Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУ ВО «АмГУ»)

Институт компьютерных и инженерных наук
Кафедра информационных и управляющих систем
Направление подготовки 09.03.02. – Информационные системы и технологии
Направленность (профиль) образовательной программы Информационные системы и технологии

ДОПУСТИТЬ К ЗАЩИТЕ
Зав. кафедрой
_____ А.В. Бушманов
«____» _____ 2025 г.

БАКАЛАВРСКАЯ РАБОТА

на тему: Разработка программной визуализации криптографических методов перестановки и замены

Выполнил
студент группы 1104-об _____ (подпись, дата) В.Н. Гудимов

Руководитель
доцент, канд. техн. наук _____ (подпись, дата) С.Г. Самохвалова

Консультант
по безопасности и
экологичности
доцент, канд. техн. наук _____ (подпись, дата) А.Б. Булгаков

Нормоконтроль
Инженер кафедры _____ (подпись, дата) В.Н. Адаменко

Благовещенск 2025

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АМУРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
(ФГБОУВО «АмГУ»)

Институт компьютерных и инженерных наук
Кафедра информационных и управляющих систем
Направление подготовки 09.03.02 Информационные системы и технологии
Направленность (профиль) образовательной программы Информационные системы и технологии

УТВЕРЖДАЮ
Зав. кафедрой
_____ А.В. Бушманов
«____» _____ 2025 г.

ЗАДАНИЕ

К выпускной квалификационной работе студента Гудимова В.Н.

1. Тема выпускной квалификационной работы: Разработка программной визуализации криптографических методов перестановки и замены (утверждено Приказом от 14.04.2025 №980-уч)
2. Срок сдачи студентом законченной работы (проекта): 10.06.2025
3. Содержание выпускной квалификационной работы: анализ предметной области; освоение программного и технического обеспечения; разработка алгоритма решения; применение результата на практике
4. Перечень материалов приложения: перечень вопросов из опроса, демонстрация работы программной визуализации, программная визуализация на онлайн платформе Яндекс.Игры.
5. Дата выдачи задания: 02.10.2024

Руководитель выпускной квалификационной работы: Самохвалова С.Г., доцент, канд.техн.наук.

Задание принял к исполнению: 02.10.2024 г._____

(подпись)

РЕФЕРАТ

Бакалаврская работа содержит 97 с., 48 рисунков, 2 формулы, 2 таблицы, 32 источника, 3 приложения.

C#, UNITY, BLENDER, 3D, КРИПТОГРАФИЯ, ВИЗУАЛИЗАЦИЯ, МАГИЧЕСКИЙ КВАДРАТ, ШИФР СКИТАЛА, ШИФР ЦЕЗАРЯ, ШИФР ВИЖИНЕРА, МЕТОДЫ ПЕРЕСТАНОВКИ, МЕТОДЫ ЗАМЕНЫ

Объектом исследования бакалаврской работы является криптография, в частности криптографические методы перестановки и замены. Предметом исследования является визуализация криптографических методов, с акцентом на разработку в программных решениях.

Целью бакалаврской работы является разработка программной визуализации криптографических методов перестановки и замены. Данный программный модуль должен быть предназначен для обучения пользователей основам криптографии в игровой форме.

Чтобы достичь поставленной цели, необходимо выполнить следующие задачи:

– изучить объект исследования;

– изучить предмет исследования;

– провести обзор существующих методов решения аналогичных типовых задач;

– описать разрабатываемый программный продукт;

– выбор программных средств реализации и обоснование их выбора;

– проектирование и разработка программной визуализации;

– выбор методов распространения.

СОДЕРЖАНИЕ

Введение 8
1 Обоснование актуальности темы 10
2 Характеристика объекта исследования 13
2.1 Объект исследования 13
2.1.1 Криптография 13
2.1.2 Методы перестановки 15
2.1.3 Методы замены 28
2.2 Предмет исследования 35
2.2.1 Способы визуализации 35
3 Обзор существующих методов решения аналогичных типовых задач 37
4 Постановка задачи и выбор инструментов для ее реализации 42
4.1 Проектирование программного продукта 42
4.1.1 Формулировка задачи 42
4.1.2 Выбор архитектуры 42
4.1.3 Функциональная структура 44
4.2 Методики решения 46
4.3 Существующие программные средства для решения задачи 47
4.3.1 Программное обеспечение для работы с 3D и анимацией 47
4.3.2 Среда разработки 49
4.4 Структура программы в Unity 52
5 Проектирование и разработка 3D-моделей 54
5.1 Выбор визуального стиля 54
5.2 Разработка 3D-моделей 55
5.2.1 Сбор вспомогательных изображений 56
5.2.2 Работа с объектами 57
5.2.3 Текстурирование 59
5.2.4 Создание локации 61

6 Разработка программной визуализаци 63
6.1 Проектирование структуры программной визуализации 63
6.2 Создание программы в среде разработки 64
6.2.1 Создание проекта 64
6.2.2 Создание визуализаций 68
7 Размещение программы на сервисах 73
7.1 Обзор существующих сервисов 73
7.2 Размещение на платформе Яндекс.Игры 76
8 Безопасность и экологичность 79
8.1 Безопасность 80
8.1.1 Опасные и вредные факторы на рабочем месте 80
8.1.2 Организация рабочего места 81
8.1.4 Освещение 84
8.1.5 Шум 85
8.1.6 Микроклимат 85
8.1.7 Графический интерфейс приложения 87
8.2 Экологичность 88
8.3 Чрезвычайные ситуации 90
8.3.1 Аварийные ситуации 90
8.3.2 Меры пожарной безопасности на рабочих местах 91
Заключение 93
Библиографический список 94
Приложение А Перечень вопросов из опроса 98
Приложение Б Демонстрация работы программной визуализации 104
Приложение В Программная визуализация на онлайн платформе Яндекс.Игры 109

НОРМАТИВНЫЕ ССЫЛКИ

В настоящей выпускной квалификационной работе использовались ссылки на следующие стандарты и нормативные документы:

СТО СМК 4.2.3.21-2018 – Оформление выпускных квалификационных и курсовых работ (проектов);

ГОСТ 34.601-90 Автоматизированные системы. Стадии создания;

ГОСТ 19.201-78 – Техническое задание;

ГОСТ 19.402-78 – Описание программ;

ГОСТ 19.701-90 – Схемы алгоритмов, программ, данных и систем;

ГОСТ 34.003-90 – Основные компоненты автоматизированных систем;

ГОСТ 19.101-77 – Единая система программной документации. Виды программ и программных документов;

ГОСТ 51188-98 – Испытания программных средств на наличие компьютерных вирусов.

ОПРЕДЕЛЕНИЯ, ОБОЗНАЧЕНИЯ, СОКРАЩЕНИЯ

ПО – программное обеспечение;

3D – (3-dimensional) три измерения;

3DS Max – 3D Studio Max;

UE – Unreal Engine;

GDScript – Godot Docs Script;

Lowly – низкополигональный;

X-Ray – рентгеновский луч;

UV Editing – UV-развёртка;

MVC – (Model-View-Controller) Модель-Представление-Контроллер;

UI – (User Interface) пользовательский интерфейс;

FBX – FilmBox, расширения имени файла;

DDoS – (Denial-Of-Service Attack) отказ в обслуживании;

API – (Application Programming Interface) интерфейс программирования приложения;

WebGL – Web Graphics Library, программная библиотека для языка JavaScript.

ВВЕДЕНИЕ

Криптография представляет собой науку, изучающую методы защиты информации путем ее преобразования в недоступный для несанкционированного доступа вид. В условиях стремительного развития цифровых технологий и глобальной интеграции информационных систем криптография становится краеугольным камнем обеспечения безопасности данных. Современный мир, характеризующийся резким увеличением объемов информации, требует надежных механизмов защиты, что обусловлено как ростом объема передаваемых и хранимых данных, так и возрастанием угроз со стороны киберпреступников. В условиях цифровизации, когда информация является одним из самых ценных ресурсов, необходимость в эффективных методах защиты становится особенно актуальной.

Таким образом, криптография не просто обеспечивает защиту информации, но и способствует формированию доверия в цифровом пространстве. В условиях постоянного роста угроз безопасности данных возникает необходимость в более глубоком понимании криптографических методов защиты. Это позволит не только повысить уровень осведомленности о существующих подходах к защите информации, но и создать инструменты для их анализа и изучения.

Криптографические методы перестановки представляют собой технику, используемую для защиты информации путем изменения порядка элементов в сообщении. Основная идея заключается в том, чтобы сделать текст трудным для восприятия, перемешивая его символы в соответствии с определенным правилом или ключом. Одним из известных методов является магический квадрат, который использует квадратную матрицу для организации символов и их перестановки. Простой шифр перестановки, в свою очередь, предполагает использование фиксированной последовательности для изменения порядка символов. Шифрующие таблицы также применяются для создания сложных преобразований текста, обеспечивая дополнительный уровень безопасности. Скитала,

древнегреческий метод, представляет собой цилиндр, на который наматывалась лента с текстом; при снятии ленты текст становился неразборчивым, и его можно было прочитать только при использовании цилиндра того же диаметра.

Методы замены, в отличие от перестановки, сосредоточены на изменении самих символов сообщения. Шифр Цезаря является одним из самых известных примеров, где каждый символ сдвигается на фиксированное количество позиций в алфавите, что создает простую, но эффективную защиту. Шифр Вижинера представляет собой более сложный метод, использующий ключевое слово для динамической замены символов, что значительно усложняет анализ шифра. Масонский шифр, также известный как шифр замены, использует различные символы и знаки для представления букв и слов, создавая уникальную систему кодирования, которая была популярна среди масонов.

Визуализация в разработке программ – это процесс представления данных, информации или процессов в графической форме, что позволяет пользователям легче воспринимать и анализировать информацию. Основная цель визуализации заключается в упрощении понимания сложных данных, выявлении закономерностей и тенденций, а также в улучшении взаимодействия с пользователем. Этот процесс может применяться на различных этапах разработки программного обеспечения, включая проектирование, тестирование и эксплуатацию.

Существует множество видов визуализации, включая прототипы интерфейсов и Wireframes, которые помогают разработчикам тестировать и демонстрировать концепции пользовательского интерфейса. Интерактивная визуализация, такая как дашборды, позволяет пользователям исследовать данные в реальном времени с помощью фильтров и инструментов взаимодействия.

1 ОБОСНОВАНИЕ АКТУАЛЬНОСТИ ТЕМЫ

Криптография, как область знаний и практик, на протяжении многих веков играла ключевую роль в обеспечении безопасности информации и коммуникаций. Однако, в последние десятилетия наблюдается явный тренд к забвению этой важной дисциплины. Основной причиной такого явления можно считать стремительное развитие технологий, которое изменило подход к защите данных и сделало шифрование более доступным и обыденным.

С каждым годом технологии становятся все более интегрированными в повседневную жизнь. Современные устройства, такие как смартфоны и компьютеры, оснащены встроенными средствами шифрования, которые работают в фоновом режиме. Пользователи часто не осознают, что их данные защищены сложными алгоритмами, и воспринимают это как должное. Например, большинство мессенджеров сегодня используют сквозное шифрование, обеспечивая безопасность переписки. Это создает у пользователей иллюзию полной защиты их информации и приводит к тому, что они перестают интересоваться основами криптографии и ее историей.

Поколение, выросшее в условиях повсеместного распространения цифровых технологий, воспринимает их работу как должное. Это приводит к недостаточному пониманию того, насколько интересными и сложными являются механизмы, лежащие в основе криптографических систем. Люди доверяют свои данные третьим лицам, полагаясь на их обещания о безопасности, что также снижает интерес к самостоятельному изучению криптографических методов. В результате, многие поколения людей, начиная с миллениалов и заканчивая поколением Z, могут не иметь представления о том, что такое криптография, как она работает и почему она важна.

Для подтверждения теории был проведён опрос среди студентов среднего профессионального образования. Опрос состоял из 11 вопросов, благодаря которым было выяснено насколько студенты знакомы с шифрованием и криптографией.

По результатам опроса выяснилось, что 45,3 процента опрошенных «Примерно представляют» что такое шифрование, а 3,1 процента опрошенных ответили «Нет, вообще не знаю», рисунок 1. Было также выяснено насколько студенты заинтересованы в самостоятельном изучении шифрования. Оказалось, что только 20 процентов интересуются шифрованием в полной мере. Также в опросе необходимо было выбрать в каких областях применяется шифрование информации. Правильно на этот вопрос ответили 43 процента опрашиваемых, ими были выбраны все области. Перечень вопросов представлен в Приложении А.

Рисунок 1 – Ответы опрашиваемых

В последние годы наблюдается значительный сдвиг в области криптографии, связанный с переходом от традиционных методов шифрования к квантовым технологиям. Классические алгоритмы, использующие математические методы, проигрывают квантовым вычислениям.

В связи с этим, и основываясь на результатах опроса, возникает необходимость в повышении интереса к криптографии, особенно среди молодежи для формирования нового поколения специалистов. Принято решение разработать программную визуализацию криптографических методов. Программа призвана не только продемонстрировать процесс шифрования, но и погрузить пользователей в увлекательный мир истории криптографии.

Таким образом, целью бакалаврской работы является разработка программной визуализации криптографических методов перестановки и замены.

Данный программный модуль должен быть предназначен для обучения пользователей основам криптографии в игровой форме. Важным аспектом должна являться возможность практического применения изучаемых методов, а также погружение в исторический контекст их появления.

Чтобы достичь поставленной цели, необходимо выполнить следующие задачи:

– изучить объект исследования. Необходимо изучить криптографические методы перестановки и замены, это включает в себя изучение их исторического контекста, алгоритмов, принципов работы и применяемых подходов;

– изучить предмет исследования. Следует рассмотреть различные методы визуализации криптографических процессов;

– провести обзор существующих методов решения аналогичных типовых задач. Необходимо исследовать существующие программные продукты и образовательные ресурсы, исследовать их сильные и слабые стороны;

– описать разрабатываемый программный продукт. После изучения предметной области и аналогов, важно четко сформулировать функциональные требования к программному модулю;

– выбор программных средств реализации и обоснование их выбора. Необходимо определить, какие инструменты будут использоваться для разработки;

– проектирование и разработка программной визуализации. На этом этапе следует создать архитектуру программного продукта, разработать пользовательский интерфейс и реализовать саму визуализацию;

– выбор методов распространения. Необходимо определить стратегии распространения разработанного программного продукта. Это может включать создание веб-сайта или использование сторонних ресурсов.

2 ХАРАКТЕРИСТИКА ОБЪЕКТА ИССЛЕДОВАНИЯ

2.1 Объект исследования

2.1.1 Криптография

Криптография представляет собой научную дисциплину, занимающуюся методами защиты информации посредством её преобразования в недоступный для понимания вид. Основной целью криптографии является обеспечение конфиденциальности, целостности и аутентичности данных, что достигается за счет применения различных алгоритмов и протоколов, позволяющих скрыть содержание передаваемой информации от несанкционированных лиц. Криптография охватывает широкий спектр методов, включая симметричное и асимметричное шифрование, цифровые подписи, а также хэширование, что делает её важным инструментом в области информационной безопасности.

Шифрование, в свою очередь, является одним из ключевых процессов в рамках криптографии. Оно представляет собой метод преобразования исходных данных в шифротекст с использованием определенного алгоритма и ключа. Шифрование служит основным средством реализации криптографических принципов, обеспечивая защиту информации от несанкционированного доступа. Несмотря на то, что шифрование и криптография часто используются как синонимы, они не являются идентичными понятиями. Шифрование можно рассматривать как одну из составляющих более широкой области криптографии, которая включает в себя также методы аутентификации и проверки целостности данных.

Ключ в контексте криптографии представляет собой секретную информацию, используемую для шифрования и расшифрования данных. Ключи играют центральную роль в обеспечении безопасности криптографических систем, поскольку именно они определяют, как данные будут преобразованы в шифротекст и обратно. В зависимости от типа используемой криптографической схемы ключи могут иметь различные характеристики и назначения.

Существует несколько основных типов ключей:

– симметричные ключи: В симметричной криптографии один и тот же

ключ используется как для шифрования, так и для расшифрования данных. Это означает, что обе стороны, участвующие в обмене информацией, должны иметь доступ к одному и тому же секретному ключу;

– асимметричные ключи: В асимметричной криптографии используются пара ключей – открытый и закрытый. Открытый ключ может быть распространен среди всех пользователей, в то время как закрытый ключ хранится в секрете. Данные, зашифрованные открытым ключом, могут быть расшифрованы только соответствующим закрытым ключом и наоборот;

– сессионные ключи: Эти ключи создаются для временного использования в рамках одной сессии или транзакции. Они обеспечивают дополнительный уровень безопасности, так как после завершения сессии ключи уничтожаются.

Криптостойкость – это характеристика криптографической системы, определяющая её способность противостоять различным атакам и сохранять безопасность данных в условиях потенциальных угроз. Криптостойкость зависит от множества факторов, включая сложность используемых алгоритмов, длину ключа и качество генерации случайных чисел.

Криптостойкость можно оценивать по нескольким критериям:

– сопротивляемость к атакам: Криптографическая система должна быть устойчива к различным типам атак, таким как атаки грубой силы, криптоанализ и атаки на основе статистического анализа;

– длина ключа: Чем длиннее ключ, тем сложнее его подобрать с помощью методов перебора. Например, для современных стандартов симметричного шифрования рекомендуется использовать ключи длиной не менее 128 бит;

– обновление алгоритмов: С течением времени криптографические алгоритмы могут устаревать из-за появления новых методов атаки. Поэтому важно регулярно обновлять используемые алгоритмы и ключи для поддержания криптостойкости.

Шифры – это алгоритмы, используемые для преобразования информации с целью защиты её содержания. Шифрование позволяет скрыть данные от несанкционированного доступа, превращая открытый текст (исходные данные) в

шифротекст (зашифрованные данные), который можно расшифровать только с помощью соответствующего ключа. Шифры играют ключевую роль в криптографии и используются в различных приложениях, таких как безопасная передача данных, хранение конфиденциальной информации и аутентификация.

Криптографические методы – это набор техник и алгоритмов, применяемых для шифрования и расшифрования данных. Эти методы обеспечивают защиту информации от несанкционированного доступа и позволяют гарантировать её целостность и подлинность. Криптографические методы могут быть разделены на несколько категорий, включая симметричные и асимметричные методы, а также хеширование и цифровые подписи.

Среди криптографических методов можно выделить два основных подхода:

– методы перестановки: В этих методах изменяется порядок символов в открытом тексте без изменения самих символов. Перестановка затрудняет анализ текста, так как структура исходного сообщения становится неочевидной;

– методы замены: В этих методах каждый символ открытого текста заменяется другим символом или группой символов. Замена также усложняет анализ текста, так как символы теряют свою исходную идентичность.

2.1.2 Методы перестановки

Криптографические методы перестановки представляют собой класс шифрования, в рамках которого символы открытого текста перераспределяются в соответствии с заранее установленным правилом. Основная идея заключается в том, что при помощи различных операций над символами исходного сообщения достигается его скрытие от неавторизованных пользователей. Перестановка не изменяет сами символы, а лишь изменяет их порядок, что делает данный метод относительно простым в реализации, но при этом требует наличия ключа для обратного преобразования.

Существуют несколько известных методов перестановки, среди которых:

– шифр простой перестановки;

– магический квадрат;

– шифр Скитала;

– шифрующие таблицы.

Рассмотрим каждый из методов перестановки более подробно.

Шифр простой перестановки представляет собой метод симметричного шифрования, в котором символы исходного сообщения переставляются в соответствии с заранее определённым ключом.

Процесс шифрования с использованием шифра простой перестановки начинается с выбора ключа, который определяет, как именно будут переставлены символы в сообщении. Ключ может быть представлен в виде последовательности чисел, где каждое число указывает на новую позицию соответствующего символа. Например, для сообщения «HELLO» и ключа «3 1 4 2 5» символы будут переставлены следующим образом: символ на первой позиции переместится на третью, символ на второй позиции займет вторую позицию, символ на третьей позиции переместится на четвёртую и так далее. В результате шифрование приведет к новому сообщению «ELHLO».

Дешифрование осуществляется путём применения обратного процесса, при котором используется тот же ключ для восстановления исходного порядка символов. Для этого необходимо определить обратную последовательность, которая указывает, куда должен быть перемещён каждый символ зашифрованного сообщения. В случае примера с шифром «ELHLO» и тем же ключом «3 1 4 2 5», процесс дешифрования будет включать перемещение символов обратно в их исходные позиции, что приведёт к восстановлению первоначального текста «HELLO».

Шифр простой перестановки представляет собой один из наиболее базовых методов шифрования, обладающий как преимуществами, так и недостатками.

Его простота реализации является одним из основных достоинств, что делает данный шифр доступным для понимания и применения даже для людей, не обладающих глубокими знаниями в области криптографии. Процесс шифрования и дешифрования с использованием шифра простой перестановки

характеризуется относительно высокой скоростью, что позволяет эффективно обрабатывать данные без значительных временных затрат. Кроме того, данный метод не требует сложных вычислений, что делает его привлекательным для использования в условиях ограниченных вычислительных ресурсов. Гибкость шифра также заслуживает внимания, так как возможность использования различных ключей для шифрования позволяет изменять порядок символов в зависимости от конкретных потребностей пользователя.

Тем не менее, шифр простой перестановки имеет и ряд значительных недостатков. Одним из наиболее серьезных является его уязвимость к частотному анализу. Поскольку данный шифр сохраняет символы неизменными, злоумышленники могут анализировать частоту появления символов в зашифрованном тексте, что значительно упрощает восстановление исходного сообщения. Также стоит отметить зависимость данного метода от ключа: без надежного управления ключами, например, в случае его утечки или предсказуемости, безопасность шифра существенно снижается. Легкость в угадывании ключа является еще одним фактором, способствующим уязвимости шифра. Если злоумышленник имеет доступ к нескольким зашифрованным сообщениям, он может применить методы перебора для определения ключа, что также подрывает безопасность системы.

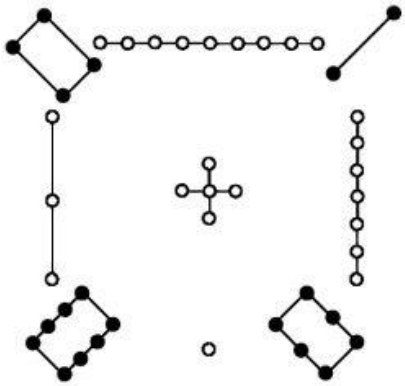
Магический квадрат представляет собой квадратную таблицу, в которой числа располагаются таким образом, что сумма чисел в каждой строке, каждом столбце и обеих диагоналях равна одной и той же величине, известной как магическая константа. Магическая константа M для квадрата размером n × n может быть вычислена с использованием формулы:

$$M = \frac{n(n^2 + 1)}{2}, \quad (1)$$

где n обозначает порядок магического квадрата.

История магических квадратов насчитывает множество веков и охватывает различные культуры. Первоначально магический квадрат третьего порядка был известен древним китайцам под названием Ло шу, как показано на рисунке 2.

Согласно преданию, он впервые появился на панцире священной черепахи, выбравшейся из реки Ло в XXIII веке до нашей эры. Однако современные китаеведы прослеживают его существование лишь до IV века до нашей эры. С этого времени и до X века данный магический квадрат служил мистическим символом значительного значения.



4	9	2
3	5	7
8	1	6

Рисунок 2 – Магический квадрат Ло Шу

Первое сохранившееся сочинение о магических квадратах было создано византийским грамматистом и лексикографом Мануэлем Москопулосом, и его работа датируется приблизительно 1300 годом. В этом труде Москопулос представил множество вычисленных им магических квадратов, обладающих различным числом клеток в основании.

В начале XVI века магический квадрат стал предметом художественного выражения. Знаменитый немецкий художник и гравер Альбрехт Дюрер в 1514 году создал гравюру под названием «Меланхолия», на заднем плане которой, над фигурой крылатой женщины в городской одежде, изображён магический квадрат размером 4 × 4 клетки. Интересно, что два средних числа нижней строки этого квадрата (15 и 14) складываются, образуя дату выпуска гравюры.

В Европе интерес к магическим квадратам возрос в эпоху Ренессанса, когда математики начали исследовать их свойства и возможности применения в таких областях, как алхимия и астрология.

Благодаря магическим квадратам также появилась популярная логическая игра Судоку. Изначально подобные головоломки начали появляться в 18 веке. В

1979 году японский журнал «Nikoli» начал публиковать судоку под названием «Судоку», что в переводе означает «один номер в каждой клетке». Формат, который известен сегодня, был разработан американским архитектором Ховардом Гарднером в 1979 году, когда он создал головоломку под названием «Number Place». Судоку быстро приобрело популярность в Японии и затем распространилось по всему миру, став одним из самых популярных логических игр.

Шифрование с использованием магического квадрата включает несколько этапов. Сначала необходимо построить магический квадрат соответствующего порядка. Например, для шифрования сообщения длиной девять символов можно использовать квадрат размером 3 × 3. Открытый текст помещается в квадрат по строкам или столбцам, при этом, если длина текста не кратна n, добавляются пробелы или специальные символы для заполнения.

После заполнения квадрата символы подвергаются перестановке в соответствии с заранее установленным порядком, например, по диагоналям или определённым маршрутам. Зашифрованный текст получается путём чтения символов из магического квадрата в заданном порядке.

Дешифрование осуществляется через восстановление магического квадрата. Зашифрованный текст помещается обратно в квадрат того же размера, после чего производится обратная перестановка символов, что позволяет восстановить исходное сообщение.

Уникальность метода шифрования с использованием магических квадратов заключается в том, что хотя количество магических квадратов ограничено, оно всё же является огромным. Это делает задачу подбора или создания конкретного магического квадрата достаточно сложной и трудоемкой. В связи с этим были разработаны различные методы построения магических квадратов, которые упрощают этот процесс.

