Олимпиада школьников «Шаг в будущее» по профилю «Инженерное дело».

Регистрационный номер 56116.

Секция: Информационные технологии.

**Прикладной проект «Создание онлайн-платформы для дистанционной подготовки к экзаменам по математике».**

Выполнил: Чугайнов Арсений Денисович,

ученик 10А класса МБОУ Гимназии №19 им. Н.З.

Поповичевой г. Липецка.

Научный руководитель: Ложков Кирилл Германович,

педагог дополнительного образования,

место работы: ГБУ ДО ЦДО ЛО.

2026

**Аннотация**

Математика – это очень сложная, но важная наука. В школе она является основным и обязательным предметом для изучения. Для поступления в ВУЗ необходимо сдать ЕГЭ по базовой или профильной математике, для поступления на СПО необходимо сдать ОГЭ по математике. Одним словом, без математики в наши дни - никуда. Для большинства школьников её изучение составляет огромную проблему. Многим даже недостаточно школьных уроков для того, чтобы полностью понять материал, поэтому они нанимают репетиторов, но не всегда удается найти хорошего преподавателя, который жил бы в шаговой доступности от дома. Те, кто занимается очно, тратят очень много времени и денег на дорогу до своих репетиторов.

Моя работа направлена на решение проблемы недостатка доступных и эффективных инструментов для осуществления дистанционного обучения математике с преподавателем. В ходе проекта разработана веб-платформа, которая позволяет ученикам решать математические задания на различные темы с автоматической проверкой, представленные на сайте, а также домашние задания, которые в виде файлов прикрепляют им учителя; а преподавателям - контролировать и анализировать учебный процесс, проверять и оценивать решения, которые им в ответ отправляют ученики. Система может генерировать задачи в реальном времени, прилагать к ним подробные решения и начислять за них баллы, которые отображаются в таблице успеваемости группы (это создает эффект соревнования между учениками).

Платформа реализована с использованием технологий Python, Flask, SQLite, SQL Alchemy, HTML и CSS.

Результатом работы является полностью функционирующая образовательная платформа, готовая к внедрению в учебный процесс, имеющая потенциал на дальнейшее развитие.

**Содержание**

1. Введение
2. Цели и задачи
3. Обзор существующих решений
4. Технологический стек и обоснование выбора технологий
5. Архитектура системы
6. Реализация основных модулей
   1. Модуль аутентификации и управления пользователями
   2. Модуль генерации и проверки встроенных математических задач
   3. Модуль домашних заданий
   4. Модуль системы мониторинга решений
   5. Интерфейс пользователя
7. Преимущества и практическая значимость
8. Заключение и дальнейшие перспективы
9. Список использованных источников
10. Словарь терминов и сокращений

**Введение**

Современное образование активно переходит в цифровую среду, что особенно актуально в условиях дистанционного обучения. Подготовка к экзаменам по математике требует регулярной практики решения задач и контроля со стороны преподавателя. Существующие образовательные платформы часто ограничены в функционале, не предоставляют подробных разборов решений или не обеспечивают постоянную связь между учениками и учителями.

Актуальность данной работы обусловлена необходимостью создания специализированного инструмента, который объединяет в себе положительные стороны своих аналогов воедино. Предлагаемая платформа позволяет ученикам решать, как встроенные на сайте математические задачи с автоматической генерацией и проверкой, так и дополнительные задания от преподавателя; получать мгновенную обратную связь и видеть свой прогресс относительно одногруппников. Учителя получают возможность отслеживать успеваемость и активность каждого ученика, создавать домашние задания и анализировать историю решений группы.

**Цель и задачи**

**Цель проекта:** Создание веб-платформы для осуществления дистанционного обучения математике под контролем преподавателя с возможностями регистрации с выбором роли, генерацией и проверкой задач, управлением группами, системой баллов, технологией домашних заданий и аналитикой учителей через историю решений и таблицу успеваемости.

