# 1 Теоретический раздел

В данном разделе приведены сведения о решаемой задаче. Описываются функции онлайн-платформы. Производится обзор существующих платформ для решения алгоритмических задач.

## 1.1 Обзор предметной области

Онлайн-платформы для решения алгоритмических задач стали важным инструментом для разработчиков программного обеспечения, желающих улучшить свои навыки кодирования и алгоритмического мышления. Эти платформы позволяют пользователям решать задачи различной сложности, охватывающие различные аспекты программирования, включая сортировку, поиск, динамическое программирование, структуры данных и математические алгоритмы. Платформы предоставляют доступ к обширной библиотеке задач, часто разделенных по категориям и уровню сложности. Пользователи могут решать задачи на различных языках программирования, что позволяет им тренироваться и развивать навыки на выбранном языке.

Решение задач сопровождается мгновенной оценкой решений: результаты могут включать правильность ответа, эффективность решения и рекомендации по улучшению кода. Платформы также предлагают объяснения решений, советы по оптимизации, и обсуждения, где пользователи могут обмениваться идеями и лучшими практиками. Это создает сообщество разработчиков, поддерживающее рост и обучение. Некоторые платформы предлагают систему очков, рейтингов и достижений, стимулирующих пользователей к участию в решении все более сложных задач.

Таким образом, онлайн-платформы для решения алгоритмических задач представляют собой ценные ресурсы для тех, кто стремится улучшить свои навыки программирования, подготовиться к собеседованиям в крупных компаниях и оставаться в курсе новейших тенденций и методов в области разработки программного обеспечения.

## 1.2 Аналитический обзор механизма автоматизированной проверки решений

Автоматизированная проверка решений на онлайн-платформах для решения алгоритмических задач заключается в прогоне каждого отправленного решения по заранее установленному набору автоматизированных тестов. Эти тесты охватывают различные аспекты задачи, включая простые и сложные входные данные, чтобы убедиться, что решение работает корректно и эффективно во всех случаях. Авто-тесты также проверяют, чтобы решение не только выдавал правильный ответ, но и соблюдало ограничения по времени и памяти, заданные в условиях задачи.

Такой подход к проверке позволяет мгновенно и объективно оценить код пользователя, обеспечивая его точность и производительность. Прогон по авто-тестам также выявляет граничные случаи, которые могут быть неочевидными для разработчика, и помогает избежать ошибок, которые могли бы возникнуть в условиях реальной среды. Если решение пользователя не проходит один или несколько тестов, платформа возвращает соответствующее сообщение о сбое и, в некоторых случаях, предоставляет подсказки или указания о том, что именно пошло не так.

Автоматизированная проверка с прогоном по набору авто-тестов также дает возможность платформам обеспечивать конкурентную и справедливую среду для всех пользователей, гарантируя, что все решения проверяются по одинаковым критериям и стандартам. Это способствует развитию навыков программирования, улучшению качества кода и повышению уровня компетенции пользователей.

## 1.3 Аналитический обзор механизма изолированного запуска программ

Безопасность платформы является важной темой, поскольку запуск пользовательского кода небезопасен на хост-системе. Это создает потенциальные риски, включая возможность выполнения вредоносного кода и несанкционированного доступа к данным или ресурсам системы. Решением данной проблемы является применение контейнеров приложений. Docker — это платформа для контейнеризации, которая позволяет создавать, развертывать и управлять приложениями в изолированных средах, называемых контейнерами. Контейнер — это легковесный, автономный пакет, содержащий все необходимое для запуска приложения, включая код, библиотеки, зависимости и конфигурации.

Контейнеры предоставляют возможность развертывать приложения в согласованной и предсказуемой среде, независимо от того, где они запускаются — на локальной машине, в облаке или на других серверах. Возможности контейнеров включают изоляцию процессов, файловой системы и сетевых интерфейсов, что позволяет им работать в среде, отделенной от хост-системы и других контейнеров.

Использование контейнеров для изоляции автоматизированной проверки на онлайн-платформах алгоритмических задач может существенно улучшить безопасность и стабильность платформы. Контейнеры позволяют запускать каждое решение пользователя в отдельной изолированной среде, тем самым ограничивая возможные взаимодействия между кодом пользователя и основной инфраструктурой платформы. Это предотвращает потенциальные атаки, такие как доступ к системным файлам хост-системы или запуск вредоносного кода.

Контейнеры также позволяют контролировать ресурсы, такие как процессорное время и память, которые доступны для каждого контейнера. Это обеспечивает справедливые условия для всех пользователей и предотвращает злоупотребление ресурсами. Изолированная среда контейнера также позволяет платформе применять ограничения безопасности, такие как запрет на определенные системные вызовы, что дополнительно снижает риск вредоносных действий.

В целом, использование контейнеров для изоляции автоматизированной проверки на онлайн-платформах алгоритмических задач повышает безопасность и надежность платформы, обеспечивая при этом справедливую и предсказуемую среду для всех пользователей. Это способствует защите данных и ресурсов платформы, а также улучшает опыт работы пользователей.