Среди самых известных методов можно выделить:

– метод Франсуа Эдуара Лука: Этот метод основывается на систематическом заполнении квадратов, в основном метод предназначен для построения квадратов 3 × 3. Для этого необходимо лишь задать a, b, c и рассчитать значения

по формулам, рисунок 3;

c – b	c + (a + b)	c – a
c – (a – b)	c	c + (a – b)
c + a	c – (a + b)	c + b

Рисунок 3 – Метод Франсуа Эдуара Лука

– метод террас (метод Баше): Представляет собой подход для построения квадратов нечётной размерности. Этот метод предполагает добавление «террас» к исходному квадрату с четырёх сторон, что приводит к образованию зубчатого квадрата того же порядка. В полученной конфигурации числа располагаются в естественном порядке по диагональным рядам, начиная либо снизу вверх, либо сверху вниз. Числа, находящиеся в террасах и не вошедшие в квадрат, перемещаются внутрь фигуры, при этом они примыкают к противоположным сторонам квадрата, рисунок 4;

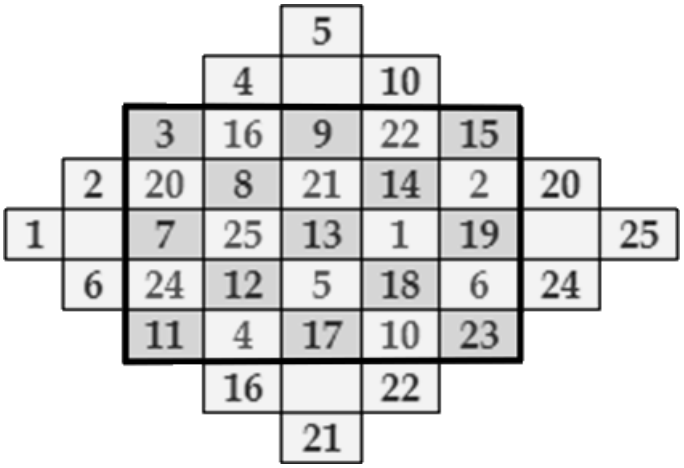


Рисунок 4 – Метод террас (метод Баше)

– индийский метод (сиамский метод): Метод построения для нечётных магических квадратов. Необходимо взять такую последовательность цифр чтобы разность между ними была одинакова, после чего поставить цифру в произвольном месте квадрата. Теперь необходимо двигаться по диагонали «вверх и в вправо». Если на месте уже есть цифра, необходимо двигаться от этой цифры «вниз на один квадрат». При построении также можно двигаться «вниз и вправо», рисунок 5;

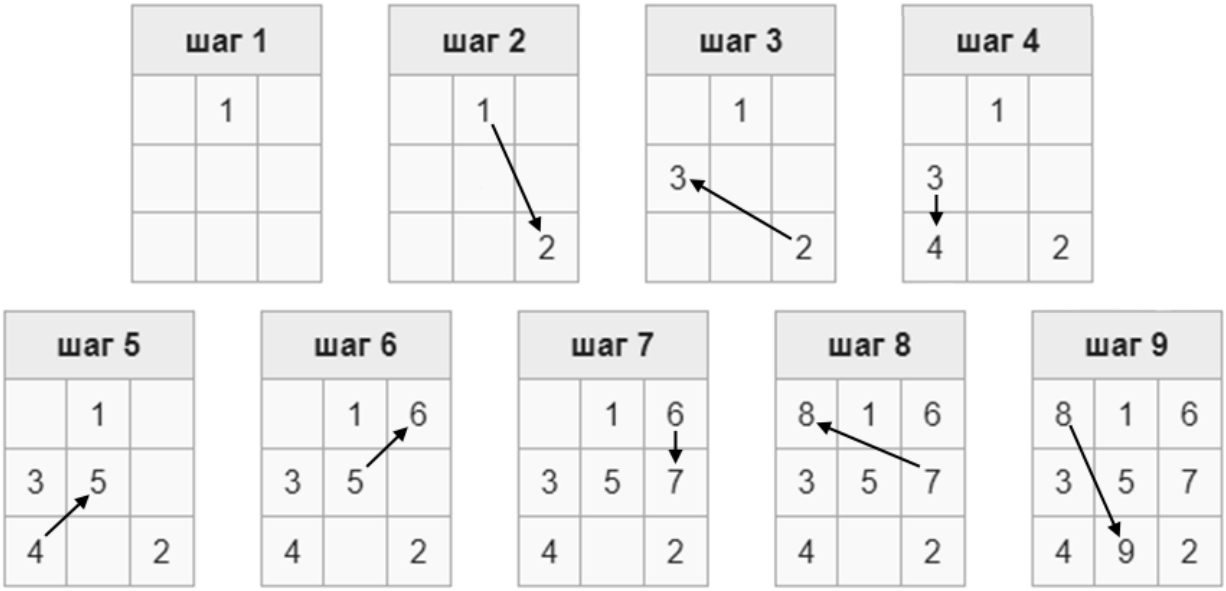


Рисунок 5 – Индийский метод (сиамский метод)

– метод Москопула (Ход конём): Этот метод использует движения шахматного коня для заполнения магического квадрата. Если ячейка, в которую должно быть вписано число, уже занята другим числом, то число вписывается в ячейку, расположенную в том же вертикальном ряду, что и ячейка с числом, но находящуюся на четыре ячейки выше;

– метод Альфила: Метод Альфила вполне аналогичен методу Москопула, только вместо хода коня в этом методе используется движение по диагонали через одну клетку.

Метод шифрования с использованием магических квадратов представляет собой интересный подход в области криптографии, обладающий как положительными, так и отрицательными аспектами.

Одним из основных преимуществ данного метода является уникальность каждого магического квадрата. Эта особенность затрудняет предсказание структуры шифра, что в свою очередь повышает уровень безопасности передаваемой информации. Простота реализации магических квадратов также является значительным достоинством. Основные принципы шифрования, основанные на использовании этих квадратов, достаточно понятны и могут быть легко реализованы даже людьми, не обладающими глубокими знаниями в области криптографии. Данная интуитивная доступность делает метод привлекательным для широкой аудитории, позволяя объяснить его основные концепции без необходимости погружаться в сложные математические теории.

Однако наряду с преимуществами метод шифрования магическими квадратами имеет и ряд недостатков. Одним из наиболее значительных является необходимость ручного создания квадратов, что требует значительных временных затрат и усилий, особенно при работе с большими размерами квадратов. Увеличение размера магического квадрата приводит к повышению сложности его построения, а также увеличивает вероятность ошибок в процессе создания. Кроме того, уязвимость к анализу является серьезной проблемой для данного метода. Если злоумышленник осведомлён о принципах построения магических квадратов, он может с легкостью разгадать шифр, применяя аналогичные техники. Ограниченная длина сообщения также представляет собой значительное ограничение, поскольку количество доступных ячеек в магическом квадрате фиксировано и не позволяет передавать длинные сообщения.

Шифр Скитала представляет собой один из древнейших известных методов шифрования, который использовался в античные времена для обеспечения конфиденциальности сообщений. В его основе лежит использование длинной ленты, на которую наносится сообщение, оборачиваемой вокруг цилиндра определенного диаметра. При этом текст становится неразборчивым, если лента будет снята с цилиндра, рисунок 6.

Рисунок 6 – Скитала

Скитала, известная также как «шифр древней Спарты», представляет собой одно из самых ранних криптографических устройств. Происхождение этого шифра восходит к Древней Греции, где спартанцы применяли его для защиты своих коммуникаций во время войны с Афинами в конце V века до н. э. Название «Скитала» происходит от древнегреческого слова сцитала, что переводится как жезл.

Предметом шифра Скитала является цилиндр, вокруг которого оборачивается лента с записанным сообщением. Цилиндр может быть изготовлен из различных материалов, таких как дерево или металл, и его диаметр определяет длину сообщения, которое может быть зашифровано. Ключом в контексте шифра Скитала выступает именно диаметр цилиндра. Зная размер цилиндра, получатель может легко расшифровать сообщение, в то время как злоумышленник без этой информации столкнется с трудностями при попытке восстановить исходный текст.

Принцип действия этого устройства был описан Аполлонием Родосским в середине III века до н. э. и Плутархом в период около 45–125 н. э., однако сохранилось только описание, оставленное последним.

Процесс шифрования осуществляется следующим образом: сначала сообщение записывается на ленте, которая оборачивается вокруг цилиндра. Затем лента снимается с цилиндра, и текст становится неразборчивым, поскольку буквы располагаются в произвольном порядке. Для дешифрования необходимо

выполнить обратную операцию: получатель должен знать диаметр цилиндра и обернуть ленту вокруг аналогичного цилиндра, что позволяет восстановить исходный порядок букв и прочитать сообщение.

Шифр Скитала обладает интересной особенностью: буквы исходного сообщения располагаются друг от друга на одинаковом расстоянии, обозначаемом как N. Это расстояние зависит от диаметра цилиндра, вокруг которого оборачивается лента с текстом. Чтобы рассчитать N, можно использовать следующую формулу:

$$N = \pi D, \quad (2)$$

где D – диаметр цилиндра.

Таким образом, зная высоту цилиндра, можно определить расстояние между буквами в зашифрованном тексте. Аналогично, если известен диаметр цилиндра и расстояние между буквами, можно вычислить длину сообщения.

Шифр Скитала представляет собой один из наиболее известных и простых методов шифрования, использующий физический объект, такой как цилиндр, для преобразования текста. Одним из ключевых преимуществ данного метода является его простота реализации. Шифр может быть осуществлён с использованием доступных материалов, таких как бумага и цилиндр, что делает его понятным и доступным для широкой аудитории. Эта простота позволяет быстро обучить пользователей основам шифрования, а также обеспечить возможность его применения в условиях ограниченных ресурсов.

Эффективность шифра также заслуживает внимания. Процесс шифрования и дешифрования сообщений осуществляется достаточно быстро при наличии цилиндра нужного диаметра. Это позволяет передавать информацию в сжатые сроки, что особенно важно в ситуациях, когда время имеет критическое значение. Для неосведомленного наблюдателя зашифрованный текст выглядит как бессмысленный набор символов, что обеспечивает относительную безопасность передачи информации. Отсутствие очевидных закономерностей в зашифрованном сообщении затрудняет его анализ и интерпретацию.

Тем не менее, шифр Скитала обладает рядом недостатков, существенно

ограничивающих его использование. Одной из ключевых уязвимостей является возможность подбора ключа. Аристотель считается автором метода взлома шифра Скиталы, при котором лента наматывается на конусообразную палку до тех пор, пока не становятся видимыми читаемые фрагменты текста. Это существенно снижает уровень безопасности шифра и делает его уязвимым для атак.

Кроме того, неэффективность шифра при больших объемах информации является серьёзным ограничением. С увеличением длины сообщения и количеством необходимых витков цилиндра процесс шифрования становится более сложным и трудоемким, что может негативно сказаться на скорости передачи информации.

С началом эпохи Возрождения, которая приходится на конец XIV века, наблюдается возрождение криптографии. В шифрах перестановки, разработанных в этот период, используются шифрующие таблицы, которые задают правила перестановки букв в сообщении. Данный метод шифрования позволяет избежать необходимости в физических объектах, таких как цилиндры, применяемые в традиционном шифре Скитала.

Эти таблицы являются двумерными структурами, где строки и столбцы содержат символы алфавита или другие элементы, используемые для шифрования. Основная идея заключается в том, что каждое сообщение преобразуется в набор координат, соответствующих расположению символов в таблице, что позволяет эффективно осуществлять как шифрование, так и дешифрование.

В качестве ключа в шифрующих таблицах используются:

– размер таблицы;

– слово, фраза или набор цифр, задающие перестановку;

– особенности структуры таблицы.

Шифрование с использованием шифрующих таблиц осуществляется следующим образом. Исходное сообщение разбивается на символы, которые затем сопоставляются с их положением в таблице. Например, если используется квадратная таблица размером $n \times n$, то каждый символ сообщения может быть представлен парой координат (i, j), где i – номер строки, а j – номер столбца. На

рисунке 7 представлена фраза «ТЕРМИНАТОР ПРИБЫВАЕТ СЕДЬМОГО В ПОЛНОЧЬ», записанная в шифрующую таблицу.

Т	**Н**	**П**	**В**	**Е**	**Г**	**Л**
Е	**А**	**Р**	**А**	**Д**	**О**	**Н**
Р	**Т**	**И**	**Е**	**Ь**	**В**	**О**
М	**О**	**Б**	**Т**	**М**	**П**	**Ч**
И	**Р**	**Ы**	**С**	**О**	**О**	**Ь**

Рисунок 7 – Шифрующая таблица

Дешифрование, в свою очередь, представляет собой обратный процесс. Полученное зашифрованное сообщение интерпретируется с использованием той же шифрующей таблицы. Сопоставление координат зашифрованных символов с их исходными позициями в таблице позволяет восстановить первоначальное сообщение. Важно отметить, что для успешного дешифрования необходимо точно знать структуру таблицы и порядок её использования, что подчеркивает важность хранения ключа в секрете.

Для повышения уровня безопасности шифрующих таблиц существует несколько методов, которые могут значительно усложнить процесс дешифрования для злоумышленников. Один из таких методов заключается в записи сообщения не только по горизонтали, но и по диагонали в разные стороны.

Кроме того, важным аспектом шифрования является использование одиночной перестановки по ключу. Этот метод заключается в том, что символы сообщения располагаются в таблице в определённом порядке, заданном ключом. Ключ представляет собой слово, которое указывает, в каком порядке строки или столбцы должны быть переставлены. На рисунке 8 изображён пример шифрования сообщения с использованием одиночной перестановки по ключу, в качестве ключевого слова используется «ПЕЛИКАН».

КЛЮЧ →

П	Е	Л	И	К	А	Н
7	2	5	3	4	1	6
Т	Н	П	В	Е	Г	Л
Е	А	Р	А	Д	О	Н
Р	Т	И	Е	Ь	В	О
М	О	Б	Т	М	П	Ч
И	Р	Ы	С	О	О	Ь

До перестановки

А	Е	И	К	Л	Н	П
1	2	3	4	5	6	7
Г	Н	В	Е	П	Л	Т
О	А	А	Д	Р	Н	Е
В	Т	Е	Ь	И	О	Р
П	О	Т	М	Б	Ч	М
О	Р	С	О	Ы	Ь	И

После перестановки

Рисунок 8 – Одиночная перестановка по ключу

Двойная перестановка – это ещё один мощный метод шифрования, который сочетает в себе два этапа перестановки. На первом этапе выполняется одиночная перестановка по ключу, как описано выше. Затем результат подвергается второй перестановке с использованием другого ключа. Эта двойная операция значительно увеличивает сложность шифрования и делает анализ зашифрованного текста ещё более затруднительным.

Объединение одиночной и двойной перестановки также может быть использовано для создания многоуровневых схем шифрования. Например, можно сначала применить одиночную перестановку для изменения порядка строк, а затем выполнить двойную перестановку на полученном результате.

Шифрующие таблицы обладают рядом преимуществ по сравнению с классическими методами шифрования. Во-первых, они обеспечивают большую гибкость в выборе алфавита и структуры шифрования, что позволяет адаптировать их под конкретные требования. Во-вторых, использование таблиц делает процесс шифрования более быстрым и удобным, так как не требуется физическая манипуляция с объектами. Однако следует учитывать и потенциальные уязвимости: если структура таблицы станет известна злоумышленнику, это может привести к компрометации конфиденциальности сообщений.

Методы перестановки использовались для защиты информации на протяжении многих веков. Однако с развитием науки о шифровании и появлением частотного анализа, который представляет собой метод изучения частоты

появления различных символов в зашифрованном тексте, эффективность таких методов значительно снизилась.

Частотный анализ основывается на статистическом исследовании языковых особенностей, позволяя криптоаналитикам выявлять закономерности в распределении букв и их комбинаций. Например, в большинстве языков определенные буквы встречаются с различной частотой, что делает возможным сопоставление зашифрованных символов с наиболее распространенными буквами языка.

Это открытие привело к тому, что методы перестановки стали менее эффективными, поскольку они не изменяли частотное распределение символов, а лишь изменяли их порядок. Также процесс перестановки букв в большом сообщении требует значительных временных затрат и усилий, что делает его менее практичным для использования в условиях, когда необходимо быстрое и надежное шифрование. В результате, на смену методам перестановки пришли более эффективные методы замены, которые обеспечивают более высокий уровень безопасности.

2.1.3 Методы замены

Криптографические методы замены представляют собой один из основных подходов к шифрованию информации, основанный на замене символов исходного текста на другие символы с целью обеспечения конфиденциальности передаваемых данных. Эти методы играют важную роль в обеспечении безопасности информации, так как они позволяют скрыть содержание сообщения от несанкционированного доступа. В отличие от методов перестановки, которые изменяют порядок букв, методы замены фокусируются на изменении самих символов, что делает их более устойчивыми к различным формам криптоанализа.

Одноалфавитные методы замены являются подкатегорией методов замены, при которых каждый символ исходного алфавита заменяется на другой символ из того же алфавита. В данном случае используется фиксированное соответствие между символами, что означает, что одна и та же буква всегда будет заменяться одной и той же буквой в процессе шифрования.

Несмотря на свою простоту, одноалфавитные методы замены имеют ряд

уязвимостей. Поскольку они сохраняют частотное распределение символов исходного текста, криптоаналитики могут применять частотный анализ для выявления закономерностей и расшифровки зашифрованного сообщения.

Среди самых известных одноалфавитных методов замены можно выделить следующие:

– масонский шифр. Метод, использующий различные символы и знаки для замены букв;

– шифр Цезаря. Простой метод с фиксированным сдвигом букв.

Рассмотрим подробнее каждый из них.

Масонский шифр – это метод шифрования, который использует набор символов или знаков для замены букв алфавита. Он часто ассоциируется с масонскими ритуалами и символикой, где каждое слово или буква могут быть заменены на уникальные символы, что делает его трудным для расшифровки без знания конкретного ключа.

Шифр был создан в XVIII веке. Вариации данного шифра применялись орденом розенкрейцеров и масонами, причем последние использовали его столь часто, что он стал известен как шифр масонов. Масоны начали использовать его в начале XVIII века с целью сохранения в тайне записей своей истории и обрядов, а также переписки между лидерами своего движения. На надгробиях масонов встречаются гравюры с надписями, выполненными с использованием этого шифра.

В масонском шифре каждая буква заменяется на определенный символ, который может быть как графическим, так и буквенным. Ключом в этом методе является таблица соответствий между буквами и символами. Основными элементами этой системы являются сетки и точки. Некоторые системы используют символ перекрещивания, но самой популярной является система Pigpen представленная на рисунке 9.

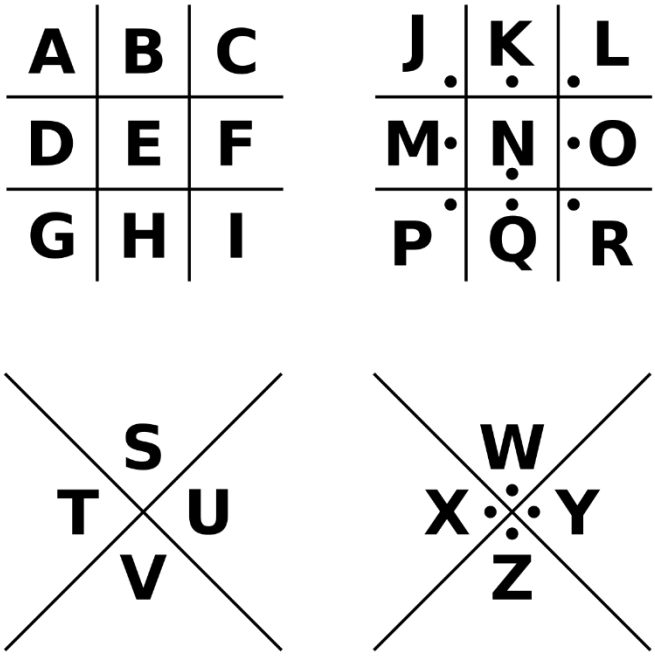


Рисунок 9 – Масонский шифр (pigpen)

Для расшифровки сообщения необходимо иметь ту же таблицу соответствий. Каждое вхождение символа в зашифрованном тексте заменяется на соответствующую букву по таблице.

При условии, что таблица замены известна исключительно отправителю и получателю, обладает высокой степенью защиты информации. Эта особенность обеспечивает значительную конфиденциальность передаваемых данных, так как даже в случае перехвата зашифрованного сообщения третьими лицами, без доступа к таблице расшифровка становится крайне затруднительной. Гибкость в выборе символов для замены также является важным преимуществом данного метода.

Однако масонский шифр не лишен недостатков. Одной из основных уязвимостей является возможность частотного анализа, который может быть использован, если таблица замены становится известной или может быть угадана. Кроме того, создание и использование таблицы для больших объемов текста представляет собой сложную задачу.

Одноалфавитный шифр замены, известный как шифр Цезаря, представляет

собой один из самых простых и древних методов шифрования, который использует систему замены букв в алфавите.

Шифр Цезаря называют в честь Юлия Цезаря, который, согласно «Жизни двенадцати цезарей» Светония, использовал его со сдвигом 3, чтобы защищать военные сообщения. Хотя Цезарь был первым зафиксированным человеком, использовавшим эту схему, другие шифры подстановки, как известно, использовались и ранее.

Принцип шифрования с использованием шифра Цезаря основан на простоте и регулярности замены. Заменяющая буква определялась путем смещения по алфавиту от исходной буквы на К букв. При достижении конца алфавита выполнялся циклический переход к его началу. Цезарь использовал шифр замены при смещении К = 3. Такой шифр замены можно задать таблицей подстановок, рисунок 10, содержащей соответствующие пары букв открытого текста и шифртекста.

Одноалфавитные подстановки (К = 3, m = 26).

A	→	**D**	**J**	→	**M**	**S**	→	**V**
B	→	**E**	**K**	→	**N**	**T**	→	**W**
C	→	**F**	**L**	→	**O**	**U**	→	**X**
D	→	**G**	**M**	→	**P**	**V**	→	**Y**
E	→	**H**	**N**	→	**Q**	**W**	→	**Z**
F	→	**I**	**O**	→	**R**	**X**	→	**A**
G	→	**J**	**P**	→	**S**	**Y**	→	**B**
H	→	**K**	**Q**	→	**T**	**Z**	→	**C**
I	→	**L**	**R**	→	**U**			

Рисунок 10 – Таблица подстановок

Так, при сдвиге на три буквы буква «A» становится «D», «B» превращается в «E», и так далее.

Например, послание Цезаря «VENI VIDI VICI» (в переводе на русский означает «Пришел, Увидел, Победил»), направленное его другу Аминтию после победы над понтийским царем Фарнаком, сыном Митридата, выглядело бы в зашифрованном виде так: «YHQL YLGL YLFL».

Таким образом, шифр Цезаря представляет собой классический пример симметричного шифрования, где для шифрования и дешифрования используется один и тот же ключ – величина сдвига. Дешифрование осуществляется обратным образом: необходимо выполнить сдвиг в противоположном направлении.

Несмотря на свою простоту, шифр Цезаря имеет определенные уязвимости. Основная проблема заключается в том, что всего существует ограниченное количество возможных сдвигов, что делает его уязвимым к атаке методом «перебора». Злоумышленник может легко перебрать все возможные варианты сдвига и расшифровать текст, так как количество сдвигов в английском алфавите равна 26.

Тем не менее, данный шифр служит важной основой для понимания более сложных методов шифрования, например для модифицированного шифра Цезаря.

Система шифрования Цезаря с ключевым словом является модификацией шифра Цезаря. Особенность этой системы – использование ключевого слова для смещения и изменения порядка символов в алфавите подстановки.

Для данного шифра необходимо выбрать некоторое число k, $0 \le k \le 25$ и слово или короткую фразу в качестве ключевого слова. Желательно, чтобы все буквы ключевого слова были различными. Пусть выбраны слово DIPLOMAT в качестве ключевого слова и число k = 5.

Ключевое слово записывается под буквами алфавита, начиная с буквы, числовой код которой совпадает с выбранным числом k, рисунок 11.

0	1	2	3	4	5					10					15					20					25
A	B	C	D		F	H	G	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
					D	I	P	L	O	M	A	T													

Рисунок 11 – Ключевое слово под буквами алфавита

Оставшиеся буквы алфавита подстановки записываются после ключевого слова в алфавитном порядке, рисунок 12.

0	1	2	3	4	5					10					15					20					25
A	B	C	D		F	H	G	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
V	W	X	Y	Z	D	I	P	L	O	M	A	T	B	C	E	F	G	H	J	K	N	Q	R	S	U

Рисунок 12 – Получившийся алфавит

Так получается подстановка для каждой буквы произвольного сообщения. Исходное сообщение SEND MORE MONEY шифруется как HZBY TCGZ TCBZS. Следует отметить, что требование о различии всех букв ключевого слова не обязательно. Можно просто записать ключевое слово (или фразу) без повторения одинаковых букв.

Несомненное достоинство системы Цезаря с ключевым словом – то, что количество возможных ключевых слов практически неисчерпаемо. Недостатком этой системы является возможность взлома шифртекста на основе анализа частот появления букв.

Многоалфавитные шифры замены представляют собой класс криптографических методов, в которых для замены символов исходного текста используются несколько алфавитов. Данная техника позволяет значительно усложнить процесс расшифровки, поскольку каждый символ может быть заменен различными символами в зависимости от контекста, что делает анализ более сложным и трудоемким.

Одним из наиболее известных многоалфавитных шифров является шифр Вижинера. Шифр Виженера изобретался многократно. Впервые этот метод описал Джованни Баттиста Беллазо в книге La cifra del. Sig. Giovan Battista Bellaso в 1553 году, однако в XIX веке получил имя Блеза Виженера, французского дипломата.

Хотя шифр легко понять и реализовать, на протяжении трёх столетий он противостоял всем попыткам его взломать, благодаря чему его называли «неразгаданным шифром».

В шифре Цезаря каждая буква алфавита сдвигается на несколько позиций, например, в шифре Цезаря при сдвиге +3, A стало бы D, B стало бы E и так далее. Шифр Виженера состоит из последовательности нескольких шифров Цезаря с

различными значениями сдвига. Для зашифровывания может использоваться таблица алфавитов, называемая tabula recta или квадрат (таблица) Виженера, рисунок 13.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
A	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Рисунок 13 – Таблица Вижинера

Применительно к латинскому алфавиту таблица Виженера составляется из строк по 26 символов, причём каждая следующая строка сдвигается на несколько позиций. Таким образом, в таблице получается 26 различных шифров Цезаря.

Каждая буква исходного текста шифруется с использованием соответствующей буквы ключевого слова. Если длина ключа меньше длины текста, то ключ повторяется.

Ключ в шифре Вижинера играет центральную роль, так как именно он определяет порядок и способ замены символов. Ключ может состоять из одного или нескольких символов и повторяться по мере необходимости для соответствия длине шифруемого текста. Это делает шифр более устойчивым к атакам, основанным на частотном анализе.

Преимущества шифра Вижинера заключаются в его способности

нейтрализовать классический частотный анализ, что значительно усложняет задачу криптоанализа. Однако следует отметить и его недостатки. Шифр требует более сложной реализации и может быть труден в использовании при больших объемах данных. Несмотря на это, его историческая значимость и влияние на развитие криптографии делают шифр Вижинера важным объектом изучения в данной области.

2.2 Предмет исследования

Предметом исследования является визуализация криптографических методов, с акцентом на разработку в программных решениях, способствующих наглядному представлению различных криптографических алгоритмов. Визуализация представляет собой важный инструмент, позволяющий не только упростить восприятие сложных процессов, связанных с шифрованием и расшифровкой данных, но и значительно улучшить понимание механики работы криптографических методов. Использование программ для визуализации криптографических алгоритмов позволяет отслеживать изменения в исходных и конечных данных, что способствует более глубокому пониманию процессов, происходящих в ходе обработки информации. Визуальные представления позволяют исследовать динамику преобразований данных, выявлять закономерности и аномалии, а также анализировать эффективность различных методов шифрования.

Визуализация – это процесс преобразования данных и информации в графические или визуальные форматы, что позволяет упростить восприятие, анализ и интерпретацию сложных наборов данных. Этот метод использует различные визуальные элементы для представления информации в наглядной и доступной форме.