**Задачи проекта:**

1. Проанализировать существующие аналоги и определить их недостатки.
2. Разработать архитектуру системы с разделением ролей.
3. Реализовать модуль генерации математических задач различных типов.
4. Создать систему автоматической проверки ответов и начисления баллов.
5. Разработать модуль управления группами с возможностью приглашения участников.
6. Реализовать систему домашних заданий с загрузкой файлов.
7. Создать интерфейс аналитики для учителей с фильтрацией решений.
8. Сформулировать преимущества и практическую значимость платформы.

**Обзор существующих решений**

Ниже приведены примеры образовательных платформ, а также их плюсы и недостатки.

1. **Яндекс Репетитор.** Включает бесплатные краткие учебные курсы для учащихся старшей школы, помогает освоить основные школьные дисциплины и подготовиться к итоговым экзаменам. Плюсы: бесплатное обучение, удобный формат дистанционных занятий, хорошая подача материала. Недостатки: отсутствие прямого контакта с преподавателем, и как следствие, необходимость высокой самоорганизации.
2. **Сайт ФИПИ.** Содержит открытый банк заданий ЕГЭ, открытые варианты КИМ, демоверсии, спецификации, кодификаторы. Плюсы: помогает получить практику решения задач, так как они там представлены в огромных количествах на все возможные темы. Недостатки: на платформе нет абсолютно никакой системы взаимодействия учеников и преподавателей, ресурс не подразумевает подготовки к экзаменам под контролем и поддержкой учителя, изучение материала исключительно самостоятельное, требует высокой самоорганизации.
3. **Платформа «Учи.ру».** Образовательная платформа для школьников с 1 по 11 класс. Плюсы: Система анализирует результаты и показывает, по какой теме у учеников возникают сложности, дает дополнительные задания по этим темам. Минусы: Доступ не является полностью бесплатным (в день дается 20 бесплатных попыток на решение задач, не по всем предметам), для того чтобы снять лимиты и ограничения родитель должен зарегистрироваться на платформе и купить своему ребенку не дешевую платную подписку.

Как итог стоит отметить, что у существующих образовательных ресурсов есть свои проблемы. В своем проекте я объединил отдельные плюсы этих платформ и решил их недостатки.

**Технологический стек**

Изображение выглядит как текст, снимок экрана, логотип, Бренд

Содержимое, созданное искусственным интеллектом, может быть неверным.

Для технической реализации платформы выбран следующий стек.

1. Python – высокоуровневый язык программирования с богатыми набором различных библиотек, инструментов и фреймворков.
2. Flask – легковесный веб-фреймворк, позволяющий быстро разрабатывать функциональные веб-приложения. Выбран за свою простоту и гибкость.
3. SQLite – встраиваемая БД, не требующая отдельного сервера, идеальна для прототипирования и небольших проектов.
4. SQL Alchemy – ORM для работы с БД на уровне объектов Python в ООП. Выбран, так как позволяет удобно работать с данными.
5. WTF-Forms – библиотека для создания и валидации форм в Flask.
6. HTML/CSS – для создания пользовательского интерфейса.
7. Встроенные модули Python.
8. Среда разработки – IDE PyCharm 2025.
9. Контроль версий – GIT hub.

**Архитектура системы**

Система построена по классической трехзвенной архитектуре, называемой MVC.

M – model. Модель – это основная логика приложения, отвечает за данные, методы работы с ними и структуру программы. Она служит связующим звеном между представлением и контроллером.

V – view. Представление – визуализация информации, которая поступает от модели и контроллера. Он отображает данные на уровне пользовательского интерфейса.

C – controller. Контроллер обеспечивает взаимодействие с системой: обрабатывает запросы пользователя, проверяет полученную информацию и передает ее модели.

Данная архитектура идеально подходит для образовательного веб-приложения среднего масштаба, балансируя между простотой разработки, производительностью и возможностями расширения.

Далее представлена визуализация и статистика архитектуры моего проекта.

Изображение выглядит как текст, снимок экрана, Самоклеющийся листок, диаграмма

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Процесс создания**

**Модуль 1. Модуль аутентификации и управления пользователями.**

Как и на большинстве сайтов, первым делом пользователь попадает на лэндинг-страницу, на которой представлены все преимущества платформы и ее предлагаемые сервисы, а также есть возможность войти в учетную запись или создать ее.

