разделам и темам курса, определяет распределение его по классам (годам изучения), дает примерное распределение учебных часов по тематическим разделам курса и рекомендуемую (примерную) последовательность их изучения с учетом межпредметных связей (информатика, математика, основы финансовой грамотности), логики учебного процесса, возрастных особенностей обучающихся.

Программа по предмету «Вайбкодинг и разработка ИИ-агентов» определяет количественные и качественные характеристики учебного материала для каждого года изучения, в том числе для содержательного наполнения разного вида контроля (промежуточной аттестации обучающихся, защиты проектов). Программа является основой для составления авторских учебных программ, поурочного планирования, а также для разработки практико-ориентированных заданий и проектов учителем.

Вайбкодинг и разработка ИИ-агентов в среднем общем образовании отражают:

- сущность современного подхода к программированию как диалога с генеративными моделями искусственного интеллекта (вайбкодинг), а также принципы проектирования автономных ИИ-агентов, способных использовать инструменты и взаимодействовать с внешней средой;
- основные области применения агентного подхода: автоматизация рутинных задач, финансовое планирование, анализ данных, образовательные ассистенты, персональные помощники;
- междисциплинарный характер дисциплины, соединяющей информатику, лингвистику (естественно-языковое взаимодействие), психологию (проектирование диалогов) и этику.

Курс вайбкодинга и разработки ИИ-агентов для уровня среднего общего образования является пропедевтическим этапом подготовки обучающихся к профессиональной деятельности в области искусственного интеллекта. Он опирается на содержание курса информатики уровня основного общего образования (базовые алгоритмические конструкции, основы программирования на Python), а также на практический опыт использования цифровых технологий, и позволяет выйти на новый уровень – создание собственных интеллектуальных систем.

Результаты углублённого уровня изучения учебного предмета «Вайбкодинг и разработка ИИ-агентов» ориентированы на получение компетентностей для последующей профессиональной деятельности как в области ИИ и программной инженерии, так и в смежных сферах (финансовые технологии, бизнес-аналитика,

цифровой маркетинг). Они включают в себя:

- овладение ключевыми понятиями: промпт, инференс, контекстное окно, function calling, ИИ-агент, многоагентная система, RAG, ReAct, фреймворк LangChain (или аналог); распознавание соответствующих им признаков и взаимосвязей, способность демонстрировать различные подходы к проектированию агентов;
- умение решать типовые практические задачи: составление эффективных промптов, интеграция БЯМ через API, разработка агентов с памятью и инструментами, отладка и оценка качества агентов;
- наличие представлений о данной предметной области как формирующейся технологической парадигме, о её связи с машинным обучением, обработкой естественного языка и программной инженерией.

В рамках углубленного уровня изучения вайбкодинга и разработки ИИ-агентов обеспечивается целенаправленная подготовка обучающихся к продолжению образования в организациях профессионального образования по специальностям, непосредственно связанным с искусственным интеллектом, таким как разработка интеллектуальных систем, анализ данных, большие языковые модели, робототехника, человеко-машинное взаимодействие, автоматизация бизнес-процессов, ИИ-инжиниринг.

Основная цель изучения учебного предмета «Вайбкодинг и разработка ИИ-агентов» на углубленном уровне среднего общего образования — формирование у обучающихся практических навыков взаимодействия с генеративным ИИ и создания простых ИИ-агентов, развитие творческого и критического мышления в контексте современных ИИ-технологий, а также подготовка к осознанному выбору карьеры в цифровой экономике. В связи с этим изучение курса в 10–11 классах должно обеспечить:

- сформированность мировоззрения, основанного на понимании роли генеративного искусственного интеллекта и агентных систем в современном обществе, производстве, финансах и повседневной жизни;
- сформированность основ промпт-инжиниринга и алгоритмического мышления применительно к постановке задач для БЯМ;
- сформированность умений различать корректные и некорректные результаты работы ИИ (галлюцинации), критически оценивать сгенерированный код и текст, проверять его на достоверность и безопасность;
- сформированность представлений о влиянии ИИ-агентов на жизнь человека в обществе, понимание социального, экономического, юридического и этического контекстов их использования;
- принятие правовых и этических аспектов разработки ИИ-агентов (ответственность за их действия, приватность данных, запрет на создание вредоносных ботов);
- создание условий для развития навыков проектной деятельности, творчества и саморазвития через реализацию собственных агентных проектов.

В содержании учебного предмета «Вайбкодинг и разработка ИИ-агентов» выделяются четыре тематических раздела.

Раздел «Промпт-инжиниринг и основы работы с БЯМ» посвящен теории и

практике составления эффективных промптов, пониманию работы генеративных моделей (токенизация, контекст, температура), сравнению различных БЯМ и их возможностей.

Раздел «Программное взаимодействие с БЯМ через API» включает в себя основы HTTP и REST API, написание скриптов на Python для отправки запросов, обработку ответов, управление контекстом, использование механизма вызова инструментов.

