**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»**

**Инв. № подл.**

**Подп. и дата**

**Взам. инв. №**

**Инв. № дубл.**

**Подп. и дата**

**Тема** Компьютерная логическая игра «Скифские шашки»

**Пояснительная записка**

Р.02069337.21/837-13 ТЗ-01

Листов 8

**Руководитель разработки**:

доцент каф. ИВК, к.т.н., доцент

*Шишкин Вадим Викторинович*

« » 2022 г.

**Исполнитель**:

студент гр. ИСТбд-21

*Молофеев Иван Сергеевич*

« » 2022 г.

**2022**

**Введение**

Наименование игры: Скифские шашки.

Условное наименование: COURSE.

Базовые правила «Скифских шашек» совпадают с правилами русских. Но есть и отличия: доска 9x9, у каждого игрока есть новая шашка – вождь. Она ходит, как обычная шашка, но, если кто-то съедает вождя противника – он автоматически выигрывает, потому что вождь – главная шашка для каждого.

**1. Проектная часть**

**1.1 Постановка задачи на разработку приложения**

Определяется общей постановкой задачи в задании на курсовую работу.

**1.2 Математические методы**

Математические методы не применялись.

**1.3 Алгоритмы**

**1.3.1 Алгоритм шифрования**

Алгоритм шифрует логин и пароль с помощью соответствующей функции.

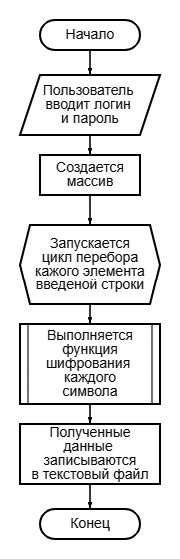


Рисунок 1 – Шифрование.

**1.3.2 Алгоритм дешифрования**

Алгоритм дешифрует логин и пароль с помощью соответствующей функции.

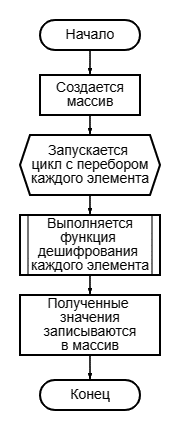


Рисунок 2 – Шифрование.

**1.3.3 Алгоритм проверка хода игрока**

Алгоритм проверяет возможность хода и делает его, если он возможен.

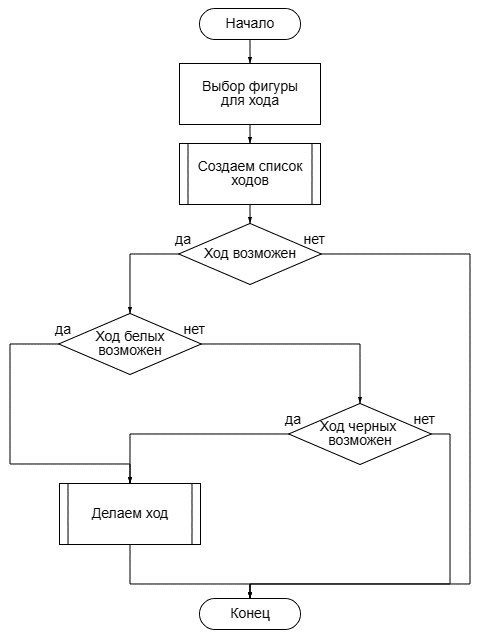


Рисунок 3 – Проверка хода игрока.

**1.3.4 Алгоритм составление списка ходов**

Алгоритм составляет список обязательных и возможных ходов.

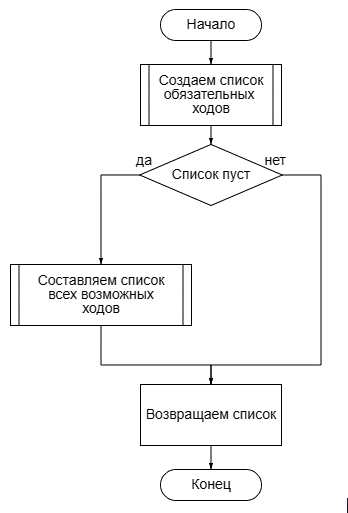


Рисунок 4 – Составление списка ходов.

**1.3.5 Алгоритм составление наличия обязательных ходов**

Данный алгоритм включает в себя два алгоритма, первый будет проверять и создавать список всех обязательных ходов, второй проверять каждую клетку поля.

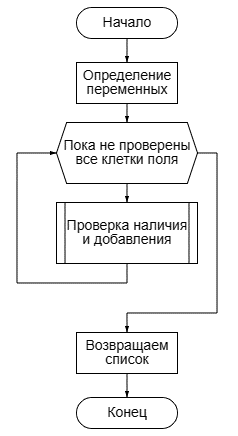


Рисунок 5 – Составление списка обязательных ходов.