При регистрации пользователь заполняет форму: ФИО, email, пароль; а также отмечает или пропускает галочку («Вы учитель?»). Если нажать эту галочку, то пользователь получит роль Учитель на платформе, если пропустить – роль Ученик.

В зависимости от выбранной роли пользователь получает доступ к разным возможностям на платформе.

**Учитель:**

1. Создает группы
2. Приглашает учеников по уникальной ссылке (модуль uuid)
3. Прикрепляет и проверяет домашние задания
4. Анализирует успеваемость при помощи таблицы и истории решений группы

**Ученик:**

1. Вступает в группы
2. Решает встроенный задания и получает за них баллы
3. Прикрепляет в виде файлов свои решения
4. Видит свои результаты в таблице успеваемости, сравнивает их с результатами своих одногруппников.

**Далее в научной работе эти сервисы будут разобраны.**

Вся информация о пользователях (users) и группах (groups) хранится в 2 таблицах БД. Для того чтобы каждый пользователь мог состоять в нескольких группах сразу, существует третья таблица (group\_members), в которой при помощи внешних ключей устанавливается связь между id пользователя и id группы.

Ниже показана визуализация параметров, свойств и взаимодействий этих 3 таблиц.

Изображение выглядит как текст, диаграмма, снимок экрана, число

Содержимое, созданное искусственным интеллектом, может быть неверным.

PK – первичный ключ

FK – внешний ключ

NOT\_NULL– первичный ключ

UQ – уникальный

Работа с БД реализована при помощи библиотеки SQL Alchemy и ORM-моделей. Они позволяют работать с данными как с объектами Python, это очень удобно и сильно упрощает данную задачу.

**На реализацию модуля ушло порядка 2 недель.**

**Модуль 2. Модуль генерации и проверки встроенных математических задач.**

Все технологии, реализующие встроенные задания, находятся в отдельном скрипте maths. Данный скрипт составляет класс MyMath, его методы отвечают за генерацию, решение и проверку всех типов заданий. При инициализации сервер создает экземпляр этого класса и дальше использует его методы до тех пор, пока сайт не закроется. Это более оптимально, чем использовать набор отдельных функций, так как в таком случае приходилось бы каждый раз заново инициализировать каждую вызванную функцию.

Логика, отвечающая за генерацию разборов заданий, реализована в скрипте configs.

**Теперь рассмотрим все моменты поподробнее.**

1. **Комплекс заданий**

На сайте представлен набор заданий по математике, предназначенных для учеников как младшей, так и старшей школы.

**Задания:**

1. Линейные уравнения
2. Квадратные уравнения
3. Биквадратные уравнения
4. Иррациональные уравнения
5. Уравнения с модулем
6. Линейные неравенства на метод интервалов
7. Примеры на арифметические операции (3 уровней сложности)

Данный комплекс задач дает умение решать уравнения разных типов, объясняет непростые математические тонкости и принципы решения на простых примерах, предоставляет фундаментальные знания школьной алгебры.

1. **Логика математических заданий**

Архитектура класса математических задач построена таким образом, что каждый тип задания имеет единую структуру методов, которая отвечает за его реализацию, это: метод генерации, решения, проверки, выявления коэффициентов, индивидуальный набор вспомогательных функций. Поэтому будем разбирать все технологии на примере одного типа задания – Квадратное уравнение.

**Метод генерации.**

Для генерации каждого типа уравнения в системе есть заранее заложенный шаблон (см изображение ниже).

Изображение выглядит как текст, снимок экрана, Шрифт, алгебра

Содержимое, созданное искусственным интеллектом, может быть неверным.

Разберем данную технологию. Квадратное уравнение - ax² + bx + c = 0. Генерируются случайные целые числа при помощи библиотеки random, но они могут быть отрицательными или равными 1, если в чистом виде их подставить в шаблон, то получится что-то на подобии: 1x² + -3x + 5 = 0. Эта запись некорректна, так как не принято писать минус после плюса и единицу перед x. Поэтому каждый коэффициент проходит этап форматирования, где он сравнивается с 0 и 1, после чего в зависимости от результатов сравнения встраивается в общую конструкцию уже корректно (1 – перед x не пишется, -1 – перед x остается только минус, к отрицательным числам прибавляется конструкция: <пробел>-<пробел>, к положительным: <пробел>+<пробел>). После такого форматирования получается корректная привычная запись: x² - 3x + 5 = 0, которую и видит пользователь на экране.