Раздел «Разработка ИИ-агентов» направлен на проектирование и реализацию агентов с памятью, инструментами, планированием; изучение фреймворков (LangChain или аналог), создание многоагентных систем, оценку качества работы агента.

Раздел «Проектная деятельность и развёртывание агентов» посвящен созданию веб-интерфейсов или телеграм-ботов для агентов, размещению их на учебном хостинге, обеспечению безопасности и этичности, а также выполнению итогового группового проекта.

В приведённом далее содержании учебного предмета курсивом выделены дополнительные темы, которые не входят в обязательную программу обучения, но могут быть предложены для изучения отдельным мотивированным и способным обучающимся (например, тонкая настройка моделей, векторные базы данных, обучение с подкреплением для агентов).

Углубленный уровень изучения вайбкодинга и разработки ИИ-агентов рекомендуется для технологического профиля, ориентированного на инженерную, информационную и предпринимательскую сферы деятельности. Он обеспечивает: подготовку обучающихся, ориентированных на специальности в области искусственного интеллекта, робототехники, программной инженерии и финансовых технологий; участие в проектной и исследовательской деятельности, связанной с современными БЯМ и агентными системами; подготовку к участию в олимпиадах по искусственному интеллекту и хакатонах по ИИ.

Последовательность изучения тем в пределах одного года обучения может быть изменена по усмотрению учителя при подготовке рабочей программы и поурочного планирования.

Общее число часов, рекомендованных для изучения вайбкодинга и разработки ИИ-агентов, – 136 часов: в 10 классе – 68 часов, в 11 классе – 68 часов.

СОДЕРЖАНИЕ ОБУЧЕНИЯ

10 КЛАСС

Промпт-инжиниринг и основы работы с БЯМ

Понятие вайбкодинга как программирования через диалог с генеративной нейросетью. Роль человека в цикле: постановка задачи, проверка и доводка результатов. Принципы работы больших языковых моделей (БЯМ) на концептуальном уровне: токенизация текста, предсказание следующего токена, контекстное окно, температура генерации. Различие между инференсом и дообучением.

Промпт как форма постановки задачи для БЯМ. Типы промптов: без примеров (zero-shot), с одним примером (one-shot) с несколькими примерами (few-shot), цепочка мыслей (Chain-of-Thought), промпты с назначением роли. Техники уточнения промптов: итеративное уточнение, добавление ограничений, указание формата вывода. Сравнение эффективности разных промптов на одной задаче.

Инструменты вайбкодинга: ChatGPT, DeepSeek, Qwen, GitHub Copilot, Cursor, Codeium. Возможности и ограничения каждого инструмента. Правила безопасного использования: запрет на передачу персональных данных, проверка сгенерированного кода на вредоносные действия, соблюдение лицензий.

Этические аспекты использования сгенерированного кода в учебных проектах. Тонкая настройка промптов для предметных областей (финансы, математика, естественные науки)¹.

Программное взаимодействие с БЯМ через API

Основы HTTP и REST API. Структура запроса и ответа. Аутентификация с помощью API-ключа. Получение и хранение ключа (безопасное хранение, переменные окружения). Отправка POST-запросов к публичным БЯМ (OpenRouter, DeepSeek API, YandexGPT API) на языке Python с использованием библиотеки requests.

Формирование тела запроса: параметры model, messages, temperature, max_tokens, top_p. Обработка ответа: извлечение текста из JSON, подсчёт использованных токенов. Обработка ошибок: неверный ключ, превышение лимитов, недоступность сервиса.

Управление контекстом диалога: хранение истории сообщений в списке, ограничение длины контекста, обрезка или суммаризация старых сообщений. Реализация консольного чат-бота с памятью.

Механизм вызова функций (вызов функций и инструментов). Описание доступных функций в запросе. Обработка ответа БЯМ с требованием вызова функции. Выполнение функции на стороне клиента и возврат результата. Примеры: вызов калькулятора, получение текущей даты, работа с простой базой данных.

Асинхронная отправка запросов для повышения производительности.

Разработка простых ИИ-агентов

Понятие ИИ-агента: автономность, реактивность, проактивность, способность использовать инструменты и адаптироваться к изменению условий. Классификация агентов: простые рефлекторные агенты, агенты с памятью, целеориентированные агенты, агенты с полезностью. Отличие ИИ-агента от обычного чат-бота.

Архитектура агента на основе БЯМ: цикл «восприятие – планирование – действие – наблюдение». Проектирование промпта системной роли, описывающего поведение агента. Реализация агента для выполнения одной задачи (например, агент-конвертер валют: при вводе суммы и валюты вызывает функцию получения курса через API и выдает результат).

¹ Курсивом выделен дополнительный материал, который не входит в содержание аттестации по предмету.

Создание агента-помощника для решения финансовых задач: расчёт сложного процента, планирование бюджета, подбор инвестиционных стратегий (упрощённо). Использование вызова функций для интеграции с внешними API (курсы валют, погода, простые математические библиотеки).