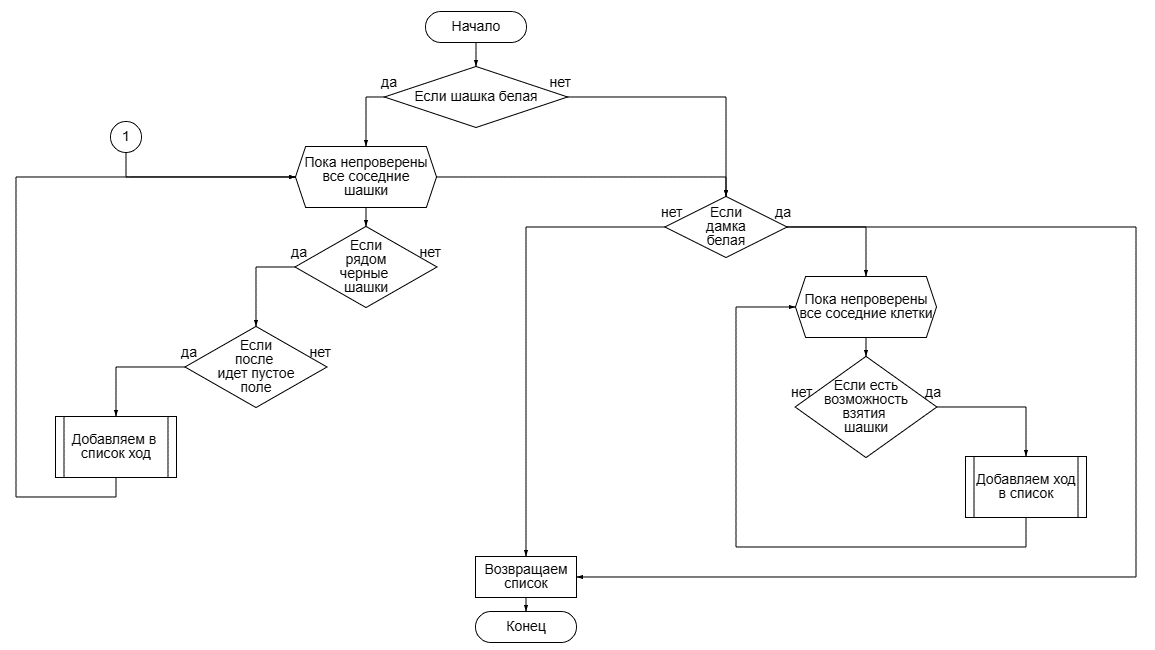


Рисунок 6 – Проверка наличия обязательных ходов.

**1.3.6 Алгоритм проверка наличия всех ходов**

Данный алгоритм будет составлять список всех возможных ходов, опять делаем это для белых, для черных алгоритм идентичный.

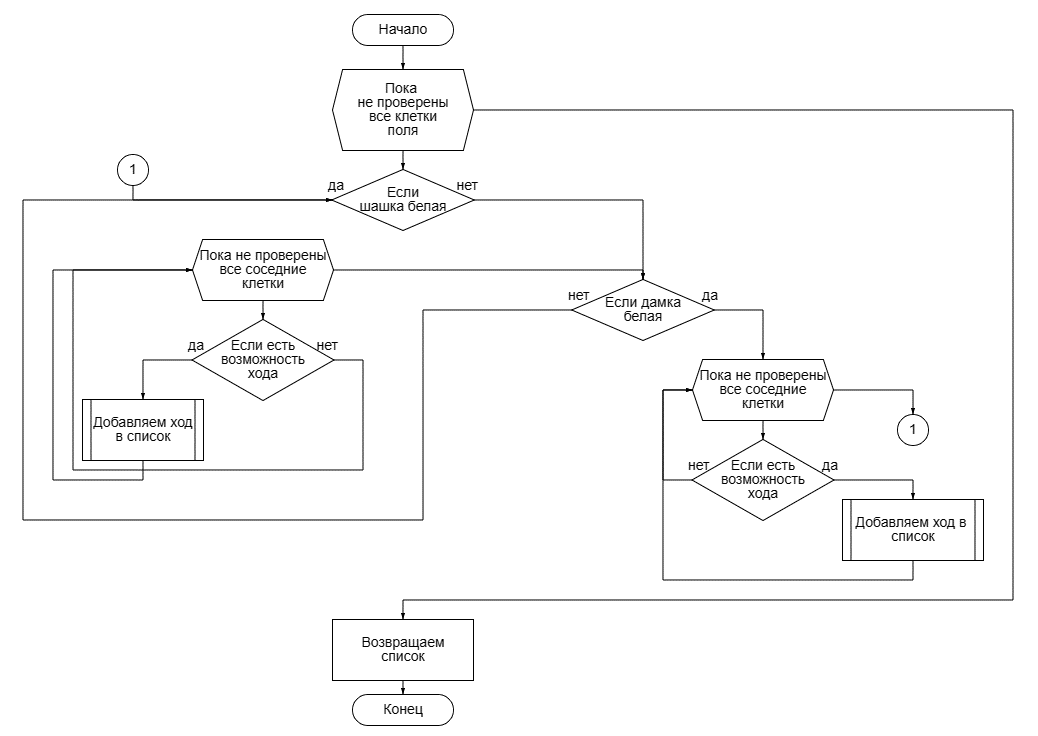


Рисунок 7 – Составление всех ходов.