2.2.1 Способы визуализации

Одними из наиболее распространенных способов визуализации, способствующий более глубокому пониманию процессов шифрования и дешифрования являются:

– графики и диаграммы. Графики и диаграмм в древовидной форме демонстрируют алгоритмы шифрования. В таких визуализациях исходные данные

представлены на входе, а конечные данные – на выходе, что позволяет наглядно проследить за последовательностью преобразований информации на каждом этапе обработки;

– сравнительные таблицы. Сравнительные таблицы являются одним из самых популярных инструментов для визуализации криптографических методов, позволяя наглядно сопоставлять исходные и зашифрованные данные. Они могут включать столбцы, такие как «Исходные данные», «Шифр», «Ключ» и «Зашифрованные данные»;

– анимация. Создание анимаций как в двухмерном, так и в трехмерном пространствах также является эффективным методом визуализации криптографических процессов. Анимации позволяют динамически продемонстрировать процесс шифрования, визуализируя изменения данных в реальном времени;

– интерактивный подход. Интерактивный подход включает применение пользовательского интерфейса в игровой форме. Такой подход предоставляет пользователю возможность активно взаимодействовать с процессом шифрования, изменяя параметры и входные данные. В результате пользователь может наблюдать за изменениями в реальном времени, что способствует более глубокому осмыслению за счёт практического применения и самостоятельно исследования.

3 ОБЗОР СУЩЕСТВУЮЩИХ МЕТОДОВ РЕШЕНИЯ АНАЛОГИЧНЫХ ТИПОВЫХ ЗАДАЧ

Анализ существующих методов решения аналогичных типовых задач является важным этапом в процессе разработки программной визуализации криптографических методов перестановки и замены. Прежде всего, такой обзор позволяет выявить существующие подходы и технологии, которые могут быть адаптированы или модифицированы для достижения поставленных целей. Важно отметить, что изучение ранее реализованных решений помогает определить их преимущества и недостатки, что в свою очередь может помочь избежать повторения ошибок, допущенных в предыдущих работах.

Кроме того, анализ аналогичных методов дает возможность оценить эффективность различных алгоритмов и подходов к визуализации, что может служить основой для выбора оптимальных решений для конкретной задачи. Также следует учитывать, что исследование аналогичных методов может способствовать генерации новых идей и подходов.

В процессе анализа существующих методов визуализации криптографических алгоритмов были выделены три основных типа аналогов:

– консольные приложения;

– веб-ресурсы;

– компьютерные игры.

Каждый из этих типов имеет свои особенности и ограничения, которые необходимо учитывать при разработке эффективного инструмента для визуализации криптографических методов.

Консольные приложения, как один из примеров, представляют собой простые программы, часто реализуемые на языках программирования, таких как C#. В качестве конкретного примера можно привести приложение, демонстрирующее принцип шифра Скитала, рисунок 14.

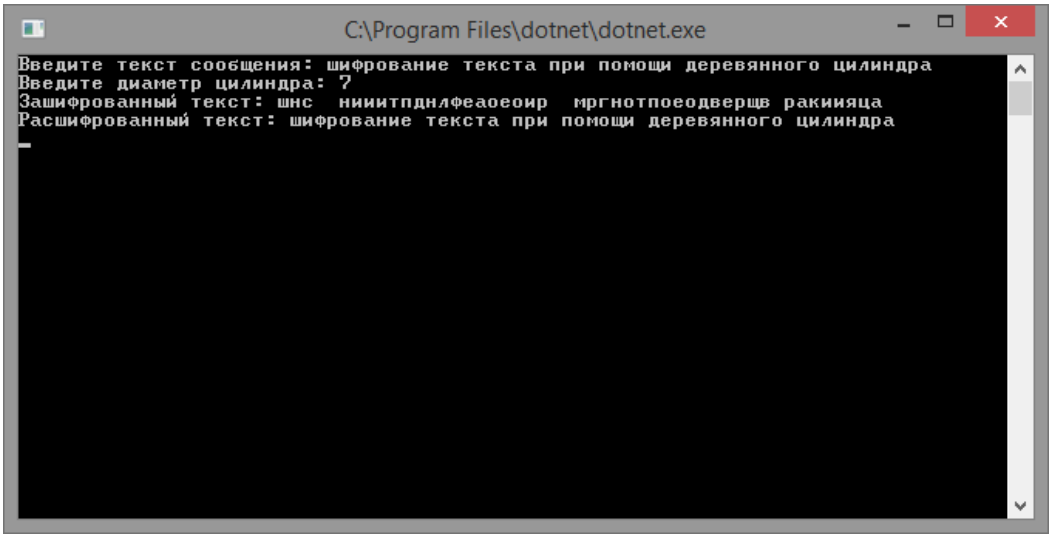


Рисунок 14 – Консольное приложение

Несмотря на свою простоту, такие приложения имеют ряд недостатков в контексте образовательных целей. Во-первых, они предлагают минимальный функционал: пользователь может только ввести сообщение и диаметр цилиндра, что ограничивает возможности взаимодействия с алгоритмом.

Кроме того, существенным минусом таких программ является недостаток информации о самом алгоритме шифрования. Обычно в консольных приложениях отсутствуют объяснения или комментарии, которые могли бы помочь пользователю лучше понять принципы работы шифра. В большинстве случаев они реализуют визуализацию в виде сравнительных таблиц, что не дает пользователю полного понимания и представления о процессе шифрования и дешифрования.

Ключевым недостатком представленного инструмента является отсутствие визуализации процесса шифрования. Для шифра Скитала, основанного на концепции физического расположения букв на цилиндре, понимание механизма работы алгоритма невозможно без наглядного представления. Визуализация позволяет пользователю осознать, как именно происходит перестановка символов и каким образом формируется зашифрованное сообщение.

В результате, несмотря на наличие множества подобных приложений, их функционал зачастую оказывается ограниченным и не предоставляет

пользователю полноценного опыта взаимодействия с алгоритмами шифрования

В качестве примера веб-ресурса, предлагающего визуализацию криптографических алгоритмов, можно рассмотреть сайт PlanetCalc, который включает в себя функционал для работы с шифром Виженера, рисунок 15.

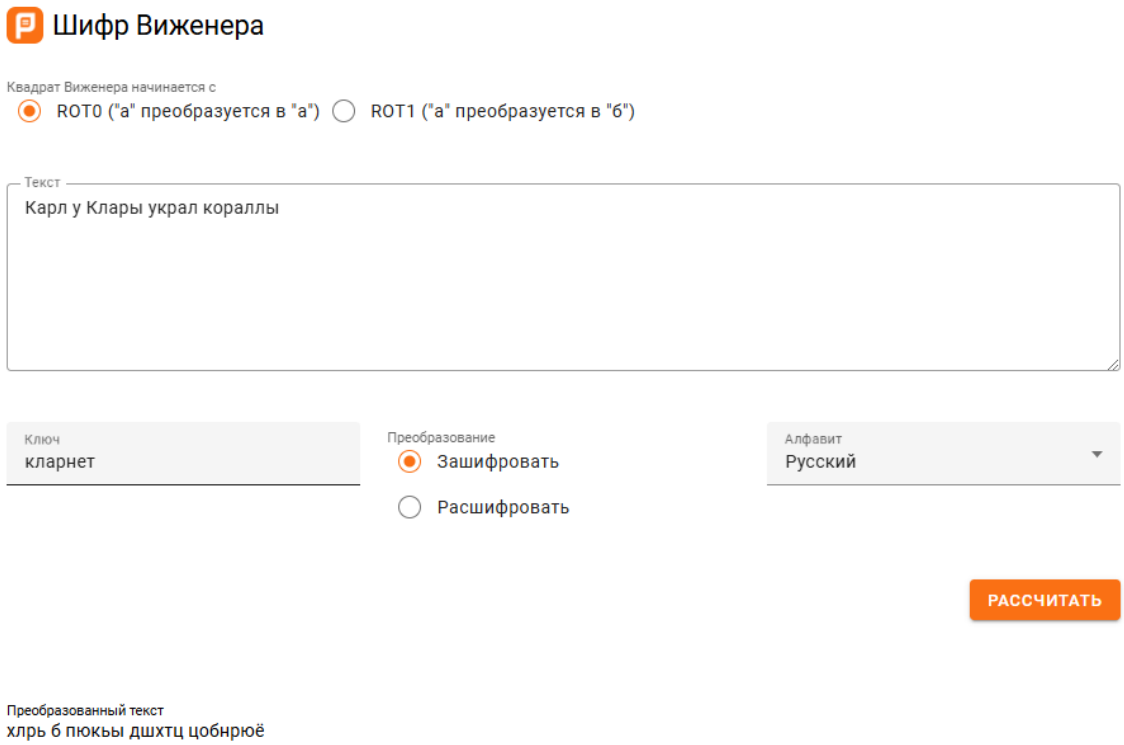


Рисунок 15 – PlanetCalc

Этот ресурс не только предоставляет пользователям возможность шифровать и дешифровать сообщения, но и предлагает теоретический материал, объясняющий принципы функционирования данного алгоритма. Наличие такой информации является значительным преимуществом, так как оно помогает пользователям лучше понять, как работает шифр и какие математические основы лежат в его основе.

Тем не менее, несмотря на наличие теоретических объяснений, подобные сайты зачастую ограничиваются представлением лишь одного метода шифрования. Это может быть недостатком для пользователей, стремящихся получить более широкое представление о криптографии и различных методах шифрования. В образовательных целях было бы более эффективно демонстрировать процесс шифрования шаг за шагом, что позволило бы пользователю не только увидеть

конечный результат, но и понять каждый этап преобразования данных.

Такой подход мог бы включать визуализацию промежуточных результатов, а также объяснения на каждом шаге, что сделало бы процесс обучения более интерактивным и увлекательным. Например, пользователи могли бы наблюдать, как каждое символическое преобразование влияет на итоговое сообщение, а также получать подсказки и комментарии о том, какие математические операции выполняются в данный момент.

Таким образом, хотя веб-ресурсы, такие как PlanetCalc, предоставляют полезные инструменты для изучения отдельных криптографических методов, их функционал можно значительно расширить для достижения более эффективного образовательного результата.

В рамках обзора существующих методов решения аналогичных типовых задач следует обратить внимание на компьютерные игры как эффективный инструмент для обучения. В качестве примера можно рассмотреть игру Cypher, в которой игрокам предоставляется возможность изучать различные методы шифрования, включая методы перестановки и замены, рисунок 16.

Рисунок 16 – Игра Cypher

Данная игра обладает рядом достоинств, среди которых выделяются наличие теоретического материала и практических заданий. Эти аспекты способствуют углублению знаний пользователей в области криптографии и формированию практических навыков.

Тем не менее, несмотря на указанные положительные характеристики, Cypher сталкивается с рядом значительных недостатков.

Одной из основных проблем является визуальный стиль игры. Дизайн, несмотря на свою простоту, включает в себя абсолютно белые и яркие элементы окружения, что может вызывать дискомфорт у игроков и отвлекать их от учебного процесса.

Кроме того, представленный теоретический материал в игре не сопровождается должным объяснением, что приводит к тому, что пользователи оказываются предоставленными самим себе в процессе освоения информации. Отсутствие четких инструкций и пояснений может затруднить понимание сложных концепций шифрования.

Серьезным ограничением является также коммерческая модель игры. Cypher распространяется на платной основе, а доступ к ней из России в настоящее время невозможен.

Все рассмотренные аспекты подчеркивают необходимость более продуманного подхода к разработке программной визуализации криптографических методов, чтобы обеспечить доступность и эффективность обучения для более широкой аудитории.

В результате проведенного анализа можно сделать вывод о том, что существующие инструменты для визуализации криптографических методов перестановки не совершенны. Это создает потребность в разработке более сложного и интуитивно понятного инструмента, который мог бы эффективно визуализировать процессы шифрования и способствовать более глубокому усвоению криптографических концепций.

Таким образом, создание интегрированного инструмента для визуализации криптографических методов перестановки и замены является актуальной задачей.

4 ПОСТАНОВКА ЗАДАЧИ И ВЫБОР ИНСТРУМЕНТОВ ДЛЯ ЕЕ РЕАЛИЗАЦИИ

4.1 Проектирование программного продукта

В результате анализа предметной области, связанной с криптографическими методами перестановки и замены, подходов к их визуализации, а также в результате исследования положительных и отрицательных сторон аналогов, были определены ключевые аспекты, требующие дальнейшего внимания при разработке программной визуализации.

Важным этапом в решении задачи является формулировка задачи, а также определения методик для её решения.

4.1.1 Формулировка задачи

В рамках программной визуализации было принято решение продемонстрировать функционирование таких шифров, как шифр Скитала, шифрующие таблицы, магический квадрат, масонский шифр, шифр Цезаря и шифр Вижинера. Эти методы являются одними из самых популярных и известных с древнейших времен. Простота и понятность алгоритмов их работы делают их доступными для понимания даже для лиц, не обладающих обширными знаниями в области криптографии.

Ключевым аспектом разработки программной визуализации является интеграция теоретических знаний о шифрах, их алгоритмах и истории происхождения. Это позволит не только углубить понимание механик шифрования, но и создать контекст для практического применения полученных знаний. Необходимо также включить в программную визуализацию реализацию практических заданий на шифрование и дешифрование. У пользователя должна быть возможность изменять настройки и входные данные, что станет важным инструментом для лучшего усвоения материала.

4.1.2 Выбор архитектуры

Перед началом разработки программной визуализации криптографических методов необходимо выбрать и изучить архитектуру для будущего проекта.

Для разработки программной визуализации была выбрана архитектура Модель-представление-контроллер (MVC), рисунок 17.

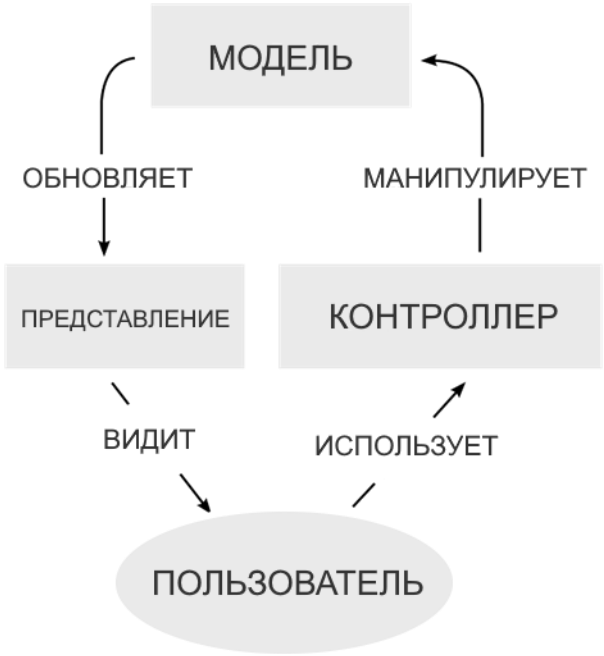


Рисунок 17 – Архитектура MVC

Данная архитектура, несмотря на свою меньшую гибкость по сравнению с альтернативными подходами, такими как Model-View-Presenter (MVP) или Model-View-ViewModel (MVVM), способствует созданию более структурированного и управляемого учебного процесса.

Она представляет собой паттерн проектирования, широко используемый в разработке программного обеспечения для разделения логики приложения на три взаимосвязанных компонента:

– модель. Модель является основным компонентом архитектуры MVC и отвечает за управление данными и бизнес-логикой приложения. Она инкапсулирует состояние приложения, обрабатывает данные и предоставляет интерфейсы для их изменения. В контексте проекта модель представляет собой классы, которые хранят информацию о настройках проекта, такие как настройки графики, пользователя, информацию о сценах. Модель не зависит от представления и контроллера, что позволяет изменять визуальную часть приложения без

необходимости модификации логики данных.

– представление. Представление отвечает за отображение данных пользователю и взаимодействие с ним. Этот компонент связан с визуальным представлением информации, полученной от модели, и обновляет интерфейс в ответ на изменения данных. Представление включает в себя игровые объекты, UI-элементы и анимации, которые отображают состояние модели.

– контроллер. Контроллер служит связующим звеном между моделью и представлением. Он получает пользовательский ввод, обрабатывает его и обновляет модель в соответствии с действиями пользователя. Контроллер также уведомляет представление о необходимости обновления, когда данные изменяются. Контроллер реализован через скрипты, которые обрабатывают события, такие как нажатия кнопок или движения мыши.

4.1.3 Функциональная структура

После выбора архитектуры необходимо более подробно рассмотреть функции программной визуализации. Для этого потребуется составить и описать функциональную структуру. Функциональная структура представляет собой систематизированное описание всех функций и возможностей, которые предоставляет программа, а также их взаимосвязей и взаимодействия с пользователем.

С помощью функциональной структуры можно детально описать, как пользователь будет взаимодействовать с программной визуализацией. В первую очередь, пользователь обращается к интерфейсу «Главного меню», где он имеет возможность выбрать интересующий его метод шифрования.

После выбора пользователь взаимодействует с интерфейсом выбранного метода, который включает в себя кнопки навигации, такие как «Далее» и «Назад».

В процессе проектирования было принято решение сделать игровой процесс более простым и интуитивно понятным. Были исключены лишние функции, такие как возможность свободного передвижения, чтобы не нагружать пользователя избыточной информацией и действиями. Таким образом, игровой процесс будет напоминать интерактивную лекцию: пользователь нажимает на кнопки

«Далее» и «Назад», просматривая лекции, дополненные изображениями и визуализацией криптографических методов.

Кроме того, пользователь будет взаимодействовать с интерфейсом практических заданий, которые помогут ему лучше разобраться с материалом. Эти задания будут включать в себя различные интерактивные элементы и визуализации, способствующие более глубокому пониманию представленных тем.

Все необходимые модели, логика, анимации и изображения для практических заданий, а также для просмотра сцен будут загружаться из папки Assets.

На каждой сцене будет предусмотрена кнопка выхода в главное меню, что позволит пользователю легко возвращаться к выбору других методов шифрования без необходимости закрывать приложение.

На рисунке 18 представлена составлена и описанная функциональная структура.

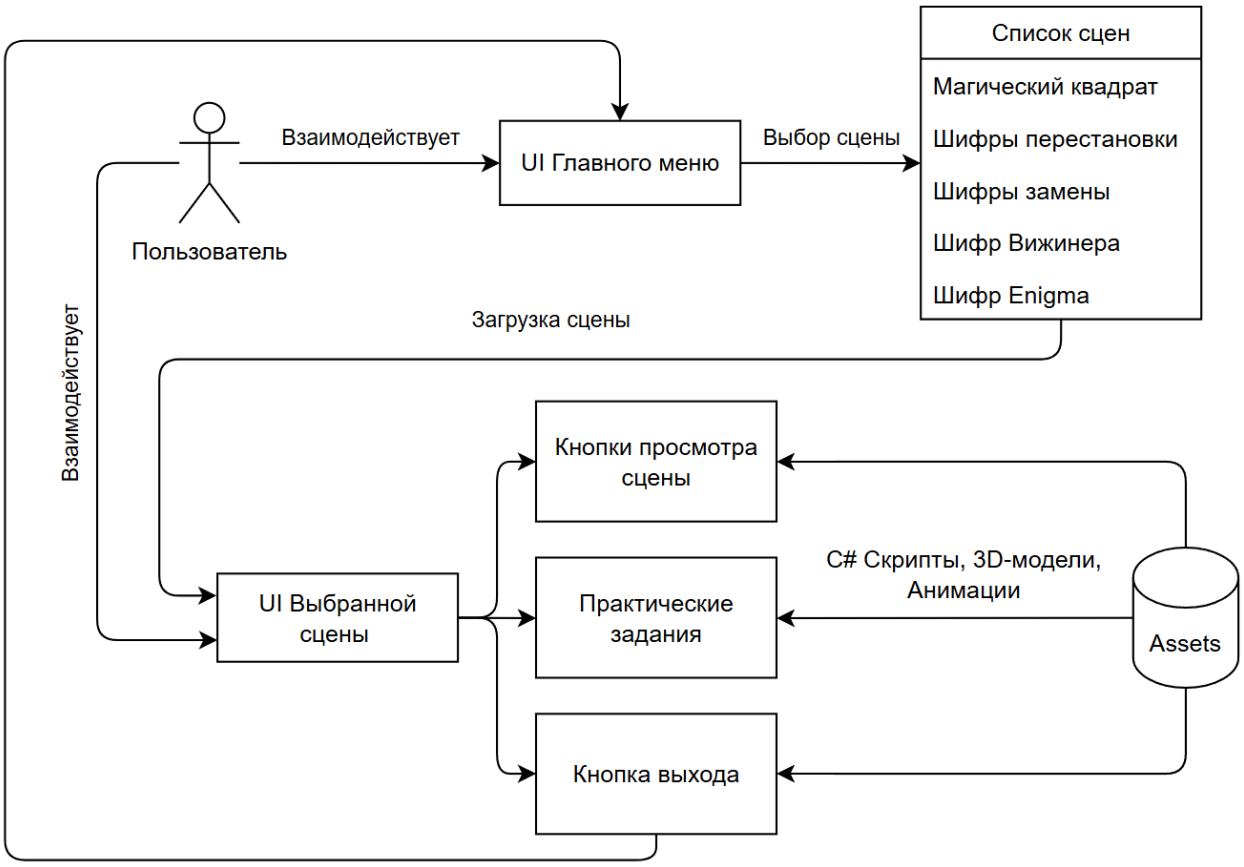


Рисунок 18 – Функциональная структура

4.2 Методики решения

Исследуя способы визуализации, были выделены самые подходящие для разработки программной визуализации криптографических методов перестановки и замены. Принято решение уделить основное внимание на анимации и на интерактивном подходе с применением пользовательского интерфейса в игровой форме.

Данные способы визуализации подразумевают использование как 2D, так и 3D анимации для динамической и наглядной демонстрации алгоритмов шифрования и дешифрования. Интерактивный пользовательский интерфейс, в свою очередь, также позволит пользователю активно участвовать в процессе обучения. Также при использовании интерактивного пользовательского интерфейса можно легко реализовать практические элементы, понятные для пользователя.

Игровая форма подразумевает, что разрабатываемое приложение должно быть доступным, в плане взаимодействия с интерфейсом и привлекательным на вид, из этого следует, что необходимо уделить особое внимания к внешнему виду приложения, который должен быть эстетически привлекательным, но при этом не отвлекать пользователя от основной деятельности. Важно создать гармоничное сочетание визуальных элементов, которое будет способствовать комфортному взаимодействию с приложением.

В результате, разрабатываемая программная визуализация будет представлять собой лекцию, дополненную теоретической информацией, иллюстрациями и визуализацией, которая предоставляет пользователю возможность на практике применять знания о шифровании и дешифровании.

Для успешной реализации задуманного необходимо определить программные средства, которые:

– позволяют создавать как 3D модели, так и анимации;

– позволяют создавать интерактивные приложения и работать как с 2D, так и с 3D графикой, а также с анимационными элементами.

4.3 Существующие программные средства для решения задачи

4.3.1 Программное обеспечение для работы с 3D и анимацией

При выборе программного обеспечения для создания 3D моделей и анимаций важно учитывать функционал и возможности различных инструментов, чтобы реализовать поставленную задачу.

3DS Max, рисунок 19, является одним из наиболее известных и мощных решений на рынке. 3DS Max – это профессиональное программное обеспечение, разработанное компанией Autodesk, которое широко используется в индустрии компьютерной графики. С помощью 3DS Max можно создавать как статичные 3D модели, так и сложные анимации с высокой степенью детализации.

Рисунок 19 – Логотип

Ключевыми преимуществами 3DS Max являются:

– мощный инструментарий для создания реалистичного освещения и материалов;

– множество шейдеров и текстур;

– инструменты для симуляции физических свойств объектов;

– возможности для анимации персонажей и создания сложных движений.

Тем не менее, несмотря на все свои достоинства, 3DS Max является профессиональным инструментом, который может оказаться избыточным для реализации поставленной задачи. Его сложный функционал и высокая степень детализации больше подходят для создания реалистичных сцен и высококлассной графики, что не является необходимым для разрабатываемого проекта. Для выполнения несложных задач по моделированию и анимации необходимо использовать более простые и доступные решения, которые обеспечат необходимый уровень качества без лишних затрат времени и ресурсов.

Cinema 4D, рисунок 20, это профессиональное программное обеспечение,

разработанное компанией Maxon, которое широко используется в индустрии графики и анимации. Cinema 4D известна своим интуитивно понятным интерфейсом и гибкостью, что позволяет пользователям быстро осваивать программу и эффективно работать над проектами.

Рисунок 20 – Логотип Cinema 4D

Ключевыми преимуществами Cinema 4D являются:

– мощный модуль для анимации и динамики;

– инструменты для создания эффектов частиц и симуляции жидкостей, что делает её отличным выбором для создания визуальных эффектов в фильмах и рекламе;

– поддержка интеграции с другими популярными приложениями, такими как Adobe After Effects.

Тем не менее, Cinema 4D является платным программным обеспечением. Более того, данная программа прекрасно подходит для создания роликов, фильмов и рекламных материалов, но не как для разработки программной визуализации.

Одним из наиболее популярных вариантов является Blender, рисунок 21, который зарекомендовал себя как универсальный инструмент для создания визуального контента.

Рисунок 21 – Логотип Blender

Blender – это бесплатное и открытое программное обеспечение, которое предлагает широкий спектр функций для моделирования, текстурирования,

рендеринга и анимации. Blender включает в себя возможности для создания анимаций, симуляции частиц и жидкостей. Одним из значительных преимуществ Blender является его активная поддержка разработчиками и пользователями. Благодаря этому пользователи могут рассчитывать на регулярные обновления и расширения.

Для реализации 3D-моделей и анимации было принято решение использовать именно Blender. Ключевым фактором при выборе стало существование огромного количества ресурсов, включая книги, форумы и обучающие материалы в текстовом и видео формате, что облегчает процесс обучения и освоения программы.

4.3.2 Среда разработки

При выборе среды разработки для программной визуализации, чтобы оптимизировать процесс создания, было принято решение остановиться на среде разработки с интегрированным движком, что позволит значительно сократить время на реализацию проекта.

Одним из наиболее известных аналогов является Unreal Engine, рисунок 22.

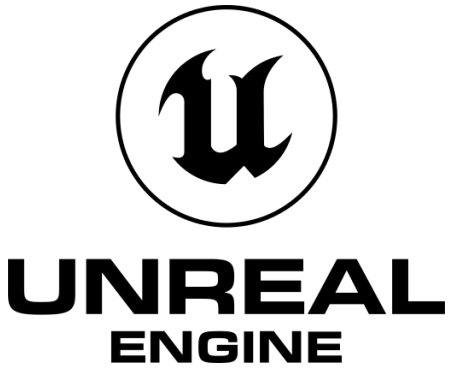


Рисунок 22 – Логотип Unreal Engine

Этот мощный инструмент предлагает широкий спектр возможностей, включая:

– поддержка реалистичного рендеринга и освещения;

– интуитивно понятный интерфейс;

– поддержка создания сложных анимаций и симуляций;

– сетевые возможности (встроенные инструменты для разработки многопользовательских игр и приложений);

– большое сообщество (активное и поддерживающее сообщество разработчиков, где можно найти множество обучающих материалов, форумов и ресурсов, что облегчает процесс обучения и решения возникающих вопросов).