Агент с краткосрочной памятью: сохранение последних действий и ответов в переменных, учёт контекста диалога. Реализация агента-модератора: проверка текста на наличие запрещённых тем (нецензурная лексика, угрозы, персональные данные). Отладка агента: логирование запросов и ответов, анализ «галлюцинаций» и их устранение через усиление ограничений в промпте.

Интеграция агента с веб-интерфейсом на базе Streamlit (минимальный вариант). Использование локальных БЯМ (через Ollama) для работы без доступа к сети.

11 КЛАСС

Продвинутая архитектура ИИ-агентов

Контекстная память в агентах: буфер диалога, скользящее окно, суммаризация истории. Понятие векторной базы данных и Retrieval-Augmented Generation (RAG) на концептуальном уровне: преобразование текста в эмбединги, поиск релевантных фрагментов, добавление их в промпт. Реализация простейшего RAG с использованием готовых библиотек (например, Chroma, FAISS) в учебном проекте.

Фреймворки для разработки агентов: LangChain (базовые компоненты: Chains, Tools, Agents, Memory). Создание цепочек (SequentialChain, LLMChain). Использование готовых инструментов (Wikipedia, Search, Calculator). Проектирование агента с выбором инструментов на основе запроса.

Режимы планирования действий: ReAct (Reasoning + Acting) – чередование рассуждения и действий, Plan-and-Execute – отдельное планирование, затем выполнение. Сравнение эффективности на задачах с несколькими шагами (например, «найди курс доллара, умножь на 100 и переведи в евро»).

Оценка качества работы агента: метрики точности выполнения шагов, полнота достижения цели, время ответа, субъективная полезность с точки зрения пользователя. Инструменты трассировки выполнения агента (LangSmith, простые логи).

Адаптивное обучение агента на основе обратной связи от пользователя (обновление промпта или базы знаний).

Многоагентные системы и взаимодействие

Понятие многоагентной системы (МАС). Кооперация и координация агентов. Роли агентов: координатор, исполнитель, критик, супервайзер. Преимущества и сложности МАС по сравнению с одним агентом.

Реализация пары агентов «исполнитель – критик». Исполнитель генерирует ответ или код, критик анализирует результат, указывает на ошибки и предлагает улучшения. Заикливание: ограничение числа итераций.

Агент-маршрутизатор (router): анализ пользовательского запроса, выбор наиболее подходящего специализированного агента (агент по математике, агент по финансам, агент-переводчик). Пример: финансовая МАС – один агент анализирует

бюджет, другой ищет инвестиционные идеи (через API новостей), третий формирует отчёт.

Балансировка нагрузки и предотвращение бесконечных циклов. Взаимодействие агентов через общую память (переменные, база данных). Распределение задач между агентами с использованием очередей сообщений.

Соревновательные МАС (игровые агенты).

Развёртывание и проектная деятельность

Создание пользовательского интерфейса для агента: веб-приложение на Streamlit или Gradio (поле ввода, вывод истории диалога, кнопки для вызова действий). Разработка Telegram-бота на Python с использованием библиотеки python-telegram-bot или Aiogram: обработка команд, диалогов, интеграция с агентом.

Развёртывание агента на удалённом сервере: платформы Render, PythonAnywhere, Replit. Настройка переменных окружения для API-ключей. Обеспечение непрерывной работы (keep-alive, бесплатные хостинги с ограничениями).

Безопасность и этика при развёртывании: фильтрация входящих промптов (санитайзер), ограничение прав агента (запрет на выполнение системных команд), защита от промпт-инъекций, журналирование для аудита. Правовые аспекты: ответственность за действия агента, информирование пользователя о том, что он общается с ИИ.

Итоговый групповой проект: разработка ИИ-агента для решения прикладной задачи из области школьной жизни, финансовой грамотности, помощи в учёбе или социального взаимодействия. Этапы проекта: анализ требований, проектирование архитектуры, выбор инструментов, реализация, тестирование, документирование, подготовка презентации и демонстрации. Защита проекта перед классом и (возможно) приглашёнными экспертами.

Размещение агента в облачном сервисе с использованием серверных функций (например, Yandex Cloud Functions, AWS Lambda). Подготовка агента к участию в хакатоне по искусственному интеллекту.

ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОСВОЕНИЯ КУРСА ПО ВЫБОРУ «ВАЙБКОДИНГ И РАЗРАБОТКА ИИ-АГЕНТОВ» НА УРОВНЕ СРЕДНЕГО ОБЩЕГО ОБРАЗОВАНИЯ

ЛИЧНОСТНЫЕ РЕЗУЛЬТАТЫ

Личностные результаты отражают готовность и способность обучающихся руководствоваться сформированной внутренней позицией личности, системой ценностных ориентаций, позитивных внутренних убеждений, соответствующих традиционным ценностям российского общества, расширение жизненного опыта и опыта деятельности в процессе реализации средствами учебного предмета основных направлений воспитательной деятельности.