**Метод решения.**

Так как каждое задание имеет фиксированный шаблон, к каждому из них можно написать свой алгоритм решения.

1. Линейное уравнение – перенос коэффициентов с x в одну сторону, без – в другую.
2. Квадратное – дискриминант.
3. Биквадратное – замена x² и решение квадратного уравнения.
4. Иррациональное и с модулем – ограничение на правую часть и возведение обеих частей в квадрат
5. Линейное неравенство – метод интервалов.

Разберем данную технологию. Принцип решения квадратного уравнения через дискриминант:

Изображение выглядит как текст, Шрифт, рукописный текст, число

Содержимое, созданное искусственным интеллектом, может быть неверным.

A, b, c – коэффициенты уравнения

Чтобы воспользоваться одной из этих формул необходимо найти коэффициенты и тогда задача будет решена.

**Метод выявления коэффициентов.**

Шаблонность заданий позволяет также написать для каждого типа уравнения паттерн регулярных выражений и при помощи библиотеки re вытаскивать из уравнений коэффициенты и подставлять их в формулы и алгоритмы решений. Эти паттерны хранятся в словаре, находящемся в конструкторе математического класса.

**Метод проверки.**

После того как пользователь решил предложенное ему уравнение, ввел ответ в текстовое поле и нажал кнопку «Ответить», сервер вызывает метод проверки. Сначала система вызывает метод решения соответствующего задания, тот при помощи метода выявления коэффициентов решает переданное ему уравнение, возвращает правильный ответ, и затем метод проверки сравнивает его с ответом пользователя (примечание: т.к. ответы пользователя и системы сравниваются как строки, оформление ответа важно, на сайте представлена по этому поводу подробные инструкции). Далее в зависимости от правильности ответа, пользователю либо начисляется, либо не начисляется балл и предлагается проверить расчёты и попробовать снова.

1. **Разборы заданий**

На сайте также есть возможность посмотреть подробный разбор любого уравнения. Стоит отметить, что разборы, представленные на сайте, для некоторых типов заданий показывают нестандартные пути решения, которые позволяют решить их быстрее и с меньшим шансом допустить ошибку. Одним из таких является уравнение с модулем.

**Стандартный метод. Метод, представленный на сайте.**

Изображение выглядит как текст, Шрифт, рукописный текст, белый

Содержимое, созданное искусственным интеллектом, может быть неверным.Изображение выглядит как Шрифт, белый, дизайн, типография

Содержимое, созданное искусственным интеллектом, может быть неверным.

Еще один важный момент – как проходит сам процесс генерации. В системе заложены заранее продуманные шаблоны, по которым будет строится решение, для каждого типа задания отдельно. Когда на сервер поступает запрос на получение разбора задачи, программа при помощи словаря конфигураций подбирает подходящий шаблон и просто подставляет в него числа для конкретного уравнения.

1. **Возможные расширения**

В будущем можно будет расширить комплекс заданий, добавить контрольные тесты с таймером.

**На реализацию модулю ушло около 3 недель.**

**Модуль 3. Модуль домашних заданий.**

Комплекс заданий, представленный на сайте, охватывает достаточно узкий спектр тем и дает понимание лишь базовых фундаментальных вещей. Поэтому я решил добавить пользователям с ролью Учитель возможность прикреплять собственные задания в виде файлов (работа с файлами реализована при помощи модуля os), а Ученикам – возможность отправлять им в ответ свои решения этих заданий. Преподаватель может просматривать их, писать комментарии, оценивать баллами вручную. Данная технология, реализованная на моем платформе, дает возможность не просто самостоятельно получать базовые знания по предмету, а полноценно готовиться к любым экзаменам вместе со своим школьным учителем или репетитором, а также постоянно быть с ним на связи, не выходя из дома.