Несмотря на все свои преимущества, Unreal Engine лучше всего подходит для создания больших и сложных проектов, требующих высокой производительности и детализированной графики. В связи с этим использование Unreal Engine для текущего проекта нецелесообразно, так как он может потребовать значительных временных и ресурсных затрат, которые не соответствуют поставленным целям. Выбор более легковесной среды разработки с интегрированным движком представляется более оптимальным решением для достижения желаемого результата.

Godot – это открытый игровой движок, который предлагает разработчикам множество возможностей для создания игровых приложений в 2D и 3D. Godot, рисунок 23, поддерживает множество платформ и работает на собственном языке GDScript.

Рисунок 23 – Логотип Godot

Преимущества Godot:

– полностью бесплатный и доступный для модификации;

– поддержка 2D и 3D;

– интуитивный интерфейс: удобная система сцены и визуальный редактор, упрощающие процесс разработки;

– гибкость: возможность использования других языков программирования, таких как C# и VisualScript;

Несмотря на все свои преимущества, Godot является относительно новым движком и не обладает такой популярностью, как более известные платформы. Это приводит к тому, что сообщество пользователей не так развито, а количество методических материалов, таких как форумы, тестовые и видео уроки, ограничено. В результате это может негативно сказаться на процессе разработки при возникновении нетипичных проблем.

Unity – это мощный и универсальный игровой движок, который широко используется для разработки игровых приложений как в 2D, так и в 3D, рисунок 24. Он предлагает разработчикам обширные возможности для создания интерактивных приложений и визуализаций, что делает его одним из самых популярных инструментов в индустрии. Unity поддерживает множество платформ, включая ПК и мобильные устройства.

Рисунок 24 – Логотип Unity

Преимущества Unity:

– многофункциональность (поддержка как 2D, так и 3D разработки);

– разнообразие инструментов для работы с анимацией, физикой и графикой;

– возможность экспорта проектов на множество платформ, включая IOS и Android;

– поддержка C# (использование популярного языка программирования, что облегчает интеграцию с другими системами);

– удобный редактор, который упрощает процесс создания игровых приложений;

– большое сообщество (из-за наличия активного сообщества пользователей для изучения доступно множество форумов и обучающих материалов);

Unity идеально подходит для реализации программной визуализации

благодаря своей способности работать как с двумерной, так и с трехмерной графикой. Возможность работы с анимацией и простота в использовании делают его отличным выбором для создания интерактивных приложений. Проекты, разработанные на Unity, могут работать на различных системах, включая мобильные устройства. Ключевым фактором при выборе этой среды стало большое сообщество пользователей, которое предлагает множество ресурсов для обучения.

4.4 Структура программы в Unity

Структура проекта в Unity организована в виде иерархии папок и файлов. Важнейшими элементами структуры являются сцены, скрипты и ассеты, рисунок 25.

Рисунок 25 – Структура проекта Unity

В среде Unity ключевым элементом является сцена (Scene). Сцены представляют собой отдельные уровни или экраны приложения, на которых размещаются игровые объекты, включая трехмерные модели и элементы пользовательского интерфейса (UI). Эти объекты динамически изменяются и взаимодействуют с пользователями.

Игровые объекты в Unity содержат компоненты, которые являются основными строительными блоками, позволяющими добавлять функциональность и определять поведение объектов, а также их внешний вид. Компоненты обеспечивают гибкость в управлении объектами, позволяя адаптировать их поведение в зависимости от условий взаимодействия с пользователем.

Важно отметить, что некоторые объекты, такие как элементы пользовательского интерфейса (UI), EventSystem и пустые объекты, существуют исключительно на уровне сцены и не требуют хранения в ассетах.

Ассеты (Assets) представляют собой папку внутри проекта, в которой хранятся скрипты на языке C#, трехмерные модели, анимации, изображения,

текстуры и аудиофайлы. Эти ресурсы подгружаются на сцену по мере необходимости.

Скрипты, написанные на языке C#, играют центральную роль в управлении логикой поведения объектов и их взаимодействием между собой. Скрипты могут быть присоединены к игровым объектам в качестве компонентов, что позволяет динамически изменять поведение объектов в зависимости от пользовательского ввода или других событий, происходящих в приложении.

Билд (Build) в Unity представляет собой процесс компиляции проекта в исполняемый файл, который можно запустить на целевой платформе, такой как Windows, macOS, Android, iOS и других.

BuildSettings в Unity – это окно, которое предоставляет разработчикам возможность настраивать параметры сборки проекта. В этом окне можно управлять различными настройками сборки. Основные элементы, включаемые в BuildSettings, это настройка платформы, управление сценами, настройки конфигурации и параметры игрока (Player Settings) – это доступ к более детализированным настройкам приложения, включая название игры, иконку, разрешение экрана и другие важные параметры.

В общем виде проект на Unity выглядит следующим образом, рисунок 26.

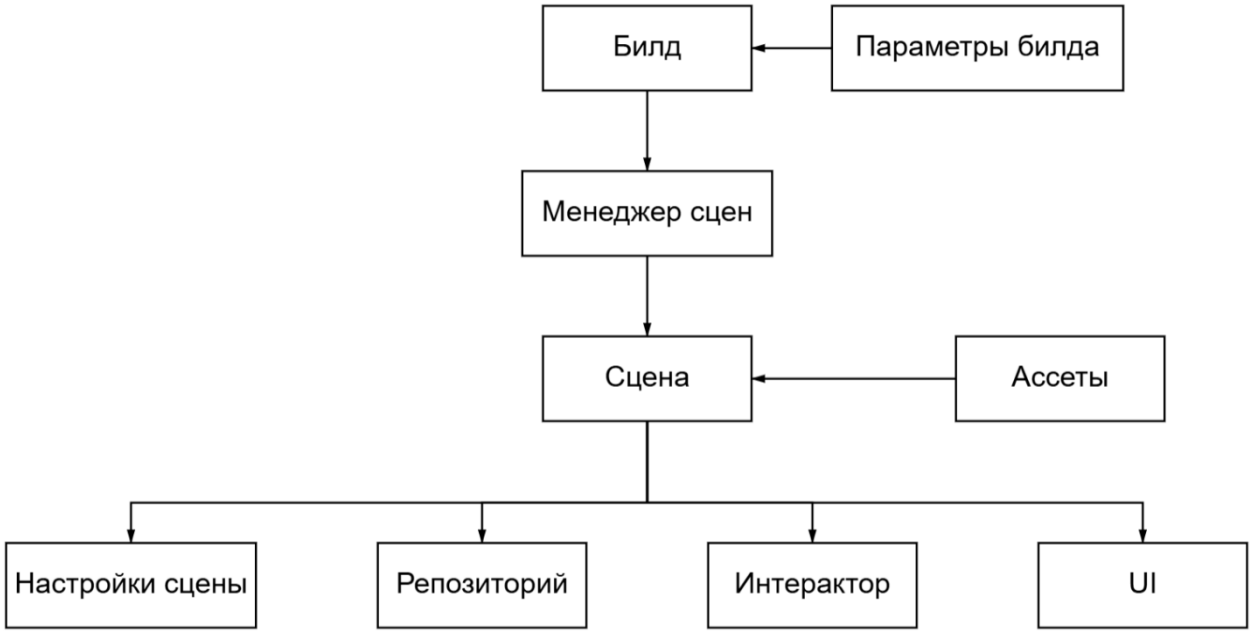


Рисунок 26 – Внутренняя структура проекта Unity

5 ПРОЕКТИРОВАНИЕ И РАЗРАБОТКА 3D-МОДЕЛЕЙ

5.1 Выбор визуального стиля

В процессе проектирования и разработки 3D-моделей для программной визуализации криптографических методов перестановки и замены первостепенное значение имеет выбор визуального стиля. Данный аспект является критически важным, поскольку создаваемые 3D-модели будут выполнять функцию заднего фона, а значит, их визуальный стиль должен быть сдержанным и не отвлекать внимание от основного функционала.

Визуальные элементы должны гармонично вписываться в общую концепцию, создавая исторический контекст происхождения шифров. При этом реалистичное окружение может оказаться избыточным, так как его разработка потребует значительных временных затрат и ресурсов. Высокая детализация моделей может привести к увеличению нагрузки на систему, что негативно скажется на производительности приложения и восприятии пользователем.

Таким образом, выбранный визуального стиль должен основываться на принципах простоты и эффективности. Оптимальным решением станет использование стилизованных 3D-моделей, которые можно быстро создать и которые не потребуют больших вычислительных ресурсов и при этом смогут создать нужную атмосферу.

Для заданных целей прекрасно подойдет визуальный стиль LowPoly (низко-полигональный), рисунок 27. Этот стиль характеризуется использованием небольшого количества полигонов для создания моделей, что придает им упрощенный, но при этом выразительный вид. Благодаря LowPoly можно создавать простые, но в то же время запоминающиеся модели, которые, несмотря на свою минималистичность, отлично ассоциируются с реальными объектами.

Рисунок 27 – Сравнение низко-полигонального стиля (слева) с высоко-полигональным стилем (справа)

Одним из ключевых преимуществ стиля LowPoly является то, что созданные модели и окружения выглядят красиво и выразительно, но не отвлекают пользователя от основного функционала. Простота форм позволяет легко воспринимать информацию, сосредоточив внимание на изучаемых криптографических методах.

Кроме того, создание моделей в этом стиле не занимает много времени, поскольку они состоят из простых геометрических фигур. Это значительно ускоряет процесс разработки и позволяет сосредоточиться на функционале приложения. Также стоит отметить, что низко-полигональные модели требуют минимальных ресурсов устройства, что способствует улучшению производительности и плавности работы приложения.

Все перечисленные факторы делают стиль LowPoly идеальным выбором для создания моделей для программной визуализации, позволяя эффективно сочетать эстетическую привлекательность и функциональность.

5.2 Разработка 3D-моделей

Для того чтобы не только продемонстрировать технические аспекты криптографии, но и создать атмосферу, соответствующую культурным особенностям той эпохи и региона, в котором данный шифр был разработан, необходимо

создать модели, которые способны погрузить пользователя в исторический контекст возникновения шифров.

В качестве примера рассмотрим подробнее процесс создания модели китайской пагоды и локации для представления шифра «Магический квадрат».

5.2.1 Сбор вспомогательных изображений

Первым этапом в процессе создания 3D-модели китайской пагоды является сбор вспомогательных изображений, известных как референсы. Референс – это изображения, фотографии или иллюстрации, которые служат источником вдохновения и ориентиром для художника или моделлера. Они помогают визуализировать концепцию и передать желаемые характеристики объекта, а также обеспечить точность и правдоподобие в создании модели.

В данном случае, при разработке модели китайской пагоды, референсы необходимы для того, чтобы избежать ошибок и неточностей в процессе моделирования. Ссылки на такие изображения позволяют лучше понять архитектурные элементы пагоды, ее пропорции и стилистические особенности, что особенно важно для передачи культурного контекста.

Выбор подходящих референсов может быть сложной задачей, особенно когда требуется учитывать множество аспектов, таких как стиль, эпоха и региональные особенности. Однако в данном случае, поскольку не требуется создавать детализированную модель, можно воспользоваться также менее сложными изображениями. Идеальным выбором послужат плоские изображения с минимальным количеством деталей, которые акцентируют внимание на основных формах и силуэтах пагоды, как на рисунке 28. Такие референсы помогут сосредоточиться на ключевых элементах конструкции, обеспечивая при этом достаточную основу для дальнейшей работы над моделью.

Рисунок 28 – Вспомогательное изображение пагоды

5.2.2 Работа с объектами

Открываем Blender и создаём новый проект, на сцене по умолчанию присутствует только куб. Этот куб станет основой для нашей китайской пагоды, рисунок 29.

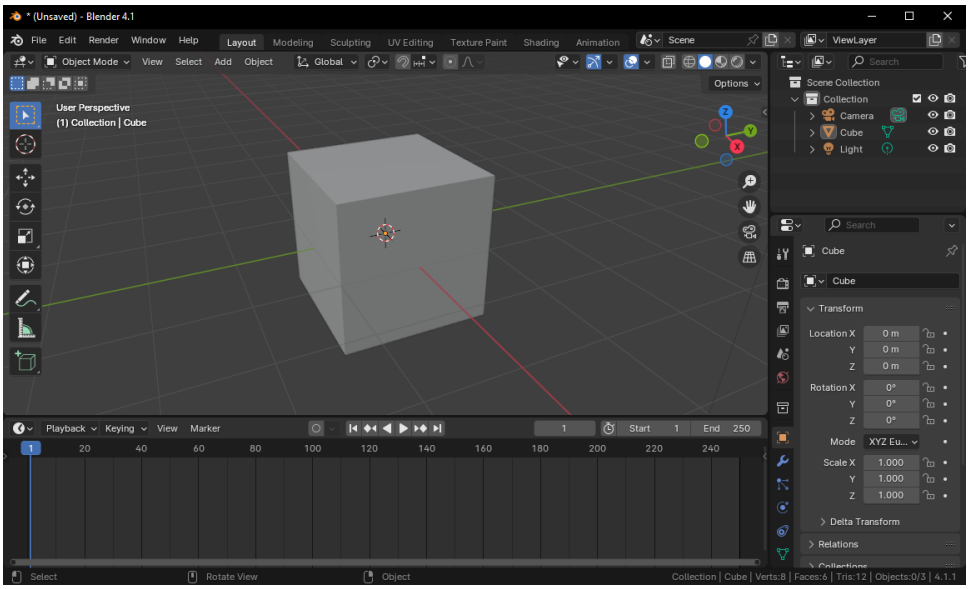


Рисунок 29 – Новый проект в Blender

Первым делом загрузим вспомогательное изображение в проект. После загрузки необходимо расположить изображение за кубом, чтобы оно служило фоном для дальнейшей работы.

Теперь активируем режим X-Ray (рентген). Эта функция позволяет видеть сквозь объекты, что особенно полезно при подгонке формы куба под контуры пагоды.

С активированным X-Ray переместим куб перед изображением, что позволит визуально ориентироваться на его силуэте. Теперь переключимся на режим редактирования (Edit Mode), нажав клавишу Tab. В этом режиме можно изменять геометрию куба.

Для того чтобы подогнать куб под форму пагоды, воспользуемся инструментами Extrude (Вытянуть) и Resize (Изменить размер). Начнем с того, что выделим верхние грани куба и вытянем их вверх с помощью Extrude, создавая тем самым первую секцию пагоды. Затем будем использовать Resize, чтобы изменить размеры и пропорции куба, подгоняя его под контуры вспомогательного изображения.

На этом этапе важно не стремиться к идеальной детализации – достаточно создать общие формы и силуэты, как на рисунке 30.

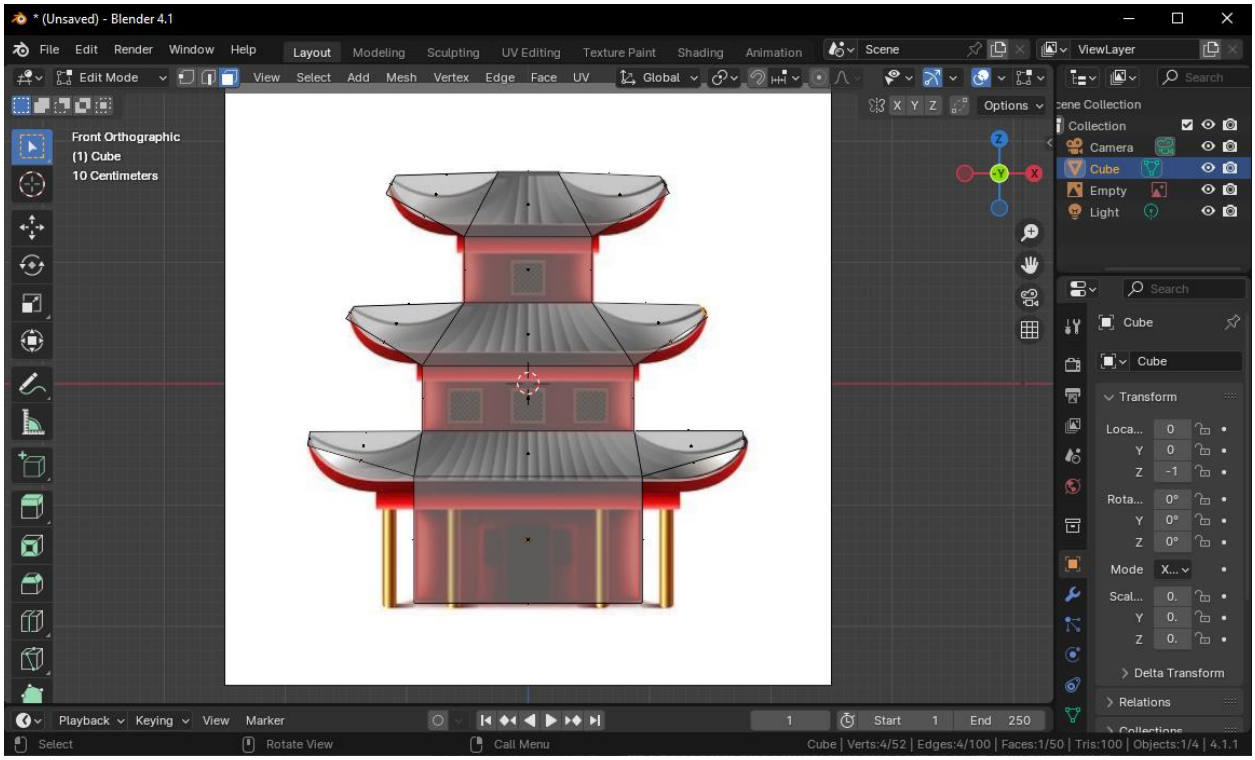


Рисунок 30 – Созданная общая форма

После того как мы создали основную форму пагоды, следующим шагом

будет добавление дополнительных граней для дальнейшей деформации. Для этого воспользуемся инструментом Loop Cut. Добавление дополнительных граней позволит более точно формировать детали пагоды, такие как крыша и стены.

После того как одна сторона пагоды будет готова, добавим модификатор Mirror (Зеркало). Это позволит автоматически отразить изменения на противоположной стороне, что значительно ускорит процесс моделирования.

Теперь, когда основная форма готова, добавим на сцену несколько кубов для создания окон, дверей и колонн, как на нашем вспомогательном изображении, рисунок 31.

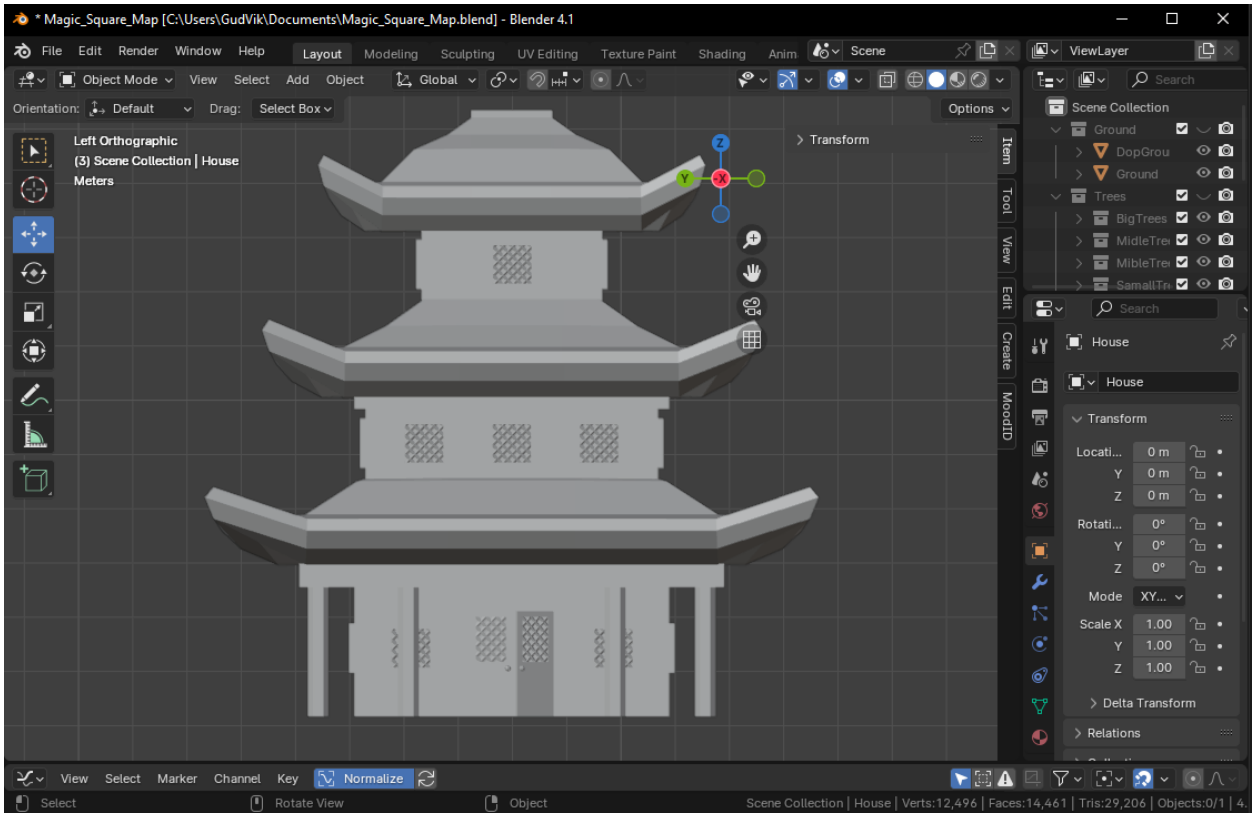


Рисунок 31 – Модель с деталями

5.2.3 Текстурирование

Как только все основные элементы будут добавлены, можно переходить к следующему этапу – текстурирование модели.

В среде Blender существует возможность создания материалов, которые позволяют изменять цвет, отражение и другие свойства поверхности объектов. Однако в данном контексте не ставится задача создания реалистичных моделей.

Вместо этого целесообразно использовать простые цвета.

Однако создание отдельного материала для каждого цвета не является оптимальным решением. При добавлении модели в среду разработки все материалы будут загружаться отдельно, что приводит к значительной нагрузке на систему, особенно если количество материалов велико. В связи с этим рекомендуется использовать единственный материал с текстурой, содержащей все необходимые цвета.

Для наложения текстуры на модель применяется метод UV Editing. Этот процесс включает в себя развертку 3D-модели на 2D-плоскости, что позволяет точно контролировать, как изображение будет отображаться на поверхности объекта. Сначала производится развертка UV, которая позволяет создать уникальную UV-карту для модели. Эта карта представляет собой 2D-отображение трехмерной поверхности, на которую затем будет наложена текстура.

После выполнения развертки в специальном редакторе можно редактировать UV-карту, перемещая, масштабируя и вращая UV-острова для достижения необходимого результата. Создание текстуры с изображением цветов осуществляется в графическом редакторе или с использованием готовых изображений. Загруженная текстура назначается материалу, что обеспечивает ее правильное отображение на модели.

Процесс текстурирования показан на рисунке 32. Слева располагается созданная текстура, в режиме UV Editing, на ней располагаются UV-острова, которые представляют плоскости объекта, перемещая их можно менять текстуру на них.

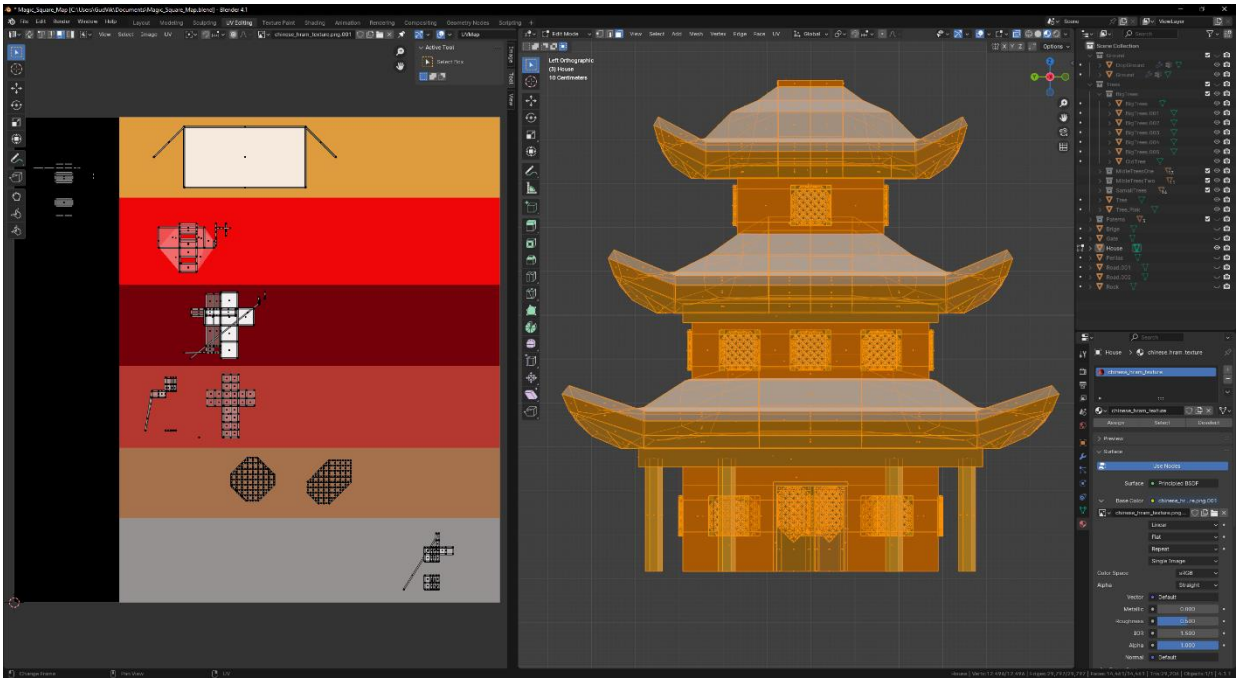


Рисунок 32 – Процесс текстурирования

Таким образом, использование метода UV Editing в сочетании с единственным материалом с текстурой значительно упрощает процесс разработки 3D-моделей. Это позволяет избежать создания множества отдельных материалов и минимизировать потребление ресурсов компьютера.

5.2.4 Создание локации

Описанным выше методом создаются и другие модели. В процессе создания локации для магического квадрата в среде Blender, также была разработана модель китайских ворот Мёдзин-тории и моста в китайском стиле. Как и в случае с предыдущими моделями, для ворот и моста использовался подход с единственным материалом и текстурой, что позволило оптимизировать загрузку ресурсов и упростить процесс визуализации.

Далее в Blender был создан макет будущей локации с использованием простых фигур, рисунок 33. Этот макет служил основой для дальнейшей работы, позволяя визуализировать общую композицию и расположение объектов в пространстве.