**1.4 Тестирование**

Уровни тестирования:

1. Компонентное/модульное тестирование.

Этот вид тестирования выполняется на самой ранней стадии разработки программы — во время написания кода. На данном этапе проводятся следующие тесты:

* Функциональное тестирование.

Тесты:

Для окон авторизации и личного кабинета: проверка корректной работы кнопок.

Для игрового поля: проверка корректного отображения поля и шашек, проверка игры по правилам.

* Тестирование безопасности.

Тесты проводятся для окна авторизации, конкретно тестируются алгоритмы регистрации, авторизации, шифрования и дешифрования.

1. Интеграционное тестирование.

Интеграционное тестирование необходимо для того, чтобы тестировать взаимосвязь между модулями. Проводится тестирование взаимодействия 3 окон: авторизации, регистрации и игрового поля.

1. Системное тестирование.

При системном тестировании наша задача уже состоит в том, чтобы убедиться в корректности работы в целом всей системы. На данном этапе проводятся следующие тесты:

* Функциональное тестирование;
* Тестирование удобства пользования;
* Тестирование на отказ и восстановление.

Тесты:

Симулировать внезапный отказ электричества на компьютере (обесточить компьютер).

Симулировать ситуацию наличия в системе неверных данных (специальный тестовый набор или база данных).

1. Приемочное тестирование.

Ошибок на этом этапе уже не должно быть. Программа должна бытьмаксимально рабочей и пригодной для использования**.** На данном этапе проводится тестирование установки программы.

**1.4.1 Методика проведения и результаты тестирования**

При разработке данной программы были допущены следующие синтаксические ошибки:

* неправильное использование операторов присваивания;
* неверное объявление циклов.

При тестировании были выполнены следующие принципы:

* тщательность подбора данных для теста программы, не только для правильных входных данных, но и для неправильных;
* доскональное изучение результатов тестирования.

**1.4.2 Отладка выявленных ошибок, обнаруженных при тестировании**

Во время выполнения отладки, были использованы следующие методы:

* анализ кода без исполнения программы;
* запуск программы из-под отладчика.

Все синтаксические ошибки были исправлены при компиляции проекта, учитывая синтаксические особенности среды программирования. Также в программном средстве возникали логические ошибки, которые были исправлены путем пересмотра кода алгоритмов и его последующего исправления.

Отладка ряда ошибок прошла успешно, были внесены изменения в интерфейс программы, а также были предусмотрены некоторые исключительные ситуации.

**2. Источники, использованные при разработке**

1. Python Checkers AI Tutorial Part 2 — Реализация и визуализация (Minimax) // YouTube URL: Python Checkers AI Tutorial Part 2 — Реализация и визуализация (Minimax) – YouTube (дата обращения: 25.12.2022).

2. Algorithms Explained – minimax and alpha-beta pruning // YouTube URL: https://www.youtube.com/watch?v=l-hh51ncgDI (дата обращения: 26.12.2022)

3. Checkers-Python // Medium URL: https://medium.com/analytics-vidhya/checkers-python-eff2786b985b (дата обращения: 22.12.2022)

4. Игра Шашки на Питоне // CyberForum URL: https://www.cyberforum.ru/blogs/914198/blog4540.html (дата обращения: 21.12.2022).

5. Tkinter — создание графического интерфейса в Python // python-scripts URL: https://python-scripts.com/tkinter (дата обращения: 23.12.2022).

6. Введение в Tkinter // Habr URL: https://habr.com/ru/post/133337/ (дата обращения: 23.12.2022).

7. Функции \_\_str \_\_() и \_\_repr \_\_() в Python // pythonim URL: https://pythonim.ru/string/funktsii-str-repr-python (дата обращения: 03.01.2023).

8. Путь к пониманию декораторов в Python // Habr URL: https://habr.com/ru/company/wunderfund/blog/657355/ (дата обращения: 03.01.2023).

9. Python Tkinter // YouTube URL: https://www.youtube.com/playlist?list=PLQAt0m1f9OHsd6U5okp1XLoYyQR0oBjMM (дата обращения: 04.01.2023).

10. Минимакс с Альфа-Бета-обрезкой в Python // pythobyte URL: https://pythobyte.com/minimax-and-alpha-beta-pruning-in-python-fe960495/ (дата обращения: 02.01.2023).

**3. Приложения**

1) Приложение 1 – «Техническое задание».

2) Приложение 2 – «Руководство программиста».