**Техническая реализация. Срок выполнения около 3 недель.**

Для того, чтобы реализовать такую технологию, были добавлены еще 2 таблицы в БД (homeworks, homework\_solutions). Через таблицы groups и users (о них ранее упоминалось) устанавливается связь каждого домашнего задания с группой, в которую оно было отправлено, и с учителем, который его отправил. Также к каждому пользовательскому заданию прикреплен список решений, который отправили ученики.

**Одной из интересных технологий является применение каскадного удаления:**

* homework\_solution\_files (файлы домашних заданий) удаляются при удалении homework\_solutions (списка решений)
* homework\_solutions (списки решений) удаляются при удалении homeworks (домашнего задания)

**Возможные расширения.**

* Добавление дедлайнов
* Уведомления учителю об отправленном решений от ученика, и ученику о выставленной оценке учителем

**Модуль 4. Модуль системы мониторинга решений.**

Встроенные задания проверяются системой автоматически, вернее сказать: система не проверяет решение, а только окончательный ответ. Этот подход не совсем корректен, ведь большинство заданий, например линейное неравенство или иррациональное уравнение – это задачи, требующие на экзаменах развернутого ответа и оформления. Поэтому я решил Ученикам добавить возможность прикреплять к заданиям файлы с решениями, а Учителям просматривать их в отдельной вкладке «История решений». Благодаря данной технологии учитель могут получать более полную информацию об активности своих учащихся. Преподаватель может искать решения при помощи умной системы поиска по следующим параметрам: Имя, Фамилия, Отчество, дата отправки (работа с датами реализована при помощи библиотеки datetime). Нет необходимости заполнять все данные, поисковая система будет выдавать список решений, основываясь на том, что введет пользователь.

**Техническая реализация. Срок выполнения около 2 недель.**

Для создания данной системы было добавлено еще 2 таблицы БД (solutions, solutions\_files) и еще больше зависимостей. Данная структура работает по похожему принципу, что и технология Домашних заданий. Список решений (solutions) привязан к группе и к пользователю, который его отправил, а к каждому решению прикреплены файлы (solution\_files).

**Здесь также реализовано каскадное удаление:**

* solution\_files (файлы с решениями) удаляются при удалении solutions (списка решений)

**Возможные расширения:**

* Уведомление учителю о новом отправленном решении.

На этом моменте дополнение БД было завершено, полная визуализация сигнатур таблиц и зависимостей между ними представлена ниже.

Изображение выглядит как текст, диаграмма, План, Технический чертеж

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Модуль 5. Интерфейс пользователя. Выполнялось в течении всего периода разработки.**

Платформа использует минималистичный и функциональный дизайн, ориентированный на удобство восприятия образовательного контента.

**Основные преимущества:**

* Чистота и простота – нет отвлекающих цветов и эффектов, фокус на содержании.
* Навигационная иерархия – интуитивно понятные пути для разных ролей.
* Подтверждение опасных действий – модальные окна при удалении пользователя из группы или очистки истории решений.
* Скрытие навигации при решении – чтобы не отвлекать ученика лишним интерфейсом.

Для отображения страниц написано более 20 HTML-шаблонов.

**Ключевые страницы:**

* Base.html – базовый каркас, подключение CSS
* Landing.html - главная
* Login.html, register.html - аутентификация
* Profile.html, edit\_profile.html - профиль
* Groups.html, group\_details.html – управление группами
* Task\_window.html, task\_opened.html – решение задач
* Group\_homeworks.html – список домашних заданий группы
* Group\_solutions.html – история решений группы

**Интересные технологии:**

* Анимации при наведении на кнопку

**Преимущества и практическая значимость**

**Преимущества платформы:**

* 1. Постоянная обратная связь с преподавателем
  2. Геймификация обучения – система баллов и таблица успеваемости
  3. Удобный мониторинг успеваемости – учитель отслеживает активность учеников через историю решений, таблицу и решения домашних заданий
  4. Подробные разборы заданий с нестандартными, более оптимальными путями решения, нежели стандартные методы.

**Практическая значимость:**

Платформа может быть внедрена в школах для дистанционной подготовки к ОГЭ и ЕГЭ по математике (и не только), а также для дополнительных занятий с репетиторами. Она сокращает время проверки шаблонный заданий учителем и концентрирует его внимание на дополнительных собственных заданиях. Создает между учениками соревновательный эффект, ведь каждый хочет стать обладателем самого высокого балла в группе, тем самым мотивирует их к продолжению обучения.