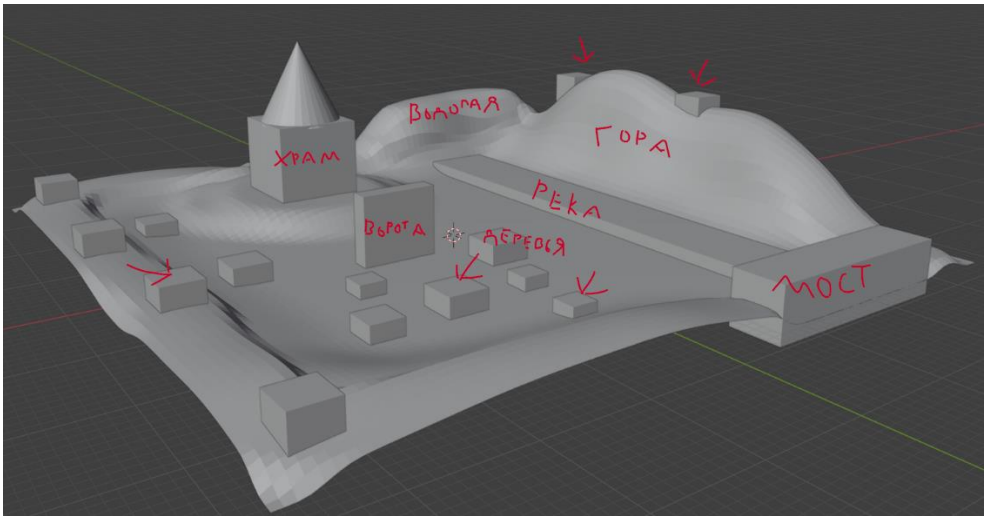


Рисунок 33 – Макет локации

Далее ориентируясь на макет, была разработана сама локация, куда были добавлены ранее созданные модели ворот, моста и пагоды. Для дополнения локации были созданы стилизованные под LowPoly элементы, такие как трава, деревья, камни, дорога и река. На рисунке 34 показан итоговый вариант локации для Магического квадрата.

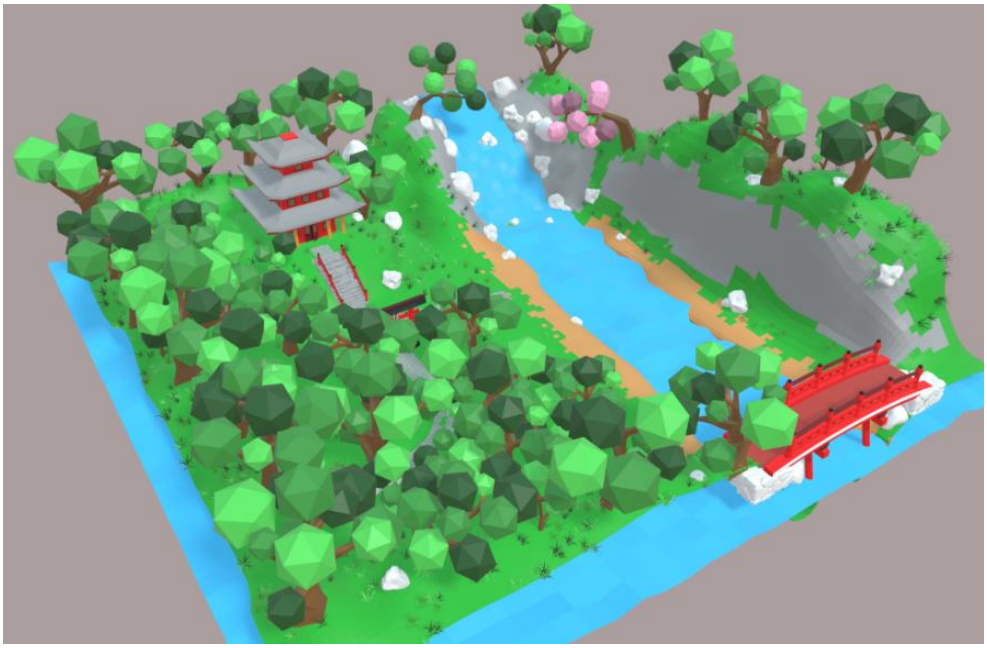


Рисунок 34 – Итоговый вариант локации для Магического квадрата

Описанным выше способом были созданы все модели и локации, которые используются в программной визуализации.

6 РАЗРАБОТКА ПРОГРАММНОЙ ВИЗУАЛИЗАЦИ

6.1 Проектирование структуры программной визуализации

После изучения общей структуры проекта в среде Unity, была описана структура будущей программной визуализации, которая описана и представлена на рисунке 35.

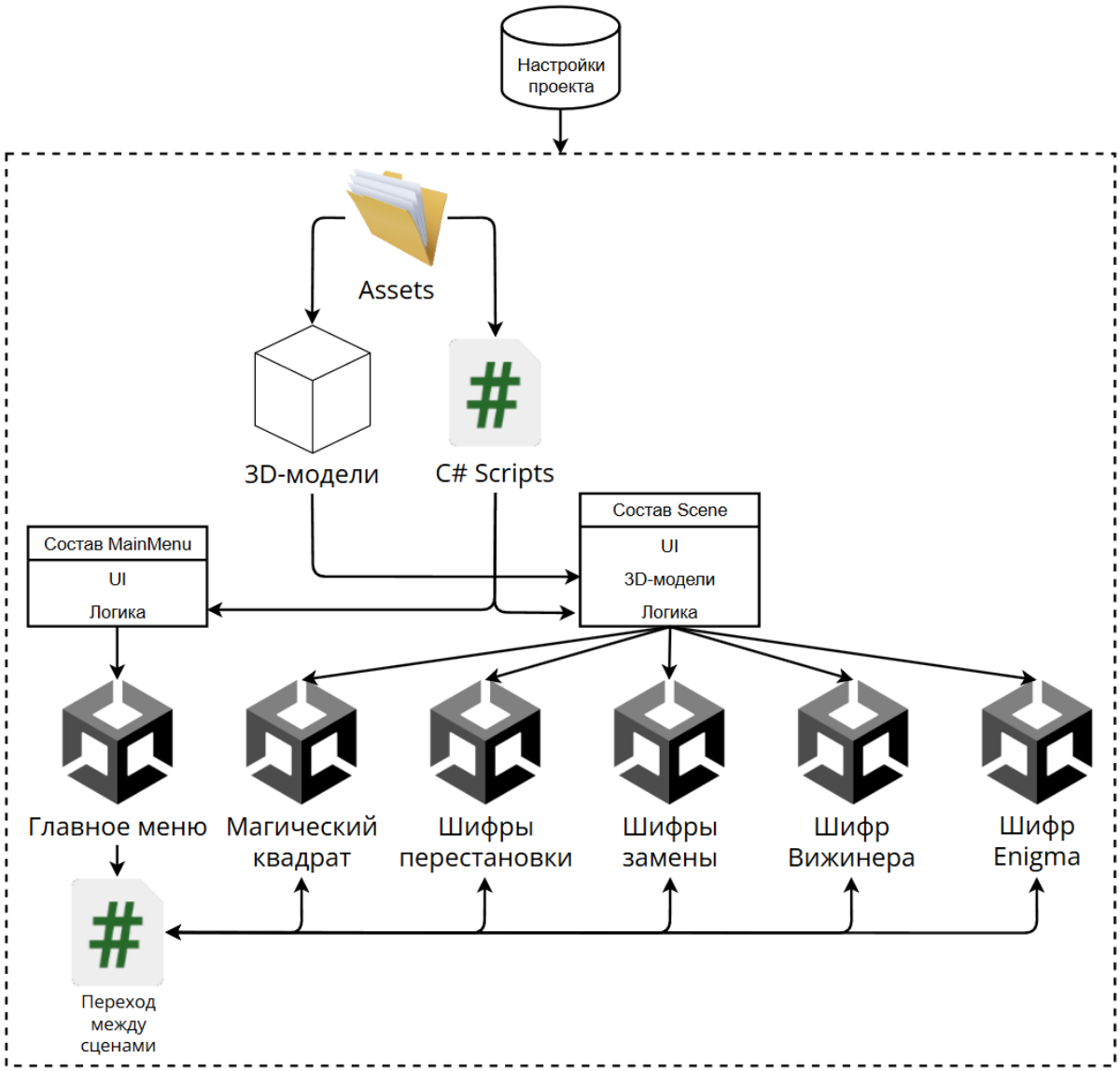


Рисунок 35 – Структура программной визуализации

Проект включает в себя шесть сцен, каждая из которых посвящена отдельному методу шифрования, а также одну сцену, выполняющую функцию главного меню. Главное меню служит центральным узлом навигации, обеспечивая

пользователю возможность выбора интересующего его метода шифрования. Переход между сценами осуществляется с помощью C# скрипта, который управляет логикой навигации и обеспечивает переход между частями программы.

Важным аспектом проектирования является использование. Все необходимые ресурсы, такие как 3D модели, логика, анимации, для каждой из сцен, подгружаются из папки Assets.

Управление настройками проекта осуществляется через его конфигурацию, однако необходимо отметить, что данная программа не требует тонкой настройки для корректной работы. В связи с этим настройки проекта будут оставлены по умолчанию, что позволит сосредоточиться на функциональности и визуализации криптографических.

6.2 Создание программы в среде разработки

6.2.1 Создание проекта

Первый этап работы заключается в создании нового проекта в Unity. После создания проекта в нижней части экрана можно видеть содержание папки Assets, которая служит хранилищем для всех объектов, скриптов и сцен, используемых в приложении.

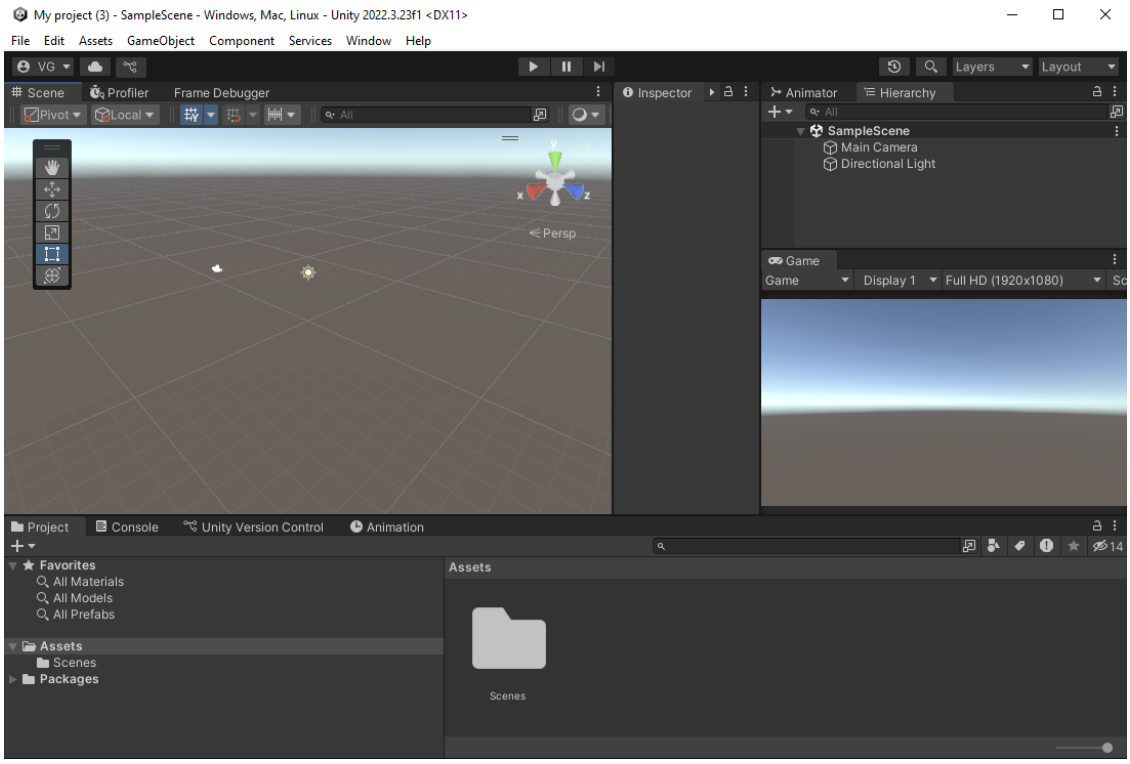


Рисунок 36 – Интерфейс Unity

В автоматическом режиме создается одна сцена, в которой изначально присутствуют только камера и источник света, рисунок 36. Камера выполняет роль игрока.

Следующим шагом является создание интерфейса для сцены с методом шифрования. Для этого на сцене добавляем элемент UI – Canvas. В инспекторе для компонента CanvasScaler устанавливаем параметр ScaleWithScreenSize, что позволяет интерфейсу адаптироваться к различным разрешениям.

На созданном Canvas могут быть добавлены различные элементы интерфейса, такие как текстовые поля, кнопки, панели, изображения и т.д. Создадим кнопки для переключения текста, что обеспечит возможность динамического изменения отображаемой информации. Также реализуем кнопку паузы, позволяющую пользователю приостанавливать текущий процесс, и кнопку выхода в главное меню.

Кроме того, необходимо предусмотреть выделенное место для отображения изображения и возможной другой дополнительной информации, которая может служить визуальным представлением криптографического процесса или иллюстрацией результата шифрования.

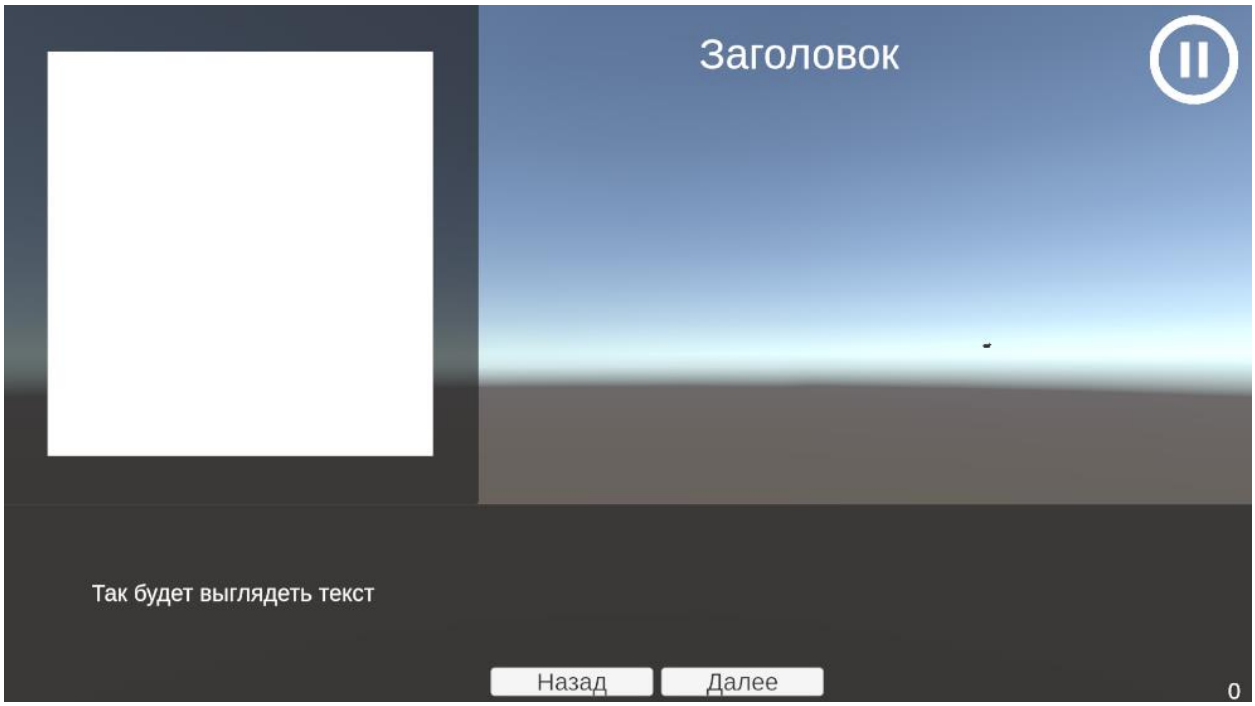


Рисунок 37 – Созданный интерфейс

Рядом с кнопками переключения текста «Назад» и «Далее» создадим текст, в котором пользователю будет представлена основная информация, а возле кнопки паузы текст, который будет служить заголовком. Также для улучшения видимости создадим, для изображения (белый квадрат) и текста, серый фон с помощью элемента UI – Panel, рисунок 37

После создания интерфейса в Unity следующим шагом является реализация логики для переключения текста и изображений. Для этого в папке Assets создадим новый C# скрипт, щелкнув правой кнопкой мыши и выбрав соответствующий пункт меню. Этот скрипт будет отвечать за обработку событий нажатия кнопок, а также за изменение отображаемого текста и изображения.

Внутри скрипта создадим публичные переменные для хранения ссылок на элементы UI, такие как текстовое поле и изображение. Это позволит привязать объекты на сцене к нашему скрипту. Для этого необходимо создать объекты соответствующего типа в скрипте, а затем вручную добавить их в инспекторе Unity. Таким образом, можно легко управлять их поведением и визуализацией, рисунок 38.

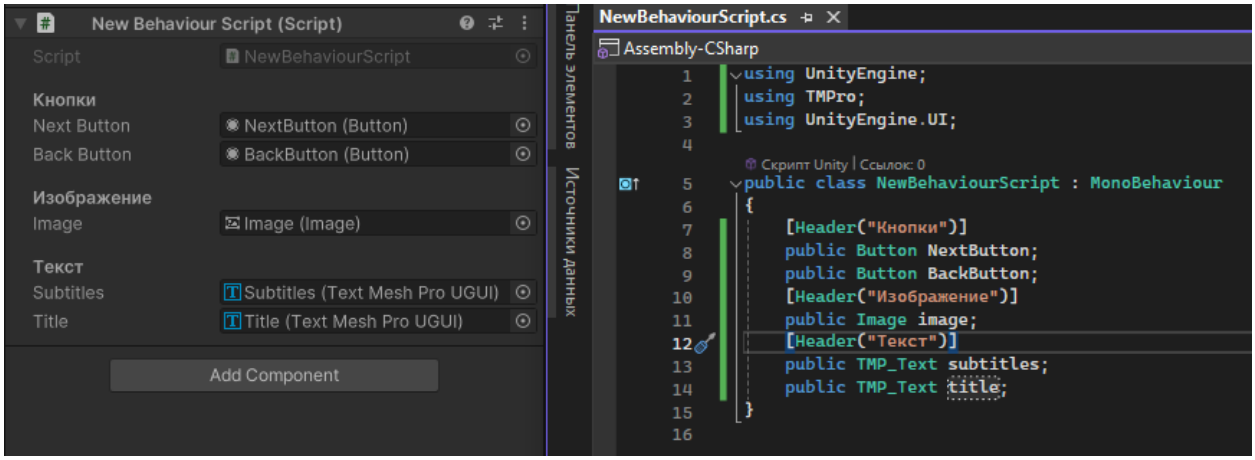


Рисунок 38 – Пример C# Скрипта

Для переключения текста и изображений будем использовать простую логику: при нажатии кнопки будет изменяться содержимое текстового поля и изображение в соответствии с заданными условиями.

Аналогичным образом создадим и настроим практические задания. На

сцене создаются необходимые объекты, такие как кнопки и текстовые поля, а в скриптах указывается логика их поведения. Пример практического задания на шифрование шифрующими таблицами представлен на рисунке 39.

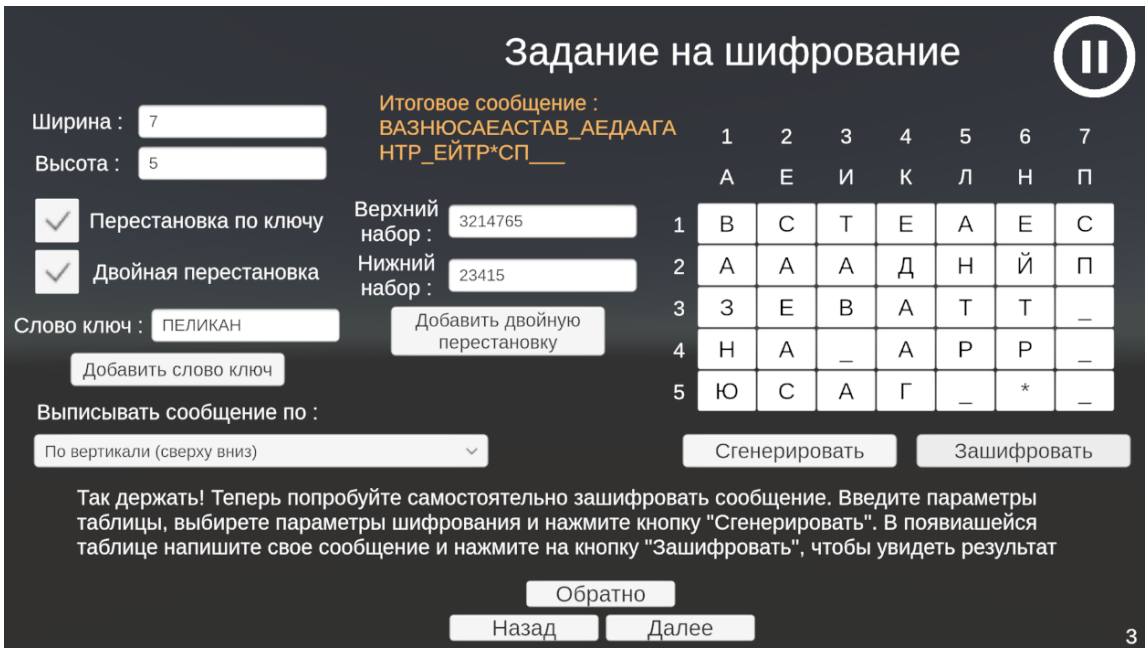


Рисунок 39 – Пример практического задания

Следующим шагом станет добавление локации, созданной в Blender. Для этого экспортируем модель из Blender в формате FBX, что обеспечит совместимость с Unity. Затем добавляем полученный файл в папку Assets. После этого модель можно перетащить на сцену, что позволит интегрировать её в программу.

Дополнительно важно настроить положение камеры и размер локации. В инспекторе Unity можно изменить параметры трансформации камеры, чтобы она корректно отображала локацию. Также можно масштабировать модель, если это необходимо, чтобы она соответствовала общей концепции сцены. На рисунке 40 представлена готовая сцена для шифров перестановки.

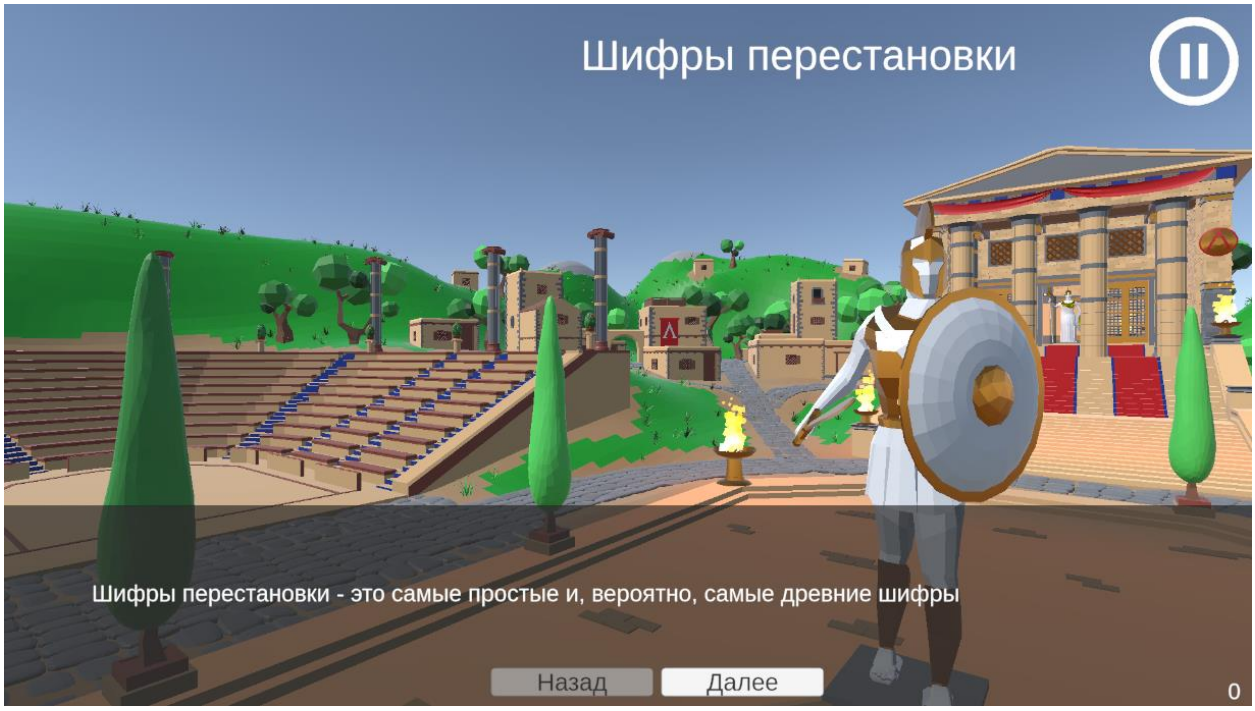


Рисунок 40 – Сцена для шифров перестановки

6.2.2 Создание визуализаций

Для разработки визуализаций были выбраны инструменты входящие в состав Blender и Unity.

В качестве примера будет рассмотрен процесс создания визуализации шифра Скитала в среде Blender. Данный шифр, основанный на использовании стержня и ленты, поэтому на первом этапе в Blender создается стержень, который будет служить основой для наматывания ленты. Лента, в свою очередь, должна быть разделена на сегменты с помощью инструмента LoopCut, что позволит более точно управлять ее движением в процессе анимации, рисунок 41.

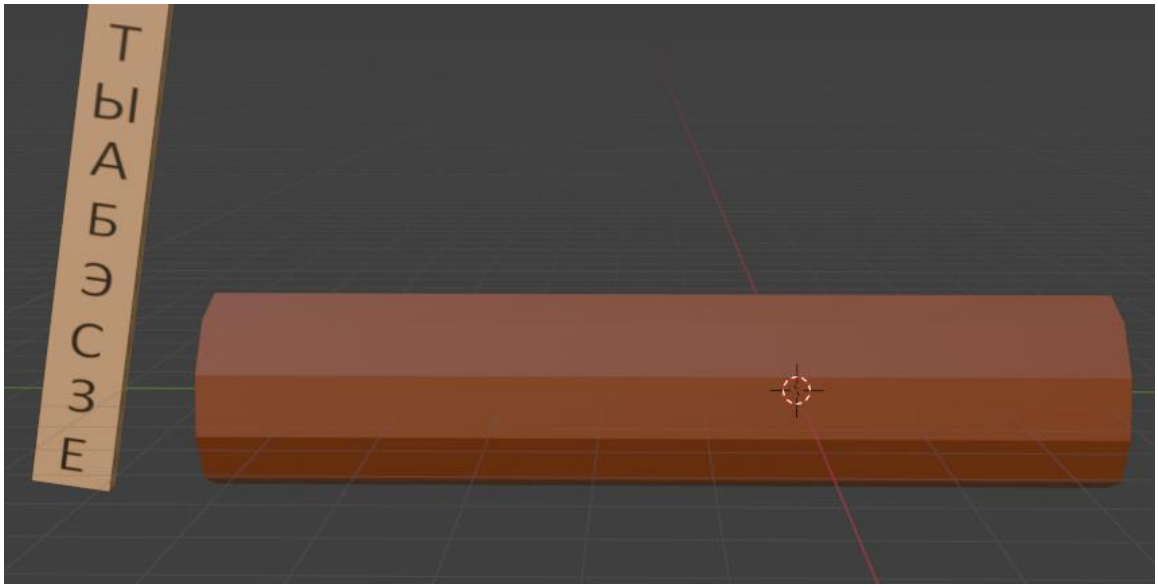


Рисунок 41 – Лента и стержень

Следующим шагом является создание «костей» для ленты. В контексте Blender «кости» представляют собой элементы, используемые для управления деформацией объектов в анимации. Каждая «кость» будет привязана к отдельному сегменту ленты, что обеспечит возможность анимации ее движения, рисунок 42. После создания костей необходимо открыть окно TimeLine, где будет производиться настройка анимации.