## 1.4 Обзор существующих решений в области обучающих веб-сервисов с возможностью автоматической проверки решений задач

На сегодняшний день существует множество онлайн-платформ для решения алгоритмических задач и проверки знаний в области программирования. Одной из самых известных и популярных платформ на международном уровне является «Leetcode», а в российском сегменте — «Информатикс».

Эти две платформы значительно отличаются по своим подходам и целевым аудиториям. «Информатикс» сосредоточена на дистанционном обучении в области информатики и предлагает разнообразные курсы, задачи и тесты для школьников и студентов, а также для учителей. Эта платформа ориентирована на обучение и развитие студентов, позволяя преподавателям контролировать и отслеживать результаты учащихся.

В отличие от «Информатикс», «Leetcode» поддерживает большее количество языков программирования, что делает платформу более доступной для различных категорий программистов. В то же время сложность задач является более высокой, что увеличивает порог входа и уменьшает возможности решения слабо подготовленным студентам. Также отмечается формат задач: в частности, на платформе «Информатикс» менее обширное описание, нет статистики, задачи не делятся по тегам, напротив, в пользу данной платформы говорит наличие сопутствующих теоретических материалов в помощь студентам.

Важным преимуществом «Leetcode» является наличие встроенного редактора с подсветкой синтаксиса и возможностью тестирования программы на предоставленных данных, что облегчает процесс отладки и проверки кода. В отличие от этого, «Информатикс» предоставляет лишь кнопку для загрузки файла, что вынуждает пользователя писать и отлаживать код локально.

Также стоит упомянуть платформу «Codewars», которая предлагает задачи для программистов всех уровней, позволяя участникам повышать свои навыки посредством обучения через решение задач, созданных сообществом. Подобно «Leetcode», «Codewars» поддерживает различные языки программирования и позволяет пользователям сравнивать свои решения с решениями других участников, что способствует улучшению своих навыков. Более того, «Codewars» предоставляет возможность создавать свои собственные задачи, что привлекает более опытных программистов и дает возможность сообществу активно участвовать в создании новых задач для платформы.

С точки зрения пользовательского интерфейса, «Информатикс» не дотягивает до современных стандартов, так как его дизайн устарел, а главная страница переполнена информацией, затрудняющей навигацию. Поиск задач также затруднен, поскольку они обозначены номерами, которые часто известны только преподавателям. Интерфейс страницы задачи представлен на рисунке 1.1.

В отличие от него, «Leetcode» предлагает современный дизайн и тщательно продуманный список задач с различными возможностями сортировки, тегирования и поиска по темам. Это упрощает пользователям поиск необходимых задач и улучшает общее взаимодействие с платформой. Интерфейс страницы задачи представлен на рисунке 1.2.



Рисунок 1.1 – Платформа «Информатикс»



Рисунок 1.2 – Платформа «LeetCode»

«Codewars» также отличается удобным пользовательским интерфейсом, ориентированным на задачи для программистов всех уровней. Платформа предоставляет возможность поиска задач по разным уровням сложности, языкам программирования и темам. Кроме того, интерфейс предоставляет возможность взаимодействия с сообществом путем обсуждения задач и решений, что повышает качество обучения и способствует обмену знаниями между участниками. Интерфейс страницы задачи представлен на рисунке 1.3.

Значительное преимущество перед другими платформами имеет «LeetCode», благодаря наличию онлайн-компилятора (playground), который позволяет выполнять пользовательские программы вне решения задач. Этот инструмент предоставляет разработчикам гибкость и свободу для экспериментов с кодом в интерактивной среде, что повышает качество обучения и развитие навыков.

Онлайн-компилятор позволяет пользователям проверять, тестировать и оптимизировать свои алгоритмы и решения без давления или ограничений конкретной задачи. Это дает возможность разработчикам работать над собственными проектами или идеями, применять новые концепции и подходы к программированию, а также исследовать различные языки программирования.

Интерактивный характер онлайн-компилятора позволяет пользователям сразу видеть результаты выполнения своего кода, что ускоряет процесс обучения и отладки. Это также позволяет пользователям проверять эффективность и производительность своего кода, сравнивая различные подходы к решению задач.

Кроме того, наличие онлайн-компилятора помогает пользователям развивать навыки программирования в более широком контексте, чем просто выполнение конкретных задач. Это стимулирует творчество и инновации, позволяя пользователям экспериментировать и исследовать новые способы решения проблем.

Еще одной интересной возможностью является совместная работа над решением задачи в реальном времени. Это позволяет пользователям обмениваться идеями, исправлять ошибки друг друга и коллективно находить наилучшие решения.

При использовании функции совместного редактирования пользователи могут одновременно просматривать, редактировать и анализировать код друг друга. Это способствует обмену опытом и знаниями, а также повышает эффективность работы в команде.

Преимущества совместного редактирования кода включают возможность мгновенного обратного связи, ускорение процесса разработки и улучшение качества кода за счет совместного обсуждения и корректировки. Кроме того, это помогает развивать навыки командной работы и обучаться новым подходам к решению задач.

На данный момент ни одна из рассматриваемых платформ не имеет возможности совместного редактирования в реальном времени.