В результате изучения предмета «Вайбкодинг и разработка ИИ-агентов» на уровне среднего общего образования у обучающегося будут сформированы следующие личностные результаты:

1) гражданского воспитания:

осознание своих конституционных прав и обязанностей, уважение закона и правопорядка, соблюдение основополагающих норм информационного права и информационной безопасности;

готовность противостоять идеологии экстремизма, национализма, ксенофобии, дискриминации по социальным, религиозным, расовым, национальным признакам в виртуальном пространстве;

2) патриотического воспитания:

ценностное отношение к историческому наследию, достижениям России в науке, искусстве, технологиях, понимание значения информатики как науки в жизни современного общества;

3) духовно-нравственного воспитания:

сформированность нравственного сознания, этического поведения; способность оценивать ситуацию и принимать осознанные решения, ориентируясь на морально-нравственные нормы и ценности, в том числе в сети Интернет;

4) эстетического воспитания:

эстетическое отношение к миру, включая эстетику научного и технического творчества;

способность воспринимать различные виды искусства, в том числе основанного на использовании информационных технологий;

5) физического воспитания:

сформированность здорового и безопасного образа жизни, ответственного отношения к своему здоровью, в том числе за счет соблюдения требований безопасной эксплуатации средств информационных и коммуникационных технологий;

6) трудового воспитания:

готовность к активной деятельности технологической и социальной направленности, способность инициировать, планировать и самостоятельно выполнять такую деятельность;

интерес к сферам профессиональной деятельности, связанным с информатикой, программированием и информационными технологиями, основанными на достижениях науки информатики и научно-технического прогресса, умение совершать осознанный выбор будущей профессии и реализовывать собственные жизненные планы;

готовность и способность к образованию и самообразованию на протяжении всей жизни;

7) экологического воспитания:

осознание глобального характера экологических проблем и путей их решения,

в том числе с учетом возможностей информационно-коммуникационных технологий;

8) ценности научного познания:

сформированность мировоззрения, соответствующего современному уровню развития науки, достижениям научно-технического прогресса и общественной практики, за счет понимания роли информационных ресурсов, информационных процессов и информационных технологий в условиях цифровой трансформации многих сфер жизни современного общества;

осознание ценности научной деятельности, готовность осуществлять проектную и исследовательскую деятельность индивидуально и в группе.

В процессе достижения личностных результатов освоения программы по информатике у обучающихся совершенствуется эмоциональный интеллект, предполагающий сформированность:

саморегулирования, включающего самоконтроль, умение принимать ответственность за свое поведение, способность адаптироваться к эмоциональным изменениям и проявлять гибкость, быть открытым новому;

внутренней мотивации, включающей стремление к достижению цели и успеху, оптимизм, инициативность, умение действовать, исходя из своих возможностей;

эмпатии, включающей способность понимать эмоциональное состояние других, учитывать его при осуществлении коммуникации, способность к сочувствию и сопереживанию;

социальных навыков, включающих способность выстраивать отношения с другими людьми, заботиться, проявлять интерес и разрешать конфликты.

МЕТАПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

В результате изучения вайбкодинга и разработки ИИ-агентов на уровне среднего общего образования у обучающегося будут сформированы метапредметные результаты, отраженные в универсальных учебных действиях, а именно — познавательные универсальные учебные действия, коммуникативные универсальные учебные действия, регулятивные универсальные учебные действия, совместная деятельность

Познавательные универсальные учебные действия

Базовые логические действия:

самостоятельно формулировать и актуализировать проблему, рассматривать ее всесторонне;

устанавливать существенный признак или основания для сравнения, классификации и обобщения;

определять цели деятельности, задавать параметры и критерии их достижения;

выявлять закономерности и противоречия в рассматриваемых явлениях; разрабатывать план решения проблемы с учетом анализа имеющихся материальных и нематериальных ресурсов;

вносить коррективы в деятельность, оценивать соответствие результатов целям, оценивать риски последствий деятельности;

координировать и выполнять работу в условиях реального, виртуального и комбинированного взаимодействия;

развивать креативное мышление при решении жизненных проблем.

Базовые исследовательские действия:

владеть навыками учебно-исследовательской и проектной деятельности, навыками разрешения проблем, способностью и готовностью к самостоятельному поиску методов решения практических задач, применению различных методов познания;

осуществлять различные виды деятельности по получению нового знания, его интерпретации, преобразованию и применению в различных учебных ситуациях, в том числе при создании учебных и социальных проектов;

формировать научный тип мышления, владеть научной терминологией, ключевыми понятиями и методами;

ставить и формулировать собственные задачи в образовательной деятельности и жизненных ситуациях;

выявлять причинно-следственные связи и актуализировать задачу, выдвигать гипотезу ее решения, находить аргументы для доказательства своих утверждений, задавать параметры и критерии решения;

анализировать полученные в ходе решения задачи результаты, критически оценивать их достоверность, прогнозировать изменение в новых условиях;

давать оценку новым ситуациям, оценивать приобретенный опыт;

осуществлять целенаправленный поиск переноса средств и способов действия в профессиональную среду;

уметь переносить знания в познавательную и практическую области жизнедеятельности;

уметь интегрировать знания из разных предметных областей;

выдвигать новые идеи, предлагать оригинальные подходы и решения, ставить проблемы и задачи, допускающие альтернативные решения.