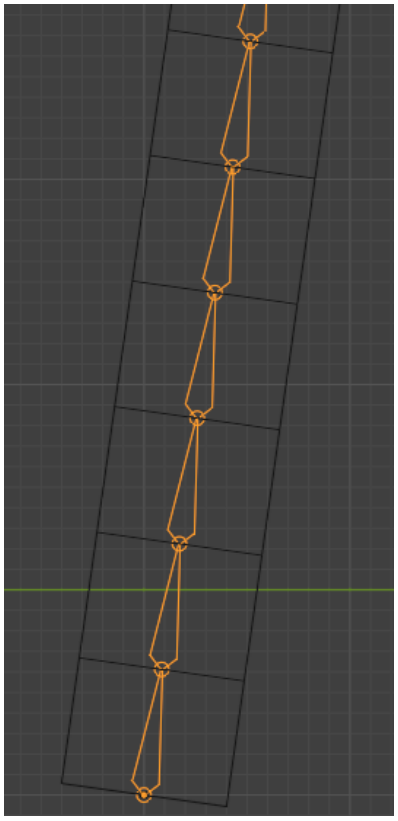


Рисунок 42 – Созданные кости для ленты (выделены оранжевым цветом)

В окне TimeLine используются Keyframe (Ключевые кадры), которые позволяют фиксировать состояние объекта в определенный момент времени. Keyframe – это точка на временной шкале, в которой задаются параметры объекта, такие как положение, вращение и масштаб. Для создания плавной анимации необходимо расставить Keyframe на одинаковых промежутках времени. Запускаем запись анимации, и при перемещении костей происходит автоматическая фиксация изменений в виде Keyframe. Таким образом, при каждом сдвиге «костей» лента будет наматываться на стержень.

Создание анимации в Unity происходит по схожему алгоритму, в котором также используются Keyframe для изменения параметров объектов.

После завершения работы над анимацией в Blender, следующим шагом будет загрузка созданной анимации в Unity. Для этого необходимо экспортировать модель в формате FBX, при этом важно отметить опцию «Bake Animation». Эта опция позволяет сохранить все анимационные данные, созданные в Blender, в одном файле, что упрощает их дальнейшее использование в Unity.

Когда модель с анимацией добавляется в проект Unity, сама анимация будет находиться в отдельном файле, который автоматически создается при импорте.

Для запуска анимации в Unity необходимо создать Animation Controller. Animation Controller – это специальный компонент, который управляет различными состояниями анимации объекта. Он позволяет организовать и переключать между разными анимациями в зависимости от условий, заданных в приложении.

Внутри Animation Controller можно создать триггер – это специальный параметр, который запускает определенные анимации при его активации. Триггеры позволяют контролировать, когда именно должна воспроизводиться анимация, например, при выполнении определенного действия.

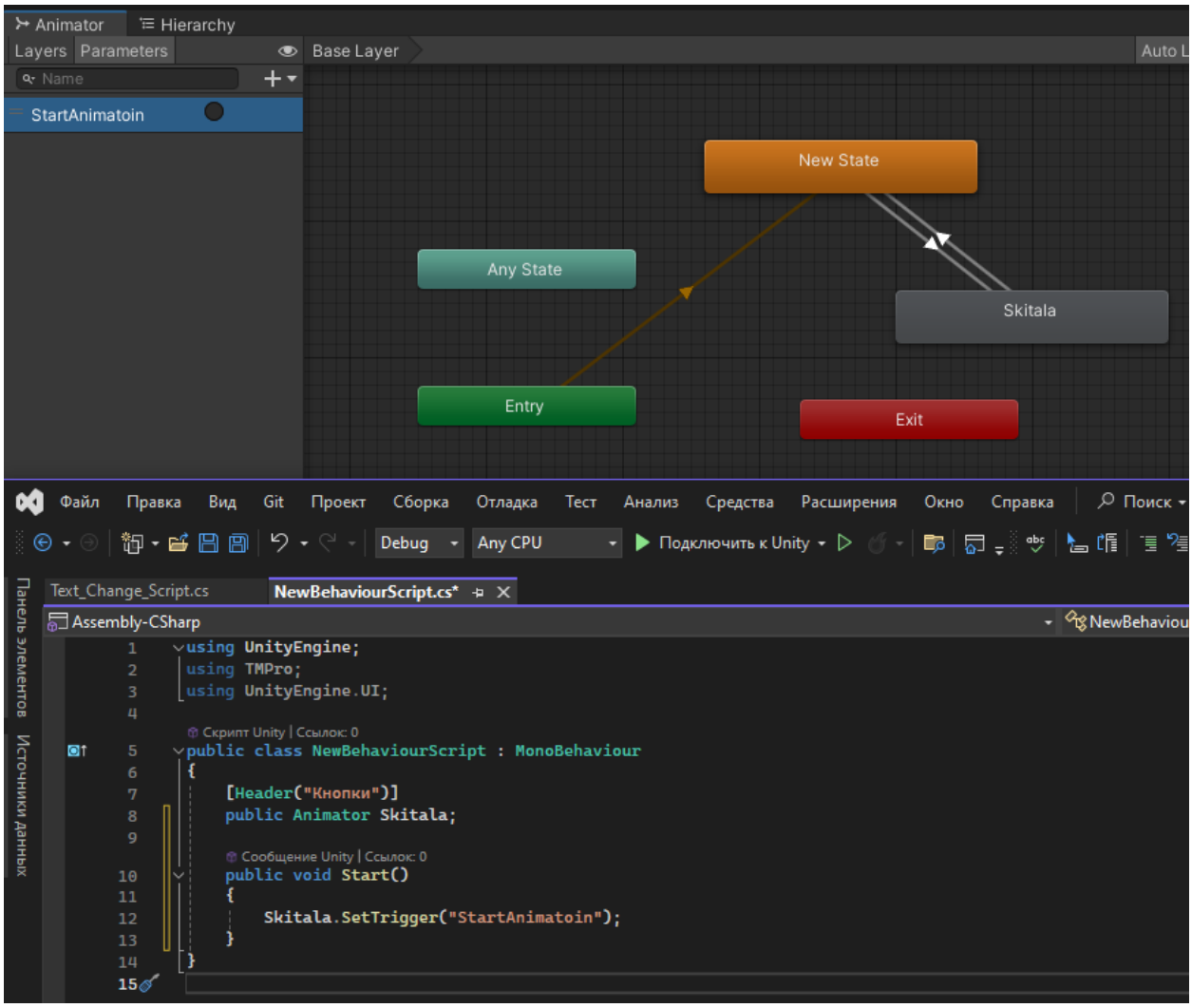


Рисунок 43 – Реализация запуска анимации

Для запуска анимации с использованием триггера необходимо написать C# скрипт, который будет ссылаться на объект типа Animator. В этом скрипте можно использовать метод SetTrigger, чтобы активировать триггер и запустить соответствующую анимацию. Реализация представлена на рисунке 43.

Таким образом, используя возможности Blender и Unity, была создана визуализация криптографических методов защиты информации. Эта интеграция позволяет не только демонстрировать работу различных шифровальных алгоритмов, но и обеспечивает интерактивный подход к обучению криптографии, делая его более доступным и интересным для широкой аудитории.

7 РАЗМЕЩЕНИЕ ПРОГРАММЫ НА СЕРВИСАХ

Для достижения максимальной охвата доступности необходимо разместить программу в открытом доступе. Как вариант можно разместить ссылку на архив с программой на специально созданном для этого сайте, но стоит учитывать, что простое размещение программы в виде архива на сайте может вызвать ряд проблем, связанных с безопасностью и управлением данными.

Одной из основных трудностей является необходимость защиты сайта от потенциальных DDoS-атак. Такие атаки могут привести к недоступности ресурса, что негативно скажется на пользователях, стремящихся получить доступ к программе. Кроме того, размещение архива на файлообменных платформах несет в себе риски, связанные с возможностью случайного удаления файла или его недоступности по другим причинам.

К тому же физическая установка программного обеспечения на локальный компьютер также может оказаться не самым оптимальным вариантом для всех категорий пользователей. Необходимость наличия свободного места на жестком диске и соответствующих технических характеристик ПК ограничивает доступность программы для части целевой аудитории.

В связи с вышеизложенными обстоятельствами целесообразно рассмотреть возможность размещения программной визуализации на стороннем сервисе. Такой подход позволяет делегировать ответственность за защиту данных и безопасность приложения на платформу. Пользователи получают возможность легко и быстро получить доступ к программе без необходимости установки и настройки, что значительно упрощает процесс взаимодействия с приложением.

Существует множество платформ, которые могут быть использованы для размещения программной визуализации.

7.1 Обзор существующих сервисов

Рассмотрим несколько популярных сервисов для размещения программной визуализации криптографических методов, а именно Steam, VK Play и Яндекс.Игры. Каждый из этих сервисов имеет свои особенности и преимущества,

которые необходимо учитывать при выборе наиболее подходящей платформы.

Steam – это зарубежный онлайн-сервис, который предоставляет игрокам со всего мира возможность покупать игры, заводить друзей, вступать в сообщества и многое другое, рисунок 44.

Рисунок 44 – Логотип Steam

Он стал одной из самых популярных платформ для дистрибуции игр благодаря своему обширному функционалу и большому количеству пользователей. В Steam существует инструмент под названием Steamworks, который позволяет разработчикам загружать свои игры или приложения в магазин.

Несмотря на все преимущества этого сервиса, для нашего проекта он не подходит.

Основная причина заключается в том, что для работы с Steam пользователю необходимо установить лаунчер. Хотя данный сервис очень популярен, и многие пользователи имеют его на своих компьютерах, необходимость установки лаунчера может отпугнуть часть аудитории. Кроме того, использование Steam не исключает физическую установку самой программной визуализации, что означает, что как наша программа, так и сам лаунчер Steam будут занимать место на жестком диске пользователя.

Также стоит отметить, что для того, чтобы выложить свой проект в Steam, разработчику необходимо заплатить 100 долларов. Это довольно высокая сумма, и на данный момент из России осуществить такую транзакцию невозможно.

Учитывая все эти факторы, использование Steam для нашего проекта представляется нецелесообразным.

VK Play – это отечественный аналог зарубежных платформ, таких как Steam, который также предоставляет пользователям возможность покупать и

играть в игры, рисунок 45.

Рисунок 45 – Логотип VK Play

Основным преимуществом VK Play является отсутствие необходимости в установке лаунчера, что позволяет игрокам запускать игры и приложения непосредственно через браузер. Это значительно упрощает процесс доступа к контенту и делает его более удобным для пользователей, которые могут не желать устанавливать дополнительные программы на свои устройства.

Еще одним важным плюсом VK Play является возможность бесплатной загрузки своих программ и игр. Это делает платформу более доступной для разработчиков.

Однако у VK Play есть и свои недостатки. Одним из них является не самая высокая популярность платформы среди пользователей. Кроме того, стоит отметить, что на платформе размещаются только игры, а программы, особенно обучающие, не поддерживаются. Это ограничивает возможности использования VK Play для проектов, которые не относятся к игровой категории.

Таким образом, несмотря на ряд преимуществ, VK Play может не подойти для размещения программной визуализации криптографических методов.

Яндекс Игры – это отечественная онлайн платформа, предоставляющая доступ к играм и приложениям непосредственно через браузер, рисунок 46.

Рисунок 46 – Логотип Яндекс Игр

Одним из главных преимуществ данной платформы является обширная

база пользователей, что создает активное сообщество, где игроки могут оставлять отзывы и делиться своими впечатлениями. Это позволяет разработчикам получать обратную связь и улучшать свои проекты.

Платформа предлагает разнообразные категории программ, включая обучающие приложения, что делает ее универсальным инструментом для пользователей с различными интересами. Яндекс Игры не требуют установки дополнительных лаунчеров, что значительно упрощает процесс доступа: пользователю достаточно стабильного интернет-подключения для запуска игр и приложений.

Еще одним плюсом является возможность бесплатной загрузки своих программ на платформу.

Однако стоит отметить и некоторые недостатки. Одним из них является реклама, которая появляется рядом с окном приложения. Это может отвлекать пользователей и создавать определенные неудобства во время игры или работы с приложением.

Тем не менее, несмотря на наличие рекламы, большое количество преимуществ делает Яндекс Игры идеальным выбором для размещения программной визуализации. Платформа предоставляет отличные условия для взаимодействия с пользователями и способствует развитию проектов различной направленности.

7.2 Размещение на платформе Яндекс.Игры

Размещение программной визуализации на платформе Яндекс.Игры требует выполнения ряда последовательных шагов. В первую очередь необходимо собрать проект, разработанный в среде Unity, для целевой платформы WebGL. WebGL представляет собой JavaScript API, который позволяет рендерить 2D и 3D графику непосредственно в веб-браузерах без необходимости использования дополнительных плагинов.

Для успешного взаимодействия с Яндекс.Играми также требуется использование специального плагина PluginYG. Этот инструмент предназначен для оптимизации процесса интеграции приложения с функционалом платформы.

После завершения сборки проекта и настройки всех необходимых компонентов, полученный билд необходимо упаковать в ZIP архив. Этот архив станет

основой для загрузки на консоль Яндекс.Игр, которая представляет собой аккаунт разработчика и служит центральной точкой для управления всеми аспектами размещения приложения. Консоль предоставляет инструменты для мониторинга статистики, управления обновлениями и взаимодействия с пользователями.

Следующим этапом является добавление информации о приложении. Данная информация включает в себя такие важные параметры, как язык, возрастной рейтинг, категории, теги, ключевые слова, название, описание, а также визуальные элементы, такие как иконка, обложка и скриншоты, рисунок 47. Эти данные играют ключевую роль в представлении приложения пользователям и влияют на его восприятие.

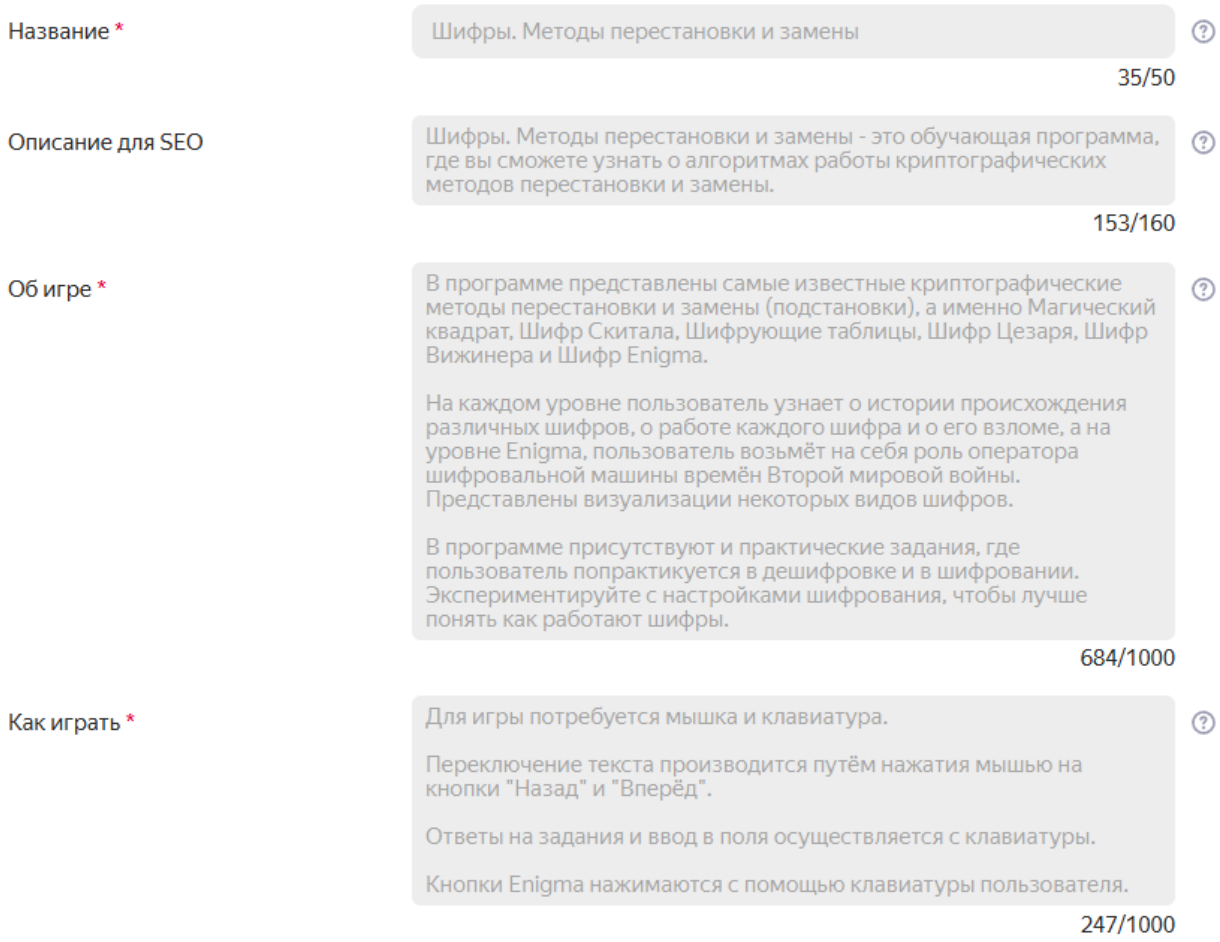


Рисунок 47 – Описание проекта на Консоль Яндекс.Игры

После завершения всех подготовительных этапов необходимо отправить собранную информацию на модерацию. После успешного прохождения модерации приложение становится доступным для пользователей Яндекс.Игр.

В Приложении В представлен результат размещения программной визуализации на онлайн платформу, а именно QR-код для перехода на страницу Яндекс.Игр, рисунок В.1, работа программы на десктопном устройстве, рисунок В.2, и экран с информацией о программе на мобильном устройстве, рисунок В.3.

8 БЕЗОПАСНОСТЬ И ЭКОЛОГИЧНОСТЬ

Работа с программной визуализацией и ее технической поддержкой требует наличия рабочих мест, что, в свою очередь, подразумевает организацию соответствующих помещений. Важно организовать эти пространства в соответствии с установленными правилами и стандартами (СанПин), а также обеспечить здоровье сотрудников, работающих на персональных компьютерах, путем разработки рекомендаций и комплекса физических упражнений.

Безопасность следует рассматривать не только как отсутствие угроз, но и как динамическое состояние, при котором потенциальные риски для жизни и здоровья человека сводятся к минимально возможному уровню с учетом вероятности их возникновения и ограничения зоны воздействия. Эти угрозы, возникающие в результате человеческой деятельности, обладают характерными особенностями: они могут оставаться скрытыми до тех пор, пока не будут созданы определенные условия, а их влияние ограничивается конкретной областью, непосредственно связанной с источником опасности.

Безопасность жизнедеятельности (БЖД) – это комплекс мер, направленных на обеспечение безопасности человека в его среде обитания, сохранение его здоровья, разработку методов и средств защиты, снижение вредного воздействия до приемлемых значений, разработку мер по ограничению ущерба при ликвидации чрезвычайных ситуаций в мирных и военных условиях.

Изучение и решение проблем, связанных с обеспечением здоровой и безопасной рабочей среды, является одной из наиболее важных задач в разработке новых технологий и производственных систем. Изучение и выявление возможных причин несчастных случаев на производстве, профессиональных заболеваний, несчастных случаев, взрывов и пожаров, а также разработка мер и требований по устранению этих причин обеспечивают безопасные и благоприятные условия для человеческого труда. Комфортные и безопасные условия труда являются одним из основных факторов, влияющих на производительность труда сотрудников, поддерживающих работу информационных систем. Работа

сотрудников напрямую связана с компьютером и, следовательно, с вредным дополнительным воздействием целой группы факторов, которые значительно снижают производительность их работы.

8.1 Безопасность

Производственная среда, сочетающая природные и профессиональные факторы, может оказывать негативное влияние на здоровье и работоспособность человека. Вредные факторы ухудшают здоровье, приводя к профессиональным заболеваниям. Опасные факторы при определенных условиях могут вызвать острые нарушения здоровья или даже смерть. Поэтому важно соблюдать установленные нормы в производственной среде, чтобы предотвратить неблагоприятные последствия для здоровья и работоспособности сотрудников.

Условия работы зависят от рабочей обстановки и характера труда. Организация труда и взаимоотношения в коллективах могут негативно влиять на здоровье и производительность. Производственные вредности – это все факторы, которые могут привести к уменьшению работоспособности, возникновению острых или хронических отравлений и заболеваний, а также увеличению заболеваемости с временной потерей трудоспособности или другими негативными последствиями.

К вредным (или неблагоприятным) факторам относятся:

– физические (статические и динамические) перегрузки – подъем и перенос тяжестей, неудобное положение тела, длительное давление на кожу, суставы, мышцы и кости;

– физиологические перегрузки – недостаточная двигательная активность (гипокинезия);

– нервно-психические перегрузки – умственное перенапряжение, эмоциональные перегрузки, перенапряжение анализаторов.

8.1.1 Опасные и вредные факторы на рабочем месте

При работе с компьютером необходимо соблюдать требования норм.

Согласно ГОСТ 12.0.003-2015, опасные и вредные факторы при работе с ПК:

– электростатическое поле;

– электромагнитное излучение;

– опасность поражения электрическим током;

– повышенная или более низкая температура воздуха в рабочей зоне;

– выброс ряда химических веществ в воздух рабочей зоны;

– повышенная или низкая влажность;

– недостаток или недостаточный естественный свет;

– недостаточное искусственное освещение рабочей зоны;

– усталость глаз;

– монотонность рабочего процесса;

– нервная и эмоциональная перегрузка;

– повышенный уровень шума.

Чтобы предотвратить или уменьшить влияние различных вредных факторов на пользователя ПК, были сформулированы требования к помещениям, освещению, уровню шума, организации рабочего места, а также разработаны рекомендации пользователю ПЭВМ.

8.1.2 Организация рабочего места

Сидячая рабочая позиция предоставляет множество преимуществ по сравнению со стоячей: центр тяжести заметно понижается над опорной точкой, улучшая тем самым стабильность, а также существенно уменьшаются энергетические расходы для её поддержания, делая её менее истощающей.

Нервно-психические перегрузки также случаются, потому что сотрудники занимаются умственным трудом.

Интеллектуальный труд связан с обработкой больших объемов разнообразной информации, что требует активации памяти и внимания, а также может

вызывать частые стрессовые ситуации. Физическая нагрузка при этом обычно невелика. Гипокинезия, характерная для умственного труда, может привести к снижению реактивности организма и увеличению эмоционального напряжения, что в свою очередь может способствовать развитию сердечно-сосудистых заболеваний. Интенсивная умственная работа негативно воздействует на

тонус гладкой мускулатуры внутренних органов и кровеносных сосудов, особенно мозга и сердца. Обилие сигналов от периферии и внутренних органов также влияет на мыслительную деятельность, которая тесно связана с работой органов чувств, особенно зрения и слуха, и более продуктивна в условиях тишины.

Рабочее место пользователя – это область, в которой находится сотрудник и его средства работы, определенная на основе технических и эргономических стандартов и оснащенная техническими и другими средствами, необходимыми сотруднику для выполнения конкретной задачи, возложенной на него. Рабочее место – это совокупность факторов окружающей среды, включая вредные факторы. Вредный производственный фактор – это фактор, воздействие которого на человека при определенных условиях может привести к заболеваниям и нетрудоспособности. В соответствии с требованиями ГОСТ 12.2.032-78 «Система стандартов безопасности труда. Рабочее место при выполнении работ сидя», предъявляются следующие требования:

– высота рабочей поверхности стола для взрослых пользователей должна быть отрегулирована от 680 до 800 мм; если это невозможно, рабочая поверхность должна составлять 725 мм;

– рабочий стол должен иметь пространство для ног высотой не менее 600 мм, шириной – не менее 500 мм, глубиной на уровне колен – не менее 450 мм и на уровне вытянутых ног – не менее 650 мм;

– сиденье должно быть не менее 400 мм в ширину и глубину, иметь закругленный передний край и регулироваться в пределах 400 мм;

– 550 мм и угол наклона вперед до 15 градусов и назад до 5 градусов угол наклона спинки в вертикальной плоскости должен составлять ± 30 градусов;

– фиксированные или съемные подлокотники сиденья должны иметь длину не менее 250 мм и ширину 50-70 мм, регулироваться над сиденьем в пределах 230 ± 30 мм, а внутреннее расстояние между подлокотниками должно быть в пределах 350-500 мм;

– рабочее место пользователя ПК должно быть оснащено подставкой для

ног шириной не менее 300 мм, глубиной не менее 400 мм, регулировкой высоты 150 мм и углом наклона до 20 градусов;

– клавиатура должна быть размещена на поверхности стола на расстоянии 100-300 мм от края, обращенного к пользователю, или на специальной рабочей поверхности с регулируемой высотой, отделенной от основной столешницы.

На рисунке 48 представлено рекомендуемое размещение пользователя ПЭВМ.

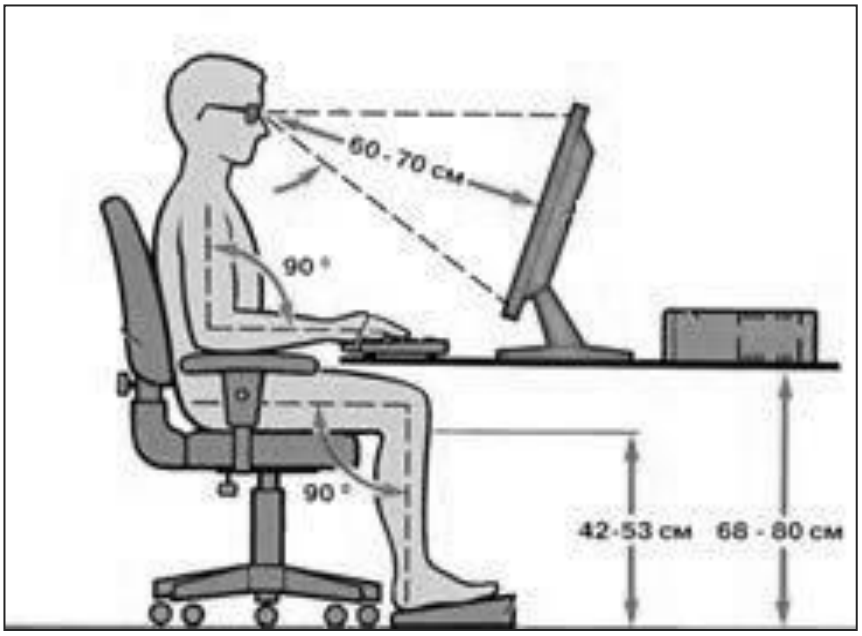


Рисунок 48 – Рекомендуемое размещение пользователя ПЭВМ

Время регламентированных перерывов во время работы с ПЭВМ зависит от категории трудовой деятельности и уровня нагрузки за смену, и определяется суммарным временем, приведенных в таблице 1.

