**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

Курсовая работа

По дисциплине «Алгоритмы и структуры данных»

Тема “Компьютерная логическая игра”  
«Двухходовые шашки»

Пояснительная записка

**Инв. № подл.**

**Подп. и дата**

**Взам. инв. №**

**Инв. № дубл.**

**Подп. и дата**

Р.02069337.2215-16 ПЗ-01

Листов 7

**Руководитель разработки**:

доцент каф. ИВК, к.т.н., доцент

*Шишкин Вадим Викторинович*

« »2022 г

**Исполнитель**:

студент гр. ИСТбд-21

*Никитин Илья Андреевич*

« »2022 г.

**2022**

**Введение**

Разработано приложение «Двухходовые шашки». За основу работы выбрана такая структура данных, как массив. Данная структура данных была использована, потому что массив имеет множество преимуществ, в совокупности с которыми разработка приложения была наиболее эффективна.

К данным преимуществам относятся:

* Массивы обеспечивают произвольный доступ к элементам. Это ускоряет доступ к элементам по положению.
* Массивы хранят несколько данных похожих типов с одним и тем же именем.
* Быстрота изменения элементов.
* Массивы компактны и занимают меньше памяти и места, что делает их более эффективными.

**1.Проектная часть**

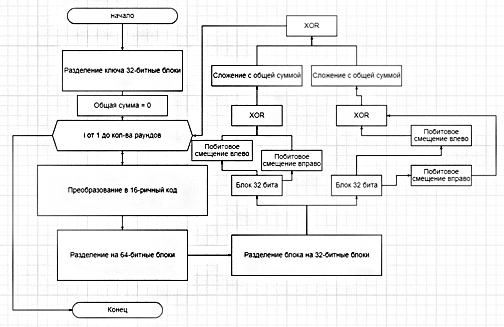
**1.1 Постановка задачи на разработку приложения**

Определяется заданием на курсовую работу. Детализируется в разработанном техническом задании (приложение 1)  
**1.2Математические методы**

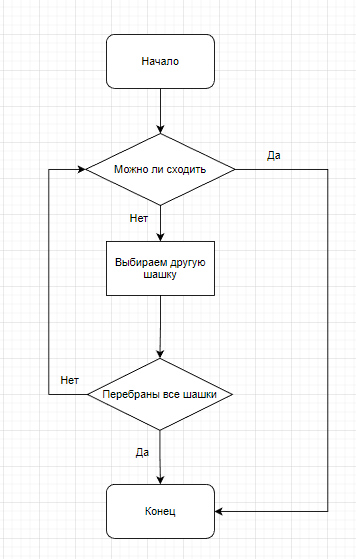
Операция XOR – исключающее или.

**1.3Архитектура и алгоритмы**

1.3.1 Алгоритм шифрования XXTE

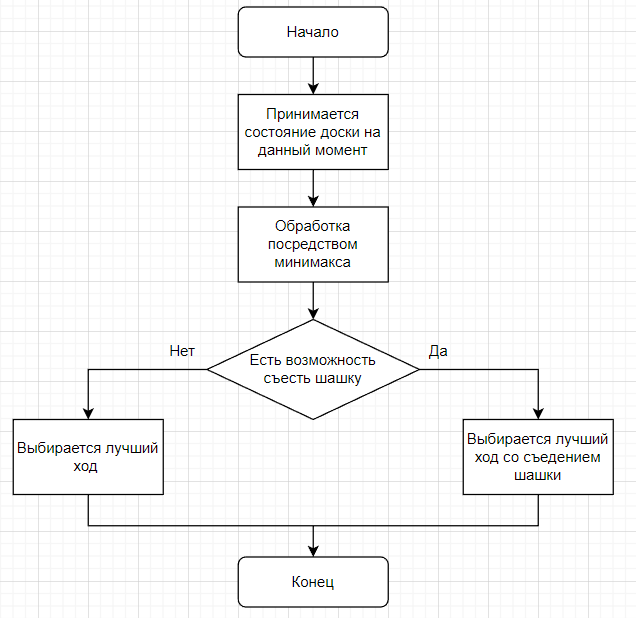


1.3.2 Алгоритм проверки возможности ход



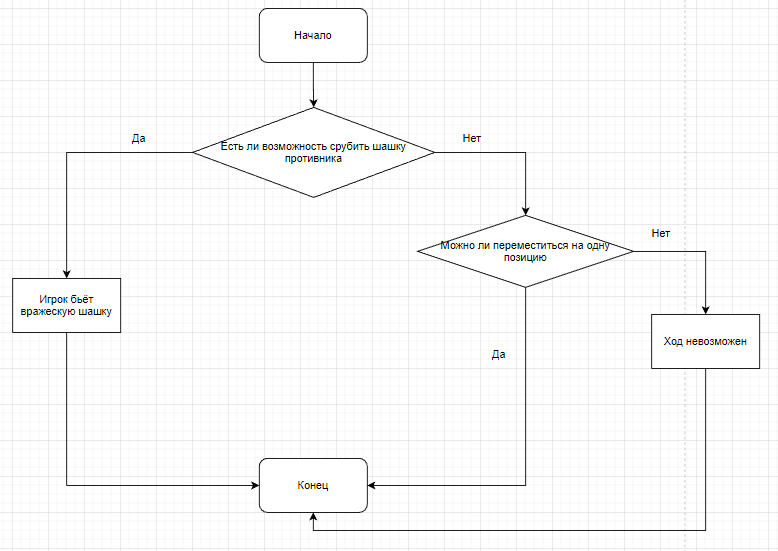
1.3.3 Алгоритм оценочной функции.