Работа с информацией:

владеть навыками получения информации из источников разных типов, самостоятельно осуществлять поиск, анализ, систематизацию и интерпретацию информации различных видов и форм представления;

создавать тексты в различных форматах с учетом назначения информации и целевой аудитории, выбирая оптимальную форму представления и визуализации;

оценивать достоверность, легитимность информации, ее соответствие правовым и морально-этическим нормам;

использовать средства информационных и коммуникационных технологий в решении когнитивных, коммуникативных и организационных задач с соблюдением требований эргономики, техники безопасности, гигиены, ресурсосбережения, правовых и этических норм, норм информационной безопасности;

владеть навыками распознавания и защиты информации, информационной безопасности личности.

Коммуникативные универсальные учебные действия

Общение:

осуществлять коммуникации во всех сферах жизни;

распознавать невербальные средства общения, понимать значение социальных знаков, распознавать предпосылки конфликтных ситуаций и смягчать конфликты;

владеть различными способами общения и взаимодействия, аргументированно вести диалог, уметь смягчать конфликтные ситуации;

развернуто и логично излагать свою точку зрения с использованием языковых средств.

Совместная деятельность:

понимать и использовать преимущества командной и индивидуальной работы;

выбирать тематику и методы совместных действий с учетом общих интересов и возможностей каждого члена коллектива;

принимать цели совместной деятельности, организовывать и координировать действия по их достижению: составлять план действий, распределять роли с учетом мнений участников, обсуждать результаты совместной работы;

оценивать качество своего вклада и каждого участника команды в общий результат по разработанным критериям;

предлагать новые проекты, оценивать идеи с позиции новизны, оригинальности, практической значимости;

осуществлять позитивное стратегическое поведение в различных ситуациях, проявлять творчество и воображение, быть инициативным.

Регулятивные универсальные учебные действия

Самоорганизация:

самостоятельно осуществлять познавательную деятельность, выявлять проблемы, ставить и формулировать собственные задачи в образовательной деятельности и жизненных ситуациях;

самостоятельно составлять план решения проблемы с учетом имеющихся ресурсов, собственных возможностей и предпочтений;

давать оценку новым ситуациям;

расширять рамки учебного предмета на основе личных предпочтений;

делать осознанный выбор, аргументировать его, брать ответственность за решение;

оценивать приобретенный опыт;

способствовать формированию и проявлению широкой эрудиции в разных областях знаний, постоянно повышать свой образовательный и культурный уровень.

Самоконтроль:

давать оценку новым ситуациям, вносить коррективы в деятельность, оценивать соответствие результатов целям;

владеть навыками познавательной рефлексии как осознания совершаемых действий и мыслительных процессов, их результатов и оснований, использовать приемы рефлексии для оценки ситуации, выбора верного решения;

оценивать риски и своевременно принимать решения по их снижению;

принимать мотивы и аргументы других при анализе результатов деятельности.

Принятия себя и других:

- принимать себя, понимая свои недостатки и достоинства;
- принимать мотивы и аргументы других при анализе результатов деятельности;
- признавать свое право и право других на ошибку;
- развивать способность понимать мир с позиции другого человека.

ПРЕДМЕТНЫЕ РЕЗУЛЬТАТЫ

В процессе изучения курса название углубленного уровня в **10 классе** обучающимися будут достигнуты следующие предметные результаты:

1. Объяснять сущность понятий «вайбкодинг», «ИИ-агент», «промт», «инференс», «контекстное окно», различать виды агентов (реактивные, с памятью, инструментальные).
2. Составлять эффективные промты для генерации программного кода (Python) с использованием БЯМ, добиваясь синтаксически верного и логически корректного кода.
3. Разрабатывать простые консольные ИИ-агенты, взаимодействующие с пользователем через текстовые диалоги и выполняющие заданные функции (калькулятор, переводчик, простой помощник по финансам).
4. Использовать публичные API БЯМ для отправки запросов и обработки ответов в своей программе на Python (через requests или SDK).
5. Отлаживать промты и агентные скрипты, анализировать ошибки генерации («галлюцинации»), вносить коррективы в формулировки запросов.
6. Соблюдать этические и правовые нормы при использовании ИИ (указание на генерацию, запрет на создание вредоносного ПО, сохранение приватности данных).

В процессе изучения курса название углубленного уровня в **11 классе** обучающимися будут достигнуты следующие предметные результаты:

1. Проектировать архитектуру ИИ-агента с поддержкой контекстной памяти (краткосрочной/долгосрочной), использованием внешних инструментов (поиск в интернете, работа с базой данных, выполнение вычислений).
2. Разрабатывать агентов с использованием фреймворков типа LangChain или аналогичных, интегрировать несколько БЯМ и цепочек промтов.
3. Создавать агентов, способных выполнять последовательные действия (планирование шагов, вызов функций, анализ результатов).
4. Проводить сравнительный анализ производительности агентов при разных параметрах (температура, top-p, системный промт) и выбирать оптимальные настройки.
5. Реализовывать простые многоагентные системы (два взаимодействующих агента, например, агент-исполнитель и агент-критик).
6. Разрабатывать простой веб-интерфейс (или интерфейс в Telegram-боте) для взаимодействия с ИИ-агентом, развёртывать его на учебном хостинге.
7. Оформлять проектную документацию на агента: цель, архитектура, промты,

примеры диалогов, оценка эффективности.