Таблица 1 – Суммарное время регламентированных перерывов в зависимости от продолжительности работы, вида и категории трудовой деятельности с ПЭВМ

Категория работы с ПЭВМ	Уровень нагрузки за рабочую смену при видах работ с ПЭВМ			Суммарное время регламентированных перерывов, мин.	
	группа А, количество знаков	группа Б, количество знаков	группа В, ч	при 8-часовой смене	при 12-часовой смене
I	до 20 000	до 15 000	до 2	50	80
II	до 40 000	до 30 000	до 4	70	110
III	до 60 000	до 40 000	до 6	90	140

Для предотвращения преждевременной утомляемости пользователей ПЭВМ рекомендуется чередовать работу с использованием ПЭВМ и без него для организации рабочей смены.

Для уменьшения негативных ощущений при работе с ПЭВМ важно применять индивидуальный подход и ограничивать время использования компьютера, даже при соблюдении всех необходимых гигиенических и эргономических правил.

Во время перерывов, чтобы снизить нервное напряжение, устранить утомление глаз и предотвратить физическую усталость, полезно делать упражнения для избавления от статической усталости.

8.1.4 Освещение

Освещение является одним из наиболее важных требований к помещениям с ПК. Правильное освещение повышает производительность труда, поскольку снижает нагрузку на зрение. С другой стороны, плохое освещение приводит к быстрой усталости, затуханию при работе на ПК, бликам и раздражительности при чрезмерной яркости.

Типы освещения следующие:

– естественное;

– искусственное;

– смешанное;

– аварийное.

Естественный свет должен присутствовать в каждой комнате, в которой находится персонал. В зависимости от положения он может быть боковым, верхним или комбинированным. При комбинированном освещении недостаточное естественное освещение дополняется искусственным освещением.

Существует искусственное освещение двух систем: общей (однородной и локализованной) и комбинированной. Комнаты оснащены общими системами искусственного освещения, когда светильники находятся в верхней части. Если расстояние между лампами считается равным, свет считается равномерным, если лампа ближе к производственному оборудованию, то освещение считается

локализованным. Такое искусственное освещение называется комбинированным освещением, когда к общему освещению добавляется местное освещение.

Согласно ГОСТ 55710-2013 «Освещение рабочих мест внутри зданий», можно определить основные требования, согласно которым коэффициент естественного освещения должен составлять не менее 1,2 % в районах с устойчивым снежным покровом, а на остальной территории – не менее 1,5 %.

Каждое рабочее место пользователя ПК должно иметь односторонний естественный свет. При недостатке естественного освещения используется искусственный свет. Освещенность рабочего места должна составлять от 300 до 500 люкс. Большинство люминесцентных ламп с высокой светоотдачей используются для достижения этого уровня освещенности. На предприятии количество естественного и искусственного освещения находится в пределах нормы. В ситуациях с недостаточным освещением у каждого сотрудника есть своя настольная лампа, которая компенсирует недостаток света в рабочей зоне.

8.1.5 Шум

На рабочем месте оператора источниками шума являются технические средства (компьютеры, принтеры, вентиляционные устройства), а также внешний шум. Уровень акустического шума на рабочем месте во время работы должен соответствовать требованиям законодательства. ГОСТ 12.1.003-83 ССБТ «Шум. Общие требования безопасности» определяет допустимые значения уровня звукового давления в октавных диапазонах, представленные в таблице 2.

Таблица 2 – Допустимые значения уровней звукового давления

Уровни звукового давления в октавных полосах со среднегеометрическими частотами									Уровни звука, дБ
31,5 Гц	63 Гц	125 Гц	250 Гц	500 Гц	1000 Гц	2000 Гц	4000 Гц	8000 Гц	
86 дБ	71 дБ	61 дБ	54 дБ	49 дБ	45 дБ	42 дБ	40 дБ	38 дБ	50 дБ

8.1.6 Микроклимат

Микроклимат производственных помещений представляет собой

комплекс нормализованных показателей, таких как температура, влажность, тепловое излучение и другие, которые влияют на теплообмен человека и определяют самочувствие, работоспособность, здоровье и производительность труда. Поэтому наиболее важной задачей охраны труда является поддержание микроклимата на рабочем месте в соответствии со стандартами гигиены.

На рабочем месте ПК, который повышает температуру человека, приводит к снижению эффективности и производительности, также повышает температуру всего помещения. В следствии этого, поддержание температуры на требуемом уровне позволит обеспечить безопасность и комфортность при работе за компьютером.

Системы вентиляции используются для поддержания микроклимата в помещении. Система вентиляции – это система изменения воздуха в помещении, предназначенная для поддержания метеорологических параметров помещения и подачи чистого воздуха извне. Для обеспечения наиболее комфортных условий используется естественная система вентиляции, а зимой и летом устанавливается дополнительный кондиционер, чтобы полностью нормализовать микроклиматические параметры на рабочем месте, создать комфортные условия работы.

Кондиционеры используются для поддержания постоянной температуры, влажности и очистки от загрязняющих веществ. Эти системы позволяют нам решить проблему, связанную с задержкой углекислого газа в помещении.

ГОСТ 12.1.005 «Общие санитарно-гигиенические требования к воздуху рабочей зоны» определяет следующие составляющие микроклимата на рабочем месте. Микроклимат любого помещения характеризуется температурой воздуха, влажностью и скоростью передвижения.

В помещениях, оборудованных ПК, компания проводит ежедневную влажную уборку и регулярную вентиляцию после каждого часа рабочего времени.

Температура в помещении – самый важный показатель комфорта. Влажность воздуха напрямую зависит от температуры. Низкие температуры провоцируют тепловыделение человеческого организма, тем самым снижая его защитные функции. Если в помещении установлена некачественная система

отопления, люди постоянно подвергаются переохлаждению, частым простудным заболеваниям, инфекционным заболеваниям и т.д.

Очень высокая температура в помещении (выше 27 градусов) приносит не меньше проблем. При борьбе с жарой организм удаляет соль из организма. Эта ситуация также чревата снижением иммунитета, нарушением водно-солевого баланса, который регулирует работу многих систем в организме. Температура на рабочем месте в холодный период должна быть в пределах 20-30 ^{0}C, в теплое время года 20-25 ^{0}C. Относительная влажность воздуха должна быть в пределах 60-40 %, а скорость воздуха не должна превышать 0,2 м/с.

8.1.7 Графический интерфейс приложения

Разрабатываемый программная визуализация имеет интерфейс, который должен соответствовать требованиям ГОСТ Р 50948-2001. «Средства отображения информации индивидуального пользования. Общие эргономические требования и требования безопасности».

Чтобы точно прочитать информацию и обеспечить комфортную среду для восприятия, дисплеи должны работать с сочетанием яркости и контрастности, окружающего света, размера угла и угла обзора экрана, которые находятся в оптимальных или максимально допустимых (для кратковременной работы) диапазонах.

Если необходимо поменять параметры цвета, прикладная программа должна предложить набор цветов по умолчанию, соответствующий требованиям этого стандарта. Если пользователь может изменить цвет, должна быть возможность восстановить набор цветов по умолчанию.

Если необходимо точно определить цвет в буквенно-цифровых строках и в полях ввода, высота знаков должна составлять не менее 20 футов на проектируемом расстоянии наблюдения.

Если пользователь определяет цвет одного изображения (например, числа или символа), размер углового изображения должен быть не менее 30' на расстоянии проекта наблюдения.

Следует избегать применения насыщенного синего цвета для

изображений, имеющих угловой размер менее 2'.

Для чтения текста, буквенно-цифровых символов и символов с отрицательной полярностью не используйте синий и красный спектры на темном фоне и красный спектр на синем фоне.

Для чтения текста, буквенно-цифровых символов и символов с положительной полярностью не используйте синий спектр с красным фоном.

Насыщенные экстремальные цвета видимого спектра приводят к нежелательным эффектам в глубине отображаемого пространства и не должны использоваться для изображений, требующих непрерывного просмотра или чтения.

Чтобы точно распознать и идентифицировать цвет, вам нужно будет использовать либо цветное изображение переднего плана на ахроматическом фоне, либо ахроматическое изображение переднего плана на цветном фоне.

Количество цветов, отображаемых на экране одновременно, должно быть минимальным. Чтобы точно определить цвет, каждый набор цветов по умолчанию должен содержать не более 11 цветов.

Если необходимо выполнить быстрый поиск на основе распознавания цвета, вы должны использовать максимум 6 разных цветов.

Если необходимо получить доступ к настройкам цвета из памяти компьютера, вы должны использовать не более 6 разных цветов.

8.2 Экологичность

Экология является научным фундаментом для защиты окружающей среды, представляющей собой область знаний в этой сфере, которая сосредоточена на разработке мер по обеспечению гармоничного взаимодействия между человеческой деятельностью и природной средой. Эти действия призваны сохранять и восстанавливать природные ресурсы, использовать их рационально, а также предотвращать негативное воздействие хозяйственной деятельности на окружающую среду и здоровье людей.

Хотя разработанный программный продукт не наносит ущерба окружающей среде, но использование технического оборудования – ПК и МФУ, при его работе, может оказывать негативное воздействие на окружающую среду.

Если компьютер или МФУ вышел из строя и не подлежит восстановлению, то следует произвести их утилизацию в соответствии с правилами.

Старые устройства не могут рассматриваться как обычный мусор из-за наличия в их составе вредных веществ, которые представляют опасность для здоровья и экологии.

Согласно законодательству РФ, оргтехника должна быть утилизирована специализированной фирмой, имеющей лицензию на обработку различных классов опасности, так как простое вывоз на свалку незаконно.

За игнорирование утилизации и намеренное загрязнение окружающей среды предусмотрены административные наказания и значительные штрафы для предприятий.

ПК состоит из большого количества компонентов, содержащих токсичные вещества и представляющих угрозу для человека и окружающей среды. К таким веществам относятся:

– ртуть находится в подсветке жидкокристаллических мониторов;

– щелочи, которые находятся в щелочных батареях бесперебойного питания;

– никель и цинк, которые находятся в материнской плате ноутбука и батареях;

– поливинилхлорид находится в проводах, подключенных к электронным устройствам.

Поэтому ПК требует специальных, сложных методов утилизации. Эта мера включает сортировку металлических и неметаллических деталей. Затем металлические детали отправляются для плавки для последующего производства, а неметаллические детали компьютера утилизируются специальным образом.

В настоящее время в ряде отраслей промышленности создаются и внедряются технологии с низким уровнем отходов, но полный перевод ведущих отраслей промышленности на технологии без отходов требует решения большого комплекса очень сложных технологических, конструктивных и организационных задач.

8.3 Чрезвычайные ситуации

Чрезвычайная ситуация – это совокупность событий, характеризующаяся внезапным возникновением обстоятельств, несущих угрозу жизни и здоровью людей, окружающей среде или материальным ценностям в результате аварий, природных катаклизмов, катастроф или других бедствий. Она может повлечь за собой человеческие жертвы, травмы, повреждение экосистем, значительные материальные потери и серьезные нарушения условий существования населения.

В офисе предприятия может возникнуть такая чрезвычайная ситуация, как пожар.

Пожар – это неконтролируемый процесс горения, при котором выделяются тепло и вредные вещества, сопровождающееся уничтожением материальных ценностей и создающее опасность для жизни людей. Пожарная безопасность представляет собой систему профилактических и защитных мероприятий, нацеленных на предупреждение и ликвидацию пожаров.

Источниками возгорания могут служить случайные искры различного происхождения, нагретые тела, перегрев электрических контактов и др.

К основным факторам возникновения пожаров на предприятиях относятся: нарушения режима работы техники, дефекты в электроустановках, недостаточная подготовка к ремонтным работам, самовоспламенение материалов, игнорирование норм пожарной безопасности сотрудниками, перегруженность помещений.

Чаще всего пожары происходят из-за человеческого фактора, когда правила пожарной безопасности не соблюдаются или огонь обращают неосторожно.

8.3.1 Аварийные ситуации

Во время работы могут возникнуть следующие чрезвычайные ситуации:

– обрыв проводов питания;

– неисправность заземления;

– повреждение электрооборудования;

– повреждение инженерных коммуникаций.

Во всех случаях, когда отмечается чрезвычайная ситуация или резкое

ухудшение самочувствия, а также во всех других ситуациях, которые непосредственно угрожают жизни или здоровью человека, необходимо:

– прекратить производство работ;

– если есть раненые, оказать первую помощь;

– при необходимости обеспечьте отключение питания;

– обеспечить открытие аварийных выходов и эвакуацию персонала;

– доложить руководителю о принятых мерах и действовать в соответствии с полученными инструкциями;

– доложите дежурному.

Сотрудник, находящийся рядом с местом аварии, должен оказать пострадавшему первую помощь и сообщить об этом дежурному оперативного отдела, начальнику отдела. Если человек подвергается воздействию напряжения, немедленно отключите питание и освободите его от тока.

8.3.2 Меры пожарной безопасности на рабочих местах

При размещении технологического и другого оборудования необходимо обеспечить наличие путей эвакуации и эвакуационных выходов.

Компьютер должен быть установлен на надежной опоре, которая не позволит ему упасть. Не устанавливайте ПК:

– в нишах мебельных «стенок», в тумбочках и т.п.;

– ближе 1 метра от электронагревателей и от легковоспламеняющихся предметов;

– ближе чем на 0,7 метра от проходов, транспортных путей и эвакуации людей.

Прежде чем запускать компьютер, вам необходимо выполнить следующие действия:

– проверьте место установки ПК и монитора снаружи и убедитесь, что вышеуказанные требования безопасности соблюдены;

– проверьте ПК, шнур питания, вилку и убедитесь, что они в хорошем состоянии, если корпус, шнур питания, вилка и если задняя крышка повреждена, работа ПК запрещена;

– если на ПК и мониторе есть легковоспламеняющиеся предметы и контейнеры с жидкостью, удалите их;

– убедитесь, что вентиляционные отверстия в задней части ПК и монитора не закрыты предметами;

– убедитесь, что рядом с компьютером есть противопожарная ткань или огнетушитель.

Эти меры безопасности при работе на ПК снижают риск его возгорания.

ЗАКЛЮЧЕНИЕ

При выполнении выпускной квалификационной работы был изучен объект исследования – криптография, в частности криптографические методы перестановки и замены, а также предмет исследования – визуализация криптографических методов.

Был проведён обзор существующих методов решения аналогичных типовых задач, были исследованы существующие программные продукты и образовательные ресурсы, их сильные и слабые стороны.

В качестве программных средств реализации были выбраны Unity и Blender. После чего была спроектирована и разработана программной визуализации криптографических методов перестановки и замены.

В результате был получен программный модуль, представляющий собой интерактивную лекцию, дополненную теоретической информацией, иллюстрациями и визуализацией криптографических методов, предназначенный для обучения пользователей основам криптографии в игровой форме. Ключевым аспектом разработки программной визуализации являлось включение практических заданий на шифрование и дешифрование.

Для достижения максимальной доступности, программа была размещена в открытом доступе на онлайн-платформе Яндекс.Игры, благодаря чему пользователи получают возможность легко и быстро получить доступ к программе через браузер, без необходимости установки и настройки.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1 Бонд, Д. Г. Unity и C#. Геймдев от идеи до реализации. 2-е изд./ Д.Г. Бонд – Питер, 2023. – 928. – ISNB 978-5-4461-0715-5.

2 Васильева, И. Н. Криптографические методы защиты информации : учебник и практикум для вузов / И. Н. Васильева. – Москва : Издательство Юрайт, 2025. – 310 с. – (Высшее образование). – ISBN 978-5-534-02883-6. – Текст : электронный // Образовательная платформа Юрайт [сайт]. – Режим доступа : https://urait.ru/bcode/560977. – 10.06.2025.

3 Вульф, А. Криптография. Основы практического шифрования и криптографии/ А. Вульф. – Екатеринбург : Издательство Ridero, – 2023. – 98. – ISNB 978-5-0060-1523-4.

4 ГОСТ 12.0.003-2015. Опасные и вредные производственные факторы. Классификация. – Взамен ГОСТ 12.0.003-74 ; Дата актуализации: 2021-01-01. – Москва : Стандартинформ, 2016 – 16 с.

5 ГОСТ 12.2.032-78. Рабочее место при выполнении работ сидя. Общие эргономические требования ; Дата актуализации: 2021–01–01 ; Изд-во стандартов, 2001. – 9 с.

6 ГОСТ 55710-2013. Освещение рабочих мест внутри зданий. Нормы и методы измерений. ; Дата актуализации: 2021–01–01 – Москва : Стандартинформ, 2016 – 19 с.

7 ГОСТ 12.1.003-83. Шум. Общие требования безопасности. – Взамен ГОСТ 12.1.003-76 ; Дата актуализации: 2023–07–01 ; Изд-во стандартов, 2008 – 13 с.

8 ГОСТ 12.1.005. Общие санитарно-гигиенические требования к воздуху рабочей зоны. – Взамен ГОСТ 12.1.005-76 ; Дата актуализации: 2021–01–01 01 – Москва : Стандартинформ, 2005 – 49 с.

9 ГОСТ Р 50948-2001. Средства отображения информации индивидуального пользования. Общие эргономические требования и требования

безопасности. – Взамен ГОСТ Р 50948-96 ; Дата актуализации: 2023–07–01 ; – Москва : Госстандарт России, 2002 – 11 с.

10 Гудимов В. Н. Программная визуализация криптографических методов защиты [Электронный ресурс] / В. Н. Гудимов, С. Г. Самохвалова; Хроники Объединенного фонда электронных ресурсов «Наука и образование». – 2025. – №03(190). – Режим доступа : https://ofernio.ru/portal/newspaper/ofernio/2025/3.pdf. – 10.06.2025.

11 Гудимов В. Н. Программная визуализация криптографических методов защиты [Электронный ресурс] / В. Н. Гудимов, С. Г. Самохвалова; Навигатор в мире науки и образования. – 2025. – №01(66). – Режим доступа : https://ofernio.ru/portal/navigator/files/navigator_2025_1_66.pdf. – 10.06.2025.

12 Гудимов В. Н. Проектирование программной визуализации криптографических методов защиты [Электронный ресурс] / В. Н. Гудимов, С. Г. Самохвалова; Вестник АмГУ. – 2024. – №107. – Режим доступа : https://vestnik.amursu.ru/wpcontent/uploads/2024/12/n107_139-146.pdf. – 10.06.2025.

13 Гудимов В. Н. Разработка программы для знакомства с криптографией [Электронный ресурс] / В. Н. Гудимов, С. Г. Самохвалова; Перспективы развития науки и образования. – 2025. – Режим доступа : http://science-peace.ru/files/PRNO_2025.pdf. –10.06.2025.

14 Гудимов В. Н. Программная реализация криптографических методов защиты: перестановки и замены [Электронный ресурс] / В. Н. Гудимов, С. Г. Самохвалова; Флагман науки: научный журнал. – 2025. – СПб., Изд. ГНИИ «Нацразвитие» – № 4(27). – Режим доступа : https://flagmannauki.ru/files/427-Gudimov_Viktor_Nikolaevich_3393_2.pdf. – 10.06.2025.

15 Денисов, Д. Разработка игры на Unity. С нуля до публикации/ Д. Денисов. – SelfPub, 2021. – 177. – ISNB 978-5-0437-2633-9.

16 Добавить игру | Руководство разработчика: [Электронный ресурс]., – 2025. – Режим доступа : https://yandex.ru/dev/games/doc/ru/console/add-new-game.

– 10.06.2025.

17 Жадаев, А. Нумерология на компьютере. Расчет судьбы по методике Пифагора/ А. Жадаев. – ЛитРес, 2022. – 66. – ISNB 978-5-0403-4924-1.

18 ИВВ. Тайны Шифрования: Взгляд в Мир Криптографии. Криптография: Магия Шифров/ ИВВ. – Екатеринбург : Издательство Ridero, – 2023. – 19. – ISNB 978-5-0060-5424-0.

19 Кабинет разработчика | VK Play: [Электронный ресурс]., – 2025. – Режим доступа : https://developers.vkplay.ru/welcome. – 10.06.2025.

20 Крючкова, Е., Крючкова, О. Китайская магия (Книга сакральных традиций Китая) / Е. Крючкова, О. Крючкова. – Велигор, 2021. – 240. – ISNB 978-5-0405-4130-0.

21 Лучшие видеоигры | VK Play: [Электронный ресурс]., – 2025. – Режим доступа : https://vkplay.ru/. – 10.06.2025.

22 Маккей С. Шифры цивилизации: Коды, секретные послания и тайные знаки в истории человечества/ С. Маккей. – Альпина Паблишер, – 2022. – 384. – ISNB 978-5-9614-9151-7.

23 Музагафаров, А. Криптография. Шифрованный мир. Азы криптографии и задачи по криптоанализу/ А. Музагафаров. – Екатеринбург : Издательство Ridero, – 2024. – 98. – ISNB 978-5-0060-2725-1.

24 Романенко, Е. Blender. Дизайн интерьеров и архитектуры / Е. Романенко – Питер, 2024. – 176. – ISNB 978-5-4461-2136-6.

25 Саллинс, С. Low Poly 3D Modeling in Blender / С. Саллинс–Packt Publishing, 2024. – 318. – ISNB 978-1-8032-4123-4.

26 Фомичёв, В. Криптография – наука о тайнописи / В. Фомичёв. – ЛитРес, 2021. – 168. – ISNB 978-5-0430-6670-1.

27 Частые вопросы о Яндекс Играх: [Электронный ресурс]., – 2025. – Режим доступа : https://yandex.ru/support/games/ru/. – 10.06.2025.

28 Шифры. Методы перестановки и замены (от GudVik): [Электронный ресурс]., – 2025. – Режим доступа

: https://yandex.ru/games/app/438164?draft=true&lang=ru. –10.06.2025.

29 Steam – превосходная игровая интернет-платформа: [Электронный ресурс]., – 2025. – Режим доступа : https://store.steampowered.com/about/. – 10.06.2025.

30 Steamworks: [Электронный ресурс]., – 2025. – Режим доступа : https://partner.steamgames.com/. – 10.06.2025.

31 PluginYG - Yandex Game integration: [Электронный ресурс]., – 2025. – Режим доступа : https://assetstore.unity.com/packages/add-ons/pluginyg-yandex-game-integration-235877. – 10.06.2025.

32 Technical Details of the Enigma Machine: [Электронный ресурс]., – 2025. – Режим доступа : https://www.ciphermachinesandcryptology.com/en/enigmatech.htm. – 10.06.2025.

ПРИЛОЖЕНИЕ А
Перечень вопросов из опроса

Рисунок А.1 – Вопрос №1

Рисунок А.2 – Вопрос №2

Рисунок А.3 – Вопрос №3

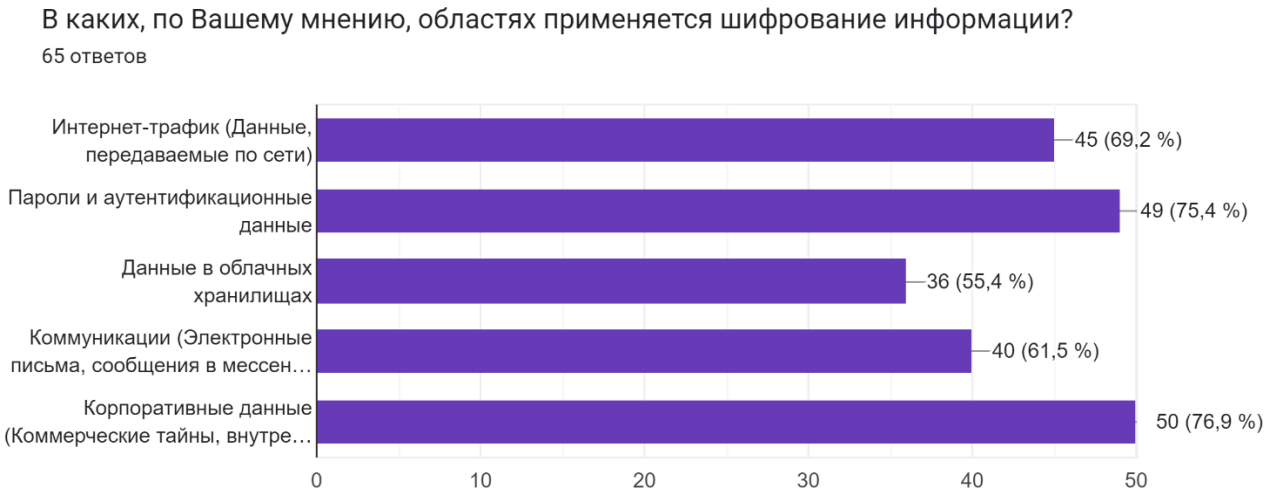


Рисунок А.4 – Вопрос №4

Знаете ли вы что такое кодирование информации? (можете написать в "Другое" определение, если Вы знаете)

65 ответов

Рисунок А.5 – Вопрос №5

Как Вы думаете кодирование информации и шифрование информации это одно и тоже?