**Заключение и дальнейшие перспективы**

В ходе работы была успешно разработана и реализована онлайн-платформа для дистанционной подготовки к экзаменам по математике. Система включает все заявленные функции: регистрацию с ролями, генерацию и проверку задач, управление группами, систему баллов, технологию домашних заданий и аналитику учителей через историю решений и таблицу успеваемости.

**Дальнейшие перспективы развития:**

* 1. Расширение комплекса встроенных заданий (тригонометрия, логарифмы, задачи с параметрами, геометрия)
  2. Разработка мобильного приложения
  3. Интеграция с системами школьного электронного журнала
  4. Добавление возможности проведения онлайн-тестирований с таймером
  5. Внедрение AI-ассистентов для учителей

Итог: Платформа представляет собой готовое решение для внедрения в образовательный процесс и имеет потенциал для масштабирования и дальнейшего развития, а также для конкуренции с другими подобными проектами.

**Ссылка на публичный гит-репозиторий с проектом:** https://github.com/Manex0990/profmat.

**Список использованных источников**

1. Документация Flask – URL: <https://flask.palletsprojects.com/>
2. Документация SQL Alchemy – URL:  <https://www.sqlalchemy.org/>
3. Официальный сайт PyCharm – URL: <https://www.jetbrains.com/pycharm/>
4. Сайт ФИПИ – URL: <https://fipi.ru/>
5. Сайт Учи ру – URL: <https://uchi.ru/>
6. Сайт Яндекс Репетитор – URL: <https://education.yandex.ru/ege/go/math>
7. Miro Интерактивная доска – URL: <https://miro.com/app/dashboard/>
8. Официальный сайт Шаг в Будущее – URL: <https://olymp.bmstu.ru/>

**Приложение. Словарь терминов и сокращений.**

1. **БД** – база данных
2. **Фреймворк** – программная платформа, определяющая структуру программной системы и предоставляющая набор компонентов и инструментов для решения типовых задач разработки. Проще готовя, каркас проекта.
3. **ЕГЭ** – единый государственный экзамен
4. **ОГЭ** – основной государственный экзамен
5. **СПО** – среднее профессиональное образование
6. **КИМ** – контрольно–измерительные материалы, экзаменационные бланки с заданиями
7. **ООП** – объекто-ориентированное программирование.
8. **ORM** – технология программирования, которая обеспечивает взаимодействия между ООП и реляционными БД
9. **Лэндинг-страница** – (главная, гостевая) автономная веб-страница, созданная специально для ознакомления пользователя с платформой.
10. **Аутентификация** – процесс проверки подлинности пользователя или системы, направленный на подтверждение их права доступа к определенным ресурсам или данным.
11. **Репозиторий** – место, где хранятся некие цифровые данные.
12. **AI** – искусственный интеллект.
13. **Дедлайн** – срок выполнения задачи
14. **Id** – уникальный идентификационный номер некого объекта.

**Приложение. Демонстрация.**

**Лэндинг страница (фрагмент).**

Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Форма регистрации.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, линия

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Профиль.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Список групп (учитель).**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Таблица успеваемости глазами ученика.**

Изображение выглядит как программное обеспечение, снимок экрана, Операционная система, Мультимедийное программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Управление группой учителя.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Выбор встроенных заданий.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Операционная система

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Страница задания на примере Иррационального уравнения.**

**Изображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Подробный разбор на примере Квадратного уравнения.**

**Изображение выглядит как текст, снимок экрана

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Список домашних заданий глазами ученика**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Список домашних заданий глазами учителями**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**Список решений.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Содержимое, созданное искусственным интеллектом, может быть неверным.**

**История решений.**

Изображение выглядит как текст, программное обеспечение, Значок на компьютере, Операционная система

Содержимое, созданное искусственным интеллектом, может быть неверным.

**Поиск по решениям.**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Шрифт

Содержимое, созданное искусственным интеллектом, может быть неверным.**