ТЕМАТИЧЕСКОЕ ПЛАНИРОВАНИЕ

Обозначения уровня сложности:

Б — базовый

П — продвинутый

В — высокий

Ф — повторение / финализация

10 КЛАСС

№ п/п	Наименование разделов и тем учебного предмета	Количество часов	Программное содержание	Основные виды деятельности обучающихся	Уровень
Раздел 1. Промпт-инжиниринг и основы работы с БЯМ					
1.1	Введение в вайбкодинг и ИИ-агентов	4	Понятие вайбкодинга как программирования через диалог с генеративной нейросетью. Роль человека в цикле: постановка задачи, проверка и доводка результатов. Примеры использования БЯМ для генерации кода, текста, идей. Обзор современных БЯМ (ChatGPT, DeepSeek, Qwen, YandexGPT).	Приводить примеры задач, которые можно решить с помощью БЯМ. Обсуждать преимущества и риски вайбкодинга. Формулировать простые запросы к ИИ в учебных целях.	Б
1.2	Как работают БЯМ: токены, контекст, температура	4	Принципы работы больших языковых моделей на концептуальном уровне: токенизация текста, предсказание следующего токена. Понятие контекстного окна. Температура генерации: влияние на случайность и креативность ответов. Различие между инференсом и дообучением.	Объяснять базовые принципы работы БЯМ своими словами. Экспериментировать с разными значениями температуры в демо-режиме. Сравнивать результаты генерации при разных параметрах.	Б
1.3	Промпт-инжиниринг: техники и шаблоны	6	Промпт как форма постановки задачи. Типы промптов: zero-shot, few-shot,	Составлять промпты разных типов для решения учебных задач. Сравнивать	Б

			Chain-of-Thought, промпты с назначением роли. Техники уточнения: итеративное уточнение, добавление ограничений, указание формата вывода. Примеры эффективных и неэффективных промптов для генерации кода на Python.	качество ответов на разных промптах. Анализировать ошибки генерации («галлюцинации») и корректировать промпты.	
1.4	Инструменты вайбкодинга	2	Обзор инструментов: ChatGPT, DeepSeek, GitHub Copilot, Cursor, Codeium. Возможности и ограничения каждого. Правила безопасного использования: запрет на передачу персональных данных, проверка сгенерированного кода на вредоносные действия. Этические аспекты.	Зарегистрироваться и ознакомиться с интерфейсом одного-двух инструментов. Сравнить ответы разных БЯМ на один промпт. Формулировать правила безопасной работы с ИИ.	Б
1.5	Контрольная работа (творческий зачёт) по модулю 1	2	Проверка понимания основ промпт-инжиниринга и устройства БЯМ. Форма: конкурс «Самый короткий промпт, решающий задачу» или тест-квест.	Демонстрировать умение составлять промпты для генерации кода и объяснения понятий. Оценивать свою работу по чек-листу.	Б
Итого по разделу		18			
Раздел 2. Программное взаимодействие с БЯМ через API					
2.1	Основы HTTP и REST API. Получение API-ключа	2	Основы HTTP: запрос, ответ, методы (GET, POST). REST API. Аутентификация с помощью API-ключа. Получение и безопасное хранение ключа (переменные окружения, файл .env).	Объяснять структуру HTTP-запроса. Получать API-ключ в одном из публичных сервисов (OpenRouter, DeepSeek API). Создавать переменную окружения для ключа.	П
2.2	Первый скрипт: отправка запроса к БЯМ на Python	4	Библиотека requests. Формирование POST-запроса: URL, заголовки (Authorization, Content-Type), тело запроса (параметры model, messages, temperature, max_tokens). Обработка	Писать скрипт, отправляющий запрос к БЯМ и выводящий ответ. Изменять параметры (температуру, модель) и наблюдать за изменениями. Обрабатывать простые ошибки (неверный ключ).	П