Алгоритм оценочной функции позволяет выбрать компьютеру лучший ход в соответствии с правилами игры. Если же ход невозможен – выявляет победителя

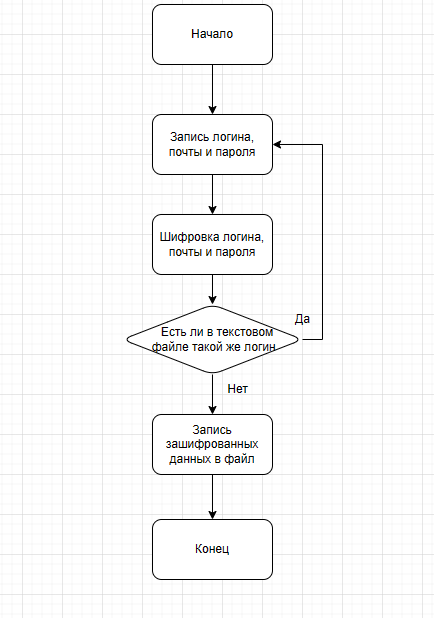


1.3.4 Алгоритм хода игрока

Алгоритм начинает свою работу с вызова функции проверки возможности хода. Затем, если в игре возникает ситуация в которой игрок может съесть вражескую пешку, то он съедает её перемещаясь по диагонали и вновь проводит проверку, если же игрок не имеет возможности съесть шашку соперника и шашка способна передвинуться на выбранную клетку – передвигает фигуру. Если же ход невозможен – выявляет победителя

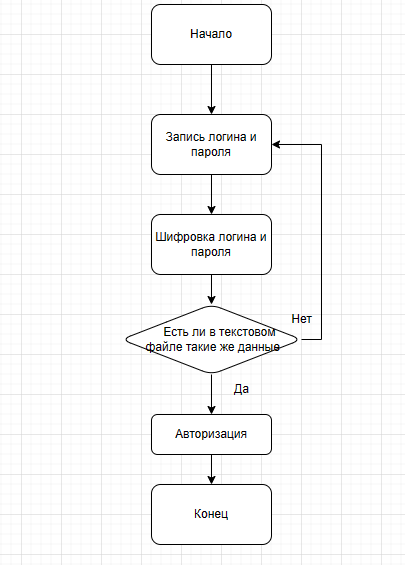


1.3.5 Алгоритм регистрации



1.3.6 Алгоритм авторизации

Алгоритм авторизации позволяет пользователю авторизоваться



1.3.7 Алгоритм начала новой игры

Алгоритм позволяет пользователю начать новую игру, очистив поле и очередь для хода.

**1.4Тестирование**

Тестирование проводилось путём игры против компьютера на заранее измененных позициях шашек для создания различных игровых ситуаций. Тесты проводится по основным принципам:

* Принцип 1 — Тестирование демонстрирует наличие дефектов (Testing shows presence of defects).  
  Тестирование только снижает вероятность наличия дефектов, которые находятся в программном обеспечении, но не гарантирует их отсутствия.
* Принцип 2 — Исчерпывающее тестирование невозможно (Exhaustive testing is impossible).  
  Полное тестирование с использованием всех входных комбинаций данных, результатов и предусловий физически невыполнимо (исключение — тривиальные случаи).
* Принцип 3 — Раннее тестирование (Early testing).  
  Следует начинать тестирование на ранних стадиях жизненного цикла разработки ПО, чтобы найти дефекты как можно раньше.
* Принцип 4 — Скопление дефектов (Defects clustering).  
  Большая часть дефектов находится в ограниченном количестве модулей.
* Принцип 5 — Парадокс пестицида (Pesticide paradox).  
  Если повторять те же тестовые сценарии снова и снова, в какой-то момент этот набор тестов перестанет выявлять новые дефекты.
* Принцип 6 — Тестирование зависит от контекста (Testing is context depending). Тестирование проводится по-разному в зависимости от контекста. Например, программное обеспечение, в котором критически важна безопасность, тестируется иначе, чем новостной портал.
* Принцип 7 — Заблуждение об отсутствии ошибок (Absence-of-errors fallacy). Отсутствие найденных дефектов при тестировании не всегда означает готовность продукта к релизу. Система должна быть удобна пользователю в использовании и удовлетворять его ожиданиям и потребностям.

Тесты проводятся по принципам соблюдения базовой верификации и

валидации, с соблюдением всех вводных требований к приложению.

**2.Источники, использованные при разработке**

1.dvahoda[Электронный ресурс]:Двухходовые шашки: http://dvahoda.ru/pravila/  
(Дата обращения: 01.12.2022)

2. Habr [Электронный ресурс]: Введение в Tkinter:

<https://habr.com/ru/post/133337/>

(Дата обращения: 02.12.2022)

3. Temofeev [Электронный ресурс]: Шифрование TEA, XTEA, XXTEA

https://temofeev.ru/info/articles/shifrovanie-tea-xtea-xxtea/  
(Дата обращения 07.12.2022)

4.Wikipedia[Электронный ресурс]: XXTEA<https://ru.wikipedia.org/wiki/XXTEA>

(Дата обращения: 07.12.2022)

**3.Приложения**

Приложение 1 - Техническое задание.

Приложение 2 - Руководство программиста.