64 ответа

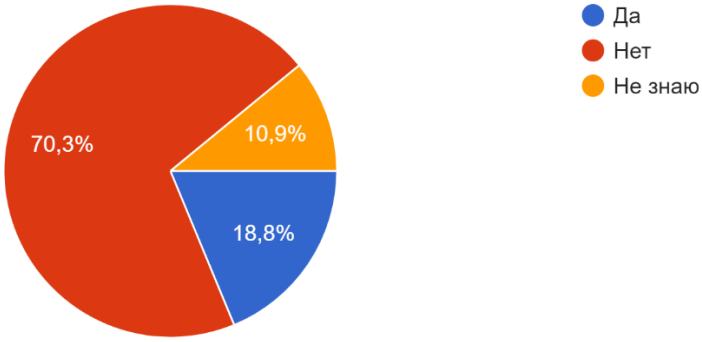


Рисунок А.6 – Вопрос №6

Рисунок А.7 – Вопрос №7

Рисунок А.8 – Вопрос №8

Рисунок А.9 – Вопрос №9

Рисунок А.10 – Вопрос №10

Знаете ли Вы игры или программы в которых визуализирован процесс шифрования одним из вышеперечисленных методов? (если Вы знает...рограмм, пожалуйста, укажите их в "Другое")

65 ответов

Рисунок А.11 – Вопрос №11

ПРИЛОЖЕНИЕ Б
Демонстрация работы программной визуализации

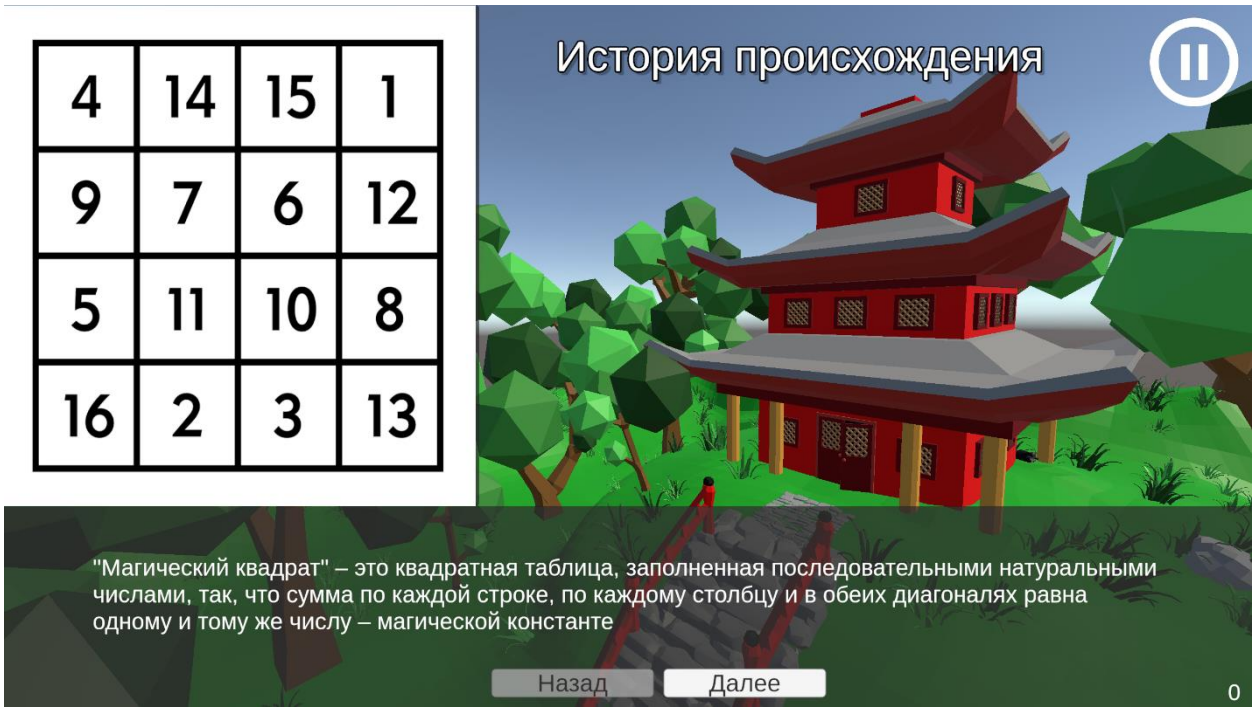


Рисунок Б.1 – Сцена шифра «Магический квадрат»

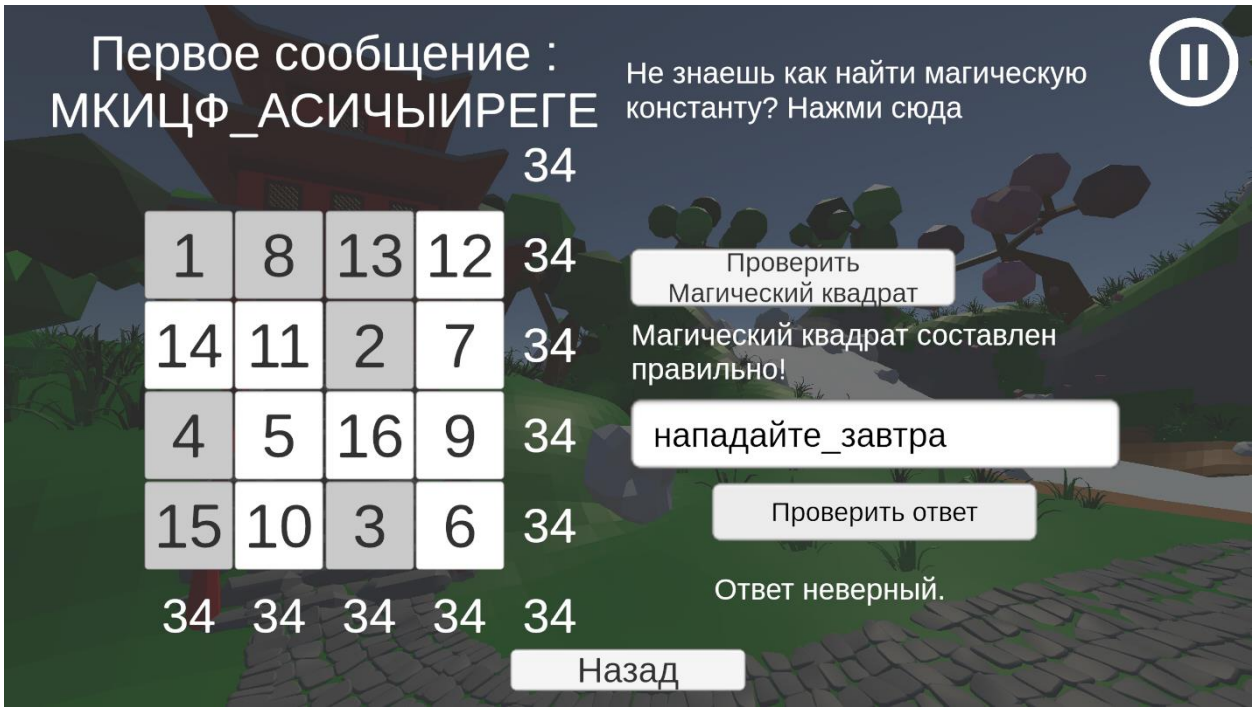


Рисунок Б.2 – Задание на дешифрование сообщения, зашифрованного магическим квадратом

Рисунок Б.3 – Сцена для перестановочных шифров

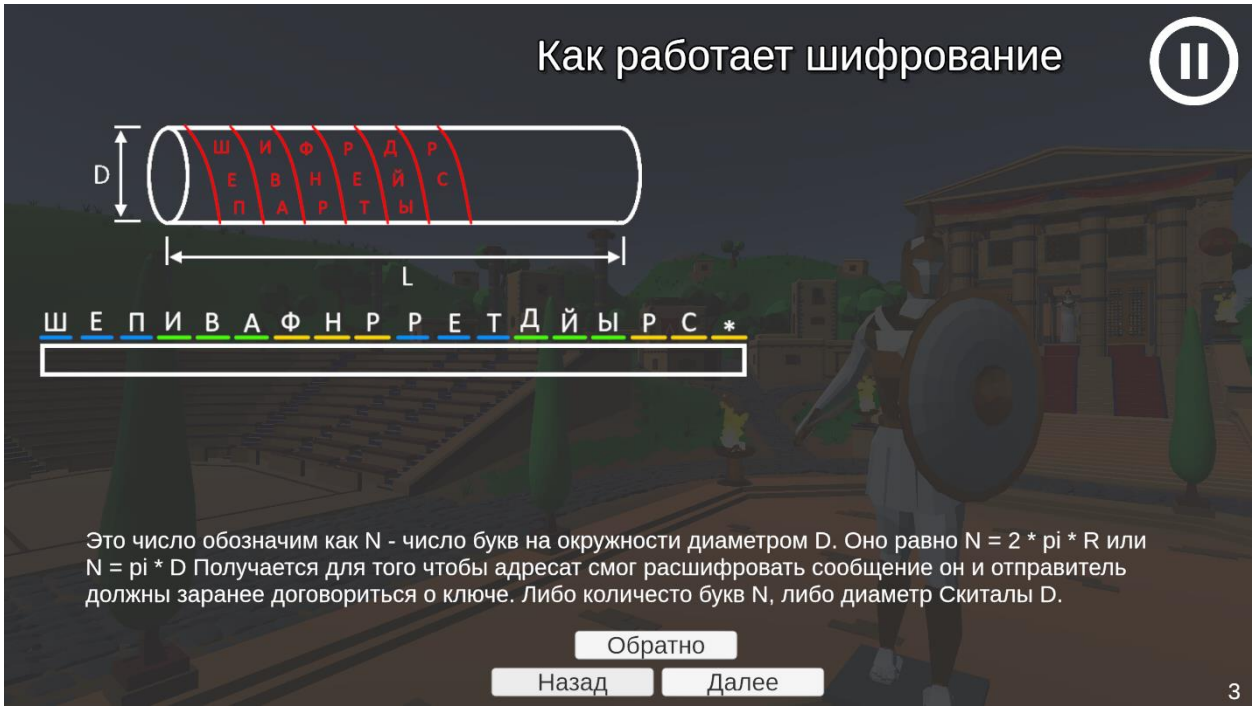


Рисунок Б.4 – Объяснение работы шифра Скитала

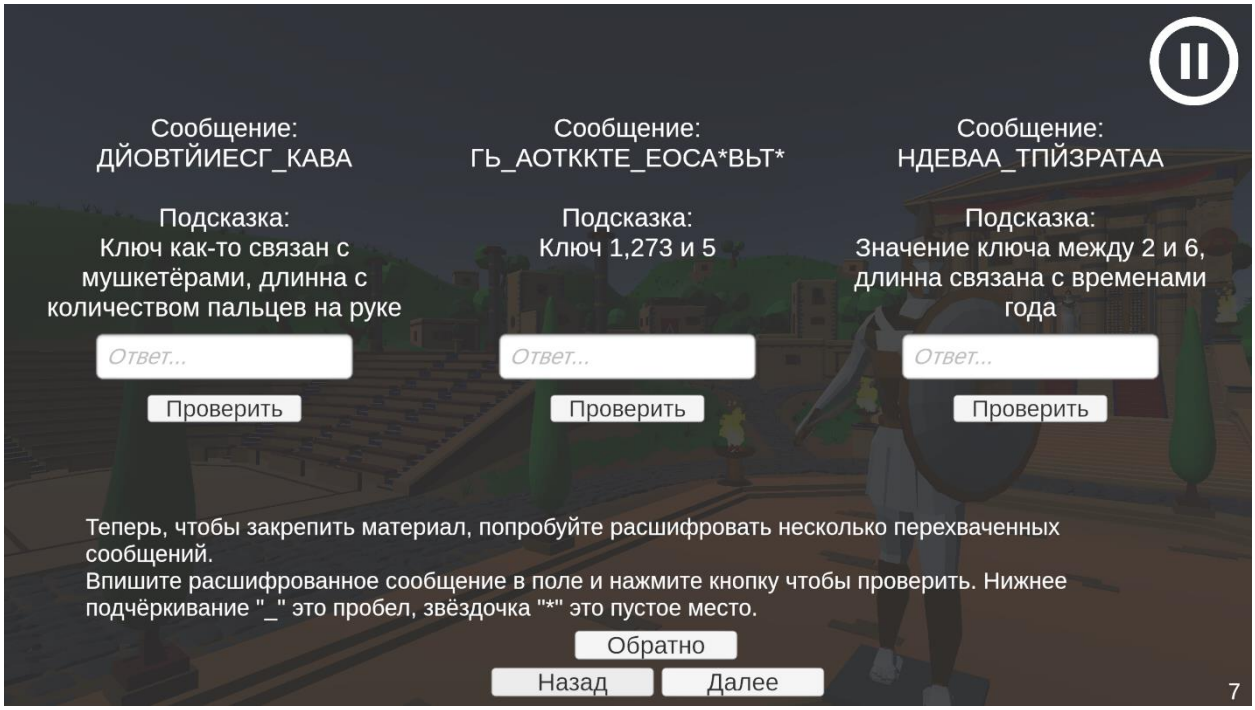


Рисунок Б.5 – Задание на дешифрование сообщений, зашифрованных шифром Скитала

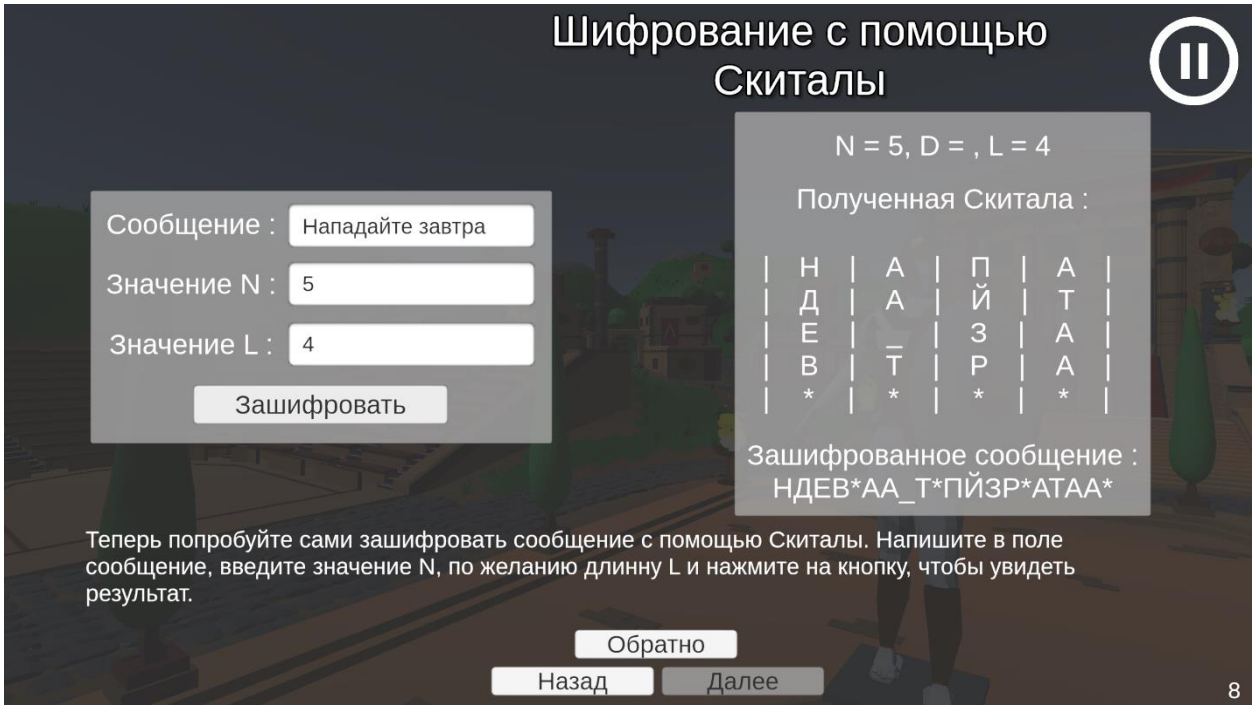


Рисунок Б.6 – Задание на шифрование с помощью шифра Скитала

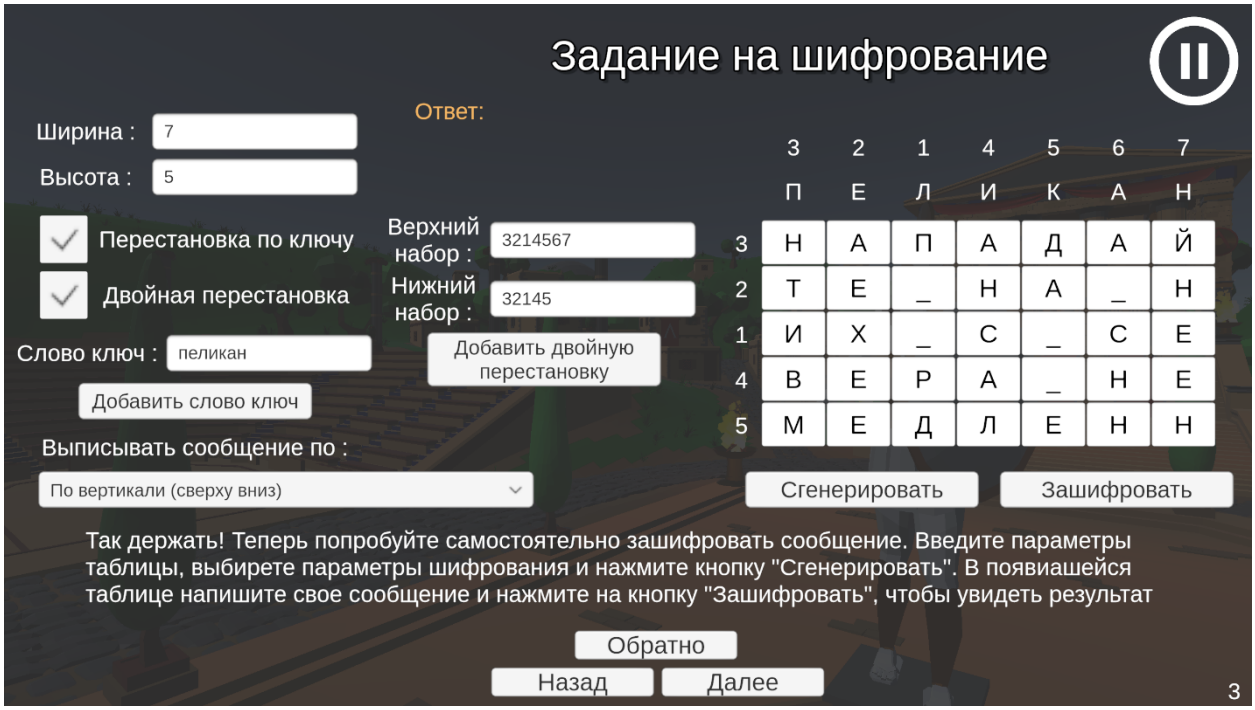


Рисунок Б.7 – Задание на шифрование с помощью Шифрующих таблиц

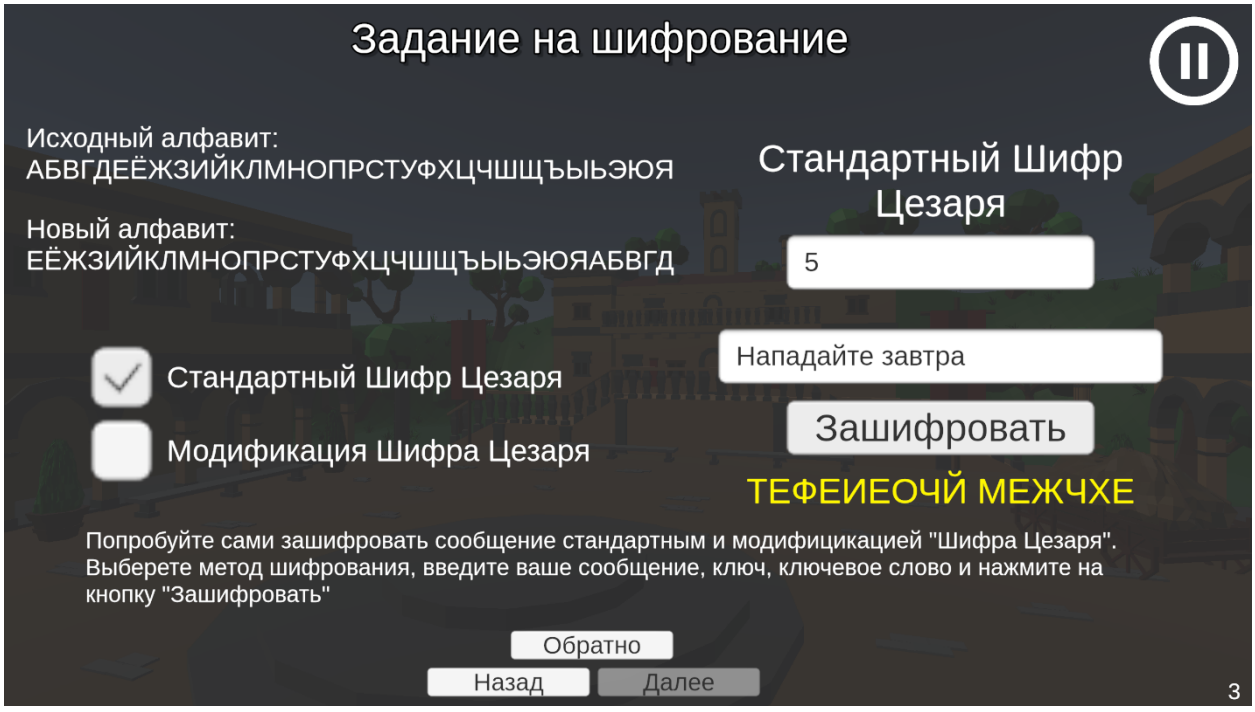


Рисунок Б.8 – Задание на шифрование с помощью шифра Цезаря

Рисунок Б.9 – Демонстрация работы шифровальной машины Enigma

ПРИЛОЖЕНИЕ В
Программная визуализация на онлайн платформе Яндекс.Игры

Рисунок В.1 – QR-код для перехода на Яндекс.Игры

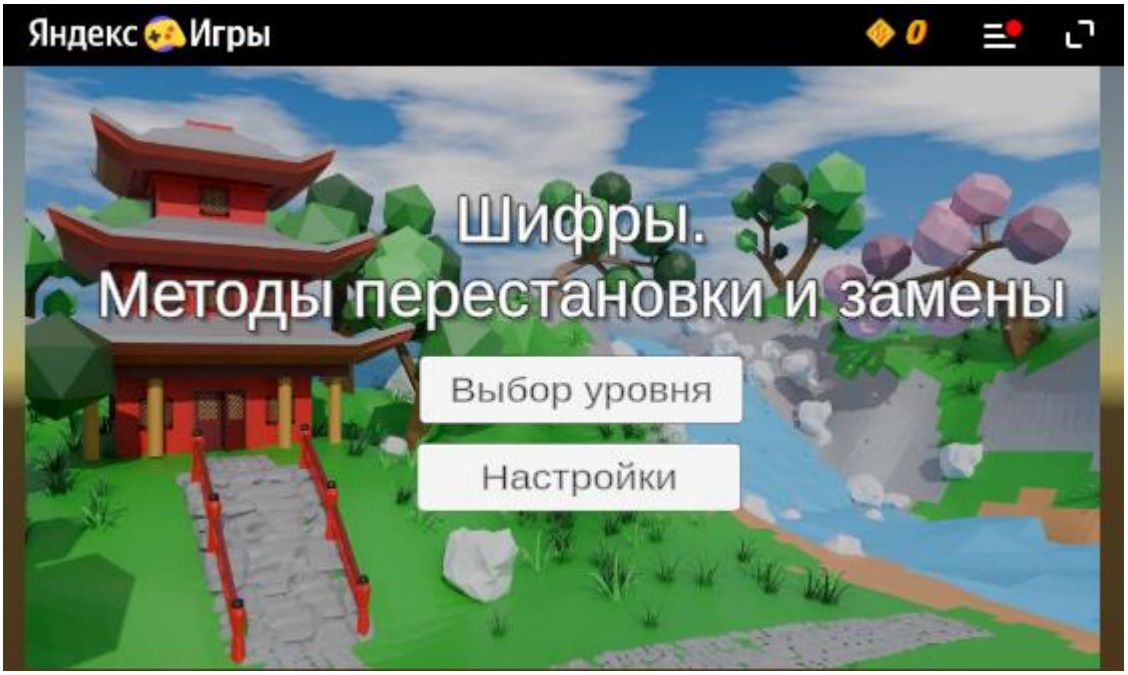


Рисунок В.2 – Работа программы на десктопном устройстве

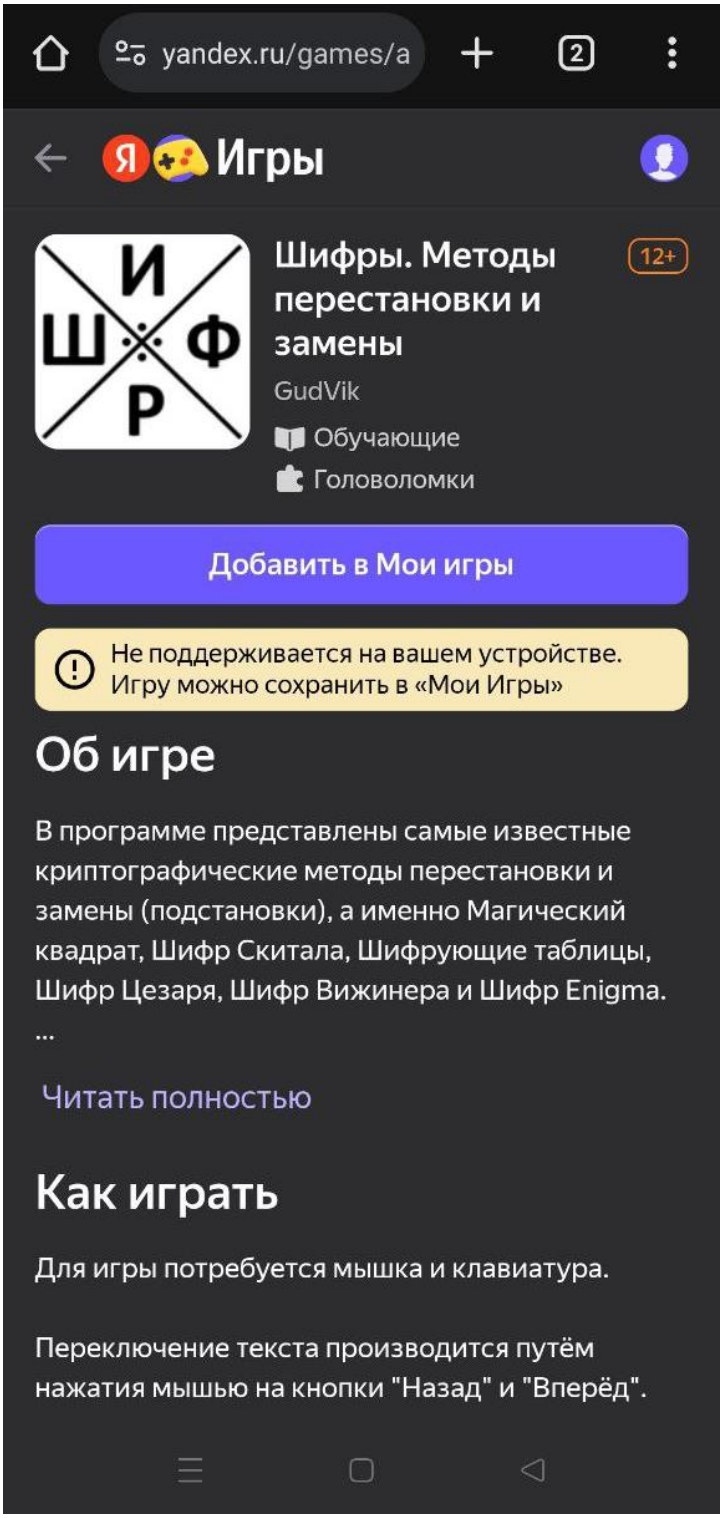


Рисунок В.3 – Экран с информацией о программе на мобильном устройстве