			ответа: извлечение текста из JSON.		
2.3	Управление контекстом: история диалога	4	Хранение истории сообщений в списке. Ограничение длины контекста. Отправка полной истории при каждом запросе. Простейший консольный чат-бот с памятью.	Реализовывать чат-бота, который помнит последние 3–5 сообщений. Анализировать, как история влияет на ответы.	П
2.4	Function calling / инструменты: описание и вызов	4	Механизм вызова функций (инструментов). Описание доступных функций в запросе (имя, параметры, описание). Обработка ответа БЯМ с требованием вызова функции. Выполнение функции на стороне клиента и возврат результата. Примеры: калькулятор, получение текущей даты.	Создавать агента, который умеет складывать числа через вызов функции add. Писать описание функции в формате JSON. Отлаживать неверные вызовы.	П
2.5	Практическая работа: интеграция с внешним API (курсы валют)	4	Использование вызова функций для получения данных из внешнего API (например, курсы валют). Объединение запроса к БЯМ, вызова функции, HTTP-запроса к внешнему сервису и формирования ответа пользователю.	Разрабатывать агента-конвертера валют: пользователь вводит сумму и валюты, агент через БЯМ определяет нужную функцию, выполняет запрос к открытому API курсов и возвращает результат.	П
2.6	Контрольная работа (кодинг) по модулю 2	2	Практическое задание: написать скрипт, который отправляет запрос к БЯМ с функцией вызова и обрабатывает ответ. Формат: починка «сломанного» кода или реализация по описанию.	Демонстрировать умение работать с API, вызовом функций, обрабатывать ошибки. Оформлять код с комментариями.	П
Итого по разделу		20			
Раздел 3. Разработка простых ИИ-агентов					
3.1	Что такое ИИ-агент: свойства и классификация	2	Понятие ИИ-агента: автономность, реактивность, проактивность, способность использовать инструменты. Классификация: простые рефлекторные агенты, агенты с памятью,	Приводить примеры агентов из реальной жизни (боты-помощники, рекомендательные системы). Сравнить свойства агентов.	П

			целеориентированные, с полезностью. Отличие агента от чат-бота.		
3.2	Проектирование агента для конкретной задачи	4	Архитектура агента на основе БЯМ: цикл «восприятие — планирование — действие — наблюдение». Проектирование системного промпта, описывающего поведение агента. Определение необходимых инструментов (функций).	В группах выбирать задачу (помощник по математике, планировщик задач, финансовый советник). Проектировать промпт и набор функций.	В
3.3	Реализация агента-помощника (финансы / математика)	6	Разработка агента с вызовом функций для вычислений и объяснения формул. Пример: агент, который рассчитывает сложный процент, переводит валюты, даёт определения финансовых терминов.	Писать код агента на Python. Тестировать на разных запросах. Исправлять ошибки в промптах и вызовах функций.	В
3.4	Агент-модератор текста	4	Создание агента, который проверяет текст на наличие запрещённых тем (нецензурная лексика, угрозы, персональные данные). Промпт с примерами. Интеграция с функцией, которая возвращает вердикт.	Разрабатывать агента-модератора. Тестировать на наборе фраз. Анализировать ложные срабатывания и корректировать промпт.	В
3.5	Агент с краткосрочной памятью	4	Реализация памяти на основе переменных: сохранение последних действий, фактов о пользователе. Использование контекстного окна для хранения истории. Ограничения и способы сжатия истории.	Создавать агента, который запоминает имя пользователя, его предпочтения (например, любимую валюту) и использует это в диалоге.	В
3.6	Промежуточная аттестация (мини-проект)	8	Выполнение мини-проекта в парах: создание простого ИИ-агента по выбору (телеграм-бот, консольный помощник, агент для генерации SQL-запросов и т. п.). Требования: наличие вызова функций, памяти, документации.	Самостоятельно планировать работу, распределять роли, реализовывать агента, тестировать, оформлять код и краткое описание.	В
3.7	Защита проектов и рефлексия	2	Презентация проекта (2–3 минуты): демонстрация работы агента,	Представлять результаты, отвечать на вопросы, оценивать свою работу и работу	В

			объяснение архитектуры, трудностей и решений. Самооценка по чек-листу.	товарищей. Формулировать планы по улучшению.	
Итого по разделу		30			
ОБЩЕЕ КОЛИЧЕСТВО ЧАСОВ ПО ПРОГРАММЕ		68			

11 КЛАСС

№ п/п	Наименование разделов и тем учебного предмета	Количество часов	Программное содержание	Основные виды деятельности обучающихся	Уровень
Раздел 4. Продвинутая архитектура ИИ-агентов					
4.1	Контекстная память: буфер, суммаризация, RAG	6	Буфер диалога, скользящее окно. Суммаризация истории: использование БЯМ для сжатия. Понятие векторной базы данных и RAG (Retrieval-Augmented Generation) на концептуальном уровне: эмбединги, поиск релевантных фрагментов. Простейшая реализация RAG с готовыми библиотеками (Chroma, FAISS).	Реализовывать агента с суммаризацией истории. Создавать простой RAG- агент, который отвечает на вопросы по загруженному документу (учебник, статья).	П
4.2	Фреймворк LangChain: Chains, Tools, Agents	8	Знакомство с LangChain (или аналогичным фреймворком). Компоненты: LLMChain, SequentialChain, простые инструменты (Wikipedia, Calculator). Создание агента с выбором инструментов на основе запроса.	Устанавливать LangChain, создавать простую цепочку. Разрабатывать агента, который умеет искать в Wikipedia и выполнять вычисления.	П
4.3	Режимы планирования: ReAct, Plan-and-Execute	4	ReAct (Reasoning + Acting): чередование рассуждения и действий. Plan-and-Execute: отдельное планирование, затем выполнение. Сравнение эффективности на задачах с несколькими шагами (например, «найди курс доллара, умножь на 100 и переведи в евро»).	Программировать агента с явным циклом рассуждения. Анализировать логи агента. Сравнить время и точность разных режимов.	П

4.4	Оценка качества работы агента	2	Метрики: точность выполнения шагов, полнота достижения цели, время ответа, субъективная полезность. Инструменты трассировки (LangSmith, журналирование).	Разрабатывать чек-лист для оценки агента. Проводить тестирование и записывать результаты.	П
4.5	Контрольная работа по модулю 4	2	Тест + кейс: спроектировать архитектуру агента с памятью и RAG для заданной предметной области (например, помощник для подготовки к ЕГЭ по истории).	Демонстрировать понимание принципов RAG, ReAct, фреймворков.	П
Итого по разделу		22			
Раздел 5. Многоагентные системы и взаимодействие					
5.1	Понятие многоагентной системы (MAS). Роли агентов	4	Кооперация и координация агентов. Роли: координатор, исполнитель, критик, супервайзер. Преимущества и сложности MAS. Примеры: финансовый консультант с несколькими специализированными агентами.	Приводить примеры MAS из реальности (беспилотные автомобили, системы управления). Анализировать роли в учебном кейсе.	В
5.2	Создание пары «исполнитель — критик»	6	Реализация двух агентов: один генерирует ответ или код, другой анализирует результат, указывает на ошибки и предлагает улучшения. Зацикливание: ограничение числа итераций.	Парами программировать агентов-исполнителя и критика. Тестировать на задаче генерации кода для сортировки массива. Улучшать ответы через итерации.	В
5.3	Агент-маршрутизатор	4	Анализ пользовательского запроса, выбор наиболее подходящего специализированного агента. Пример: маршрутизатор направляет финансовые вопросы к финансовому агенту, математические — к математическому.	Реализовывать маршрутизатора на основе промпта или простого классификатора. Интегрировать с двумя-тремя агентами.	В

5.4	Контрольная работа: проектирование MAS	2	Кейс: спроектировать многоагентную систему для заказа пиццы (агент-приёмщик, агент-кухня, агент-доставка). Описать роли, промпты, взаимодействие.	Представлять схему MAS, обосновывать выбор ролей.	В
5.5	Резерв / практикум по отладке MAS	2	Разбор типичных проблем: бесконечные циклы, противоречивые инструкции, дублирование работы. Методы синхронизации.	Отлаживать собственную MAS, использовать журналирование.	В
Итого по разделу		18			
Раздел 6. Развёртывание и проектная деятельность					
6.1	Веб-интерфейс для агента (Streamlit / Gradio)	6	Создание веб-приложения для агента с использованием Streamlit или Gradio. Поле ввода, вывод истории диалога, кнопки для вызова действий. Запуск локально.	Разрабатывать интерфейс для своего агента. Подключать агента к интерфейсу. Тестировать в браузере.	В
6.2	Telegram-бот как обёртка агента	6	Регистрация Telegram-бота через @BotFather. Библиотека python-telegram-bot или Aiogram. Обработка команд, текстовых сообщений. Интеграция с агентом.	Создавать Telegram-бота, который пересылает запросы агенту и возвращает ответ. Размещать бота на локальной машине для тестов.	В
6.3	Развёртывание на сервере (Render / PythonAnywhere)	4	Платформы для бесплатного хостинга (Render, PythonAnywhere, Replit). Настройка переменных окружения. Автоматический перезапуск. Ограничения бесплатных тарифов.	Развёртывать веб-приложение или Telegram-бота на удалённом сервере. Проверять работоспособность через интернет.	В
6.4	Безопасность и этика при развёртывании	2	Фильтрация входящих промптов (санитайзер), ограничение прав агента (запрет на системные команды). Защита от промпт-инъекций. Правовые аспекты: ответственность за действия агента,	Формулировать правила безопасной эксплуатации агента. Добавлять в код проверки и предупреждения.	В

			информирование пользователя.		
6.5	Итоговый групповой проект	8	Разработка полноценного ИИ-агента для решения прикладной задачи (финансовый планировщик, помощник по ДЗ, аналитик новостей, агент для планирования мероприятий). Этапы: анализ, проектирование, реализация, тестирование, документация, презентация.	Работать в группах, распределять задачи, писать код, готовить демонстрацию и слайды. Соблюдать требования к проекту (память, инструменты, интерфейс).	В
6.6	Защита итоговых проектов	2	Презентация проекта (5 минут + вопросы). Демонстрация работы агента. Обсуждение трудностей, результатов, перспектив. Самооценка и взаимооценка.	Представлять проект, отвечать на вопросы, оценивать вклад участников.	В
Итого по разделу		28			
ОБЩЕЕ КОЛИЧЕСТВО ЧАСОВ ПО ПРОГРАММЕ		68			