**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 «Апит Содок»

**Инв. № подл.**

**Подп. и дата**

**Взам. инв. №**

**Инв. № дубл.**

**Подп. и дата**

Пояснительная записка

Р.02069337.21/820-19 ПЗ-03

Листов 11

**Руководитель разработки**:

профессор каф. ИВК, к.т.н., доцент

*Шишкин Вадим Викторинович*

« » 2022 г.

**Исполнитель**:

студент гр. ИСТбд-22

*Кадырова Динара Руслановна*

« » 2022 г.

**2022**

**Введение**

Разрабатывается приложение «Апит Содок». Данное приложение представляет из себя игру наподобие шашек с особыми правилами игры. В работе выбраны такие структуры данных, как массив и словарь. Структура данных — массив была выбрана потому, что эта структура имеет множество преимуществ:

1. Массивы обеспечивают произвольный доступ к элементам. Это ускоряет доступ к элементам по положению;
2. Массивы хранят несколько данных похожих типов с одним и тем же именем;
3. В массиве данные организованны таким образом, что ими легко и удобно манипулировать.

Структура данных — словарь был выбран, потому что он также обладает рядом преимуществ:

1. **В словарях доступ к элементам выполняется по ключу, а не по индексу**;
2. Словари могут содержать объединенные данные в виде записей;
3. **Словари имеют переменную длину;**
4. **Словари представляют неупорядоченные коллекции произвольных объектов.**

**1. Проектная часть**

**1.1 Постановка задачи на разработку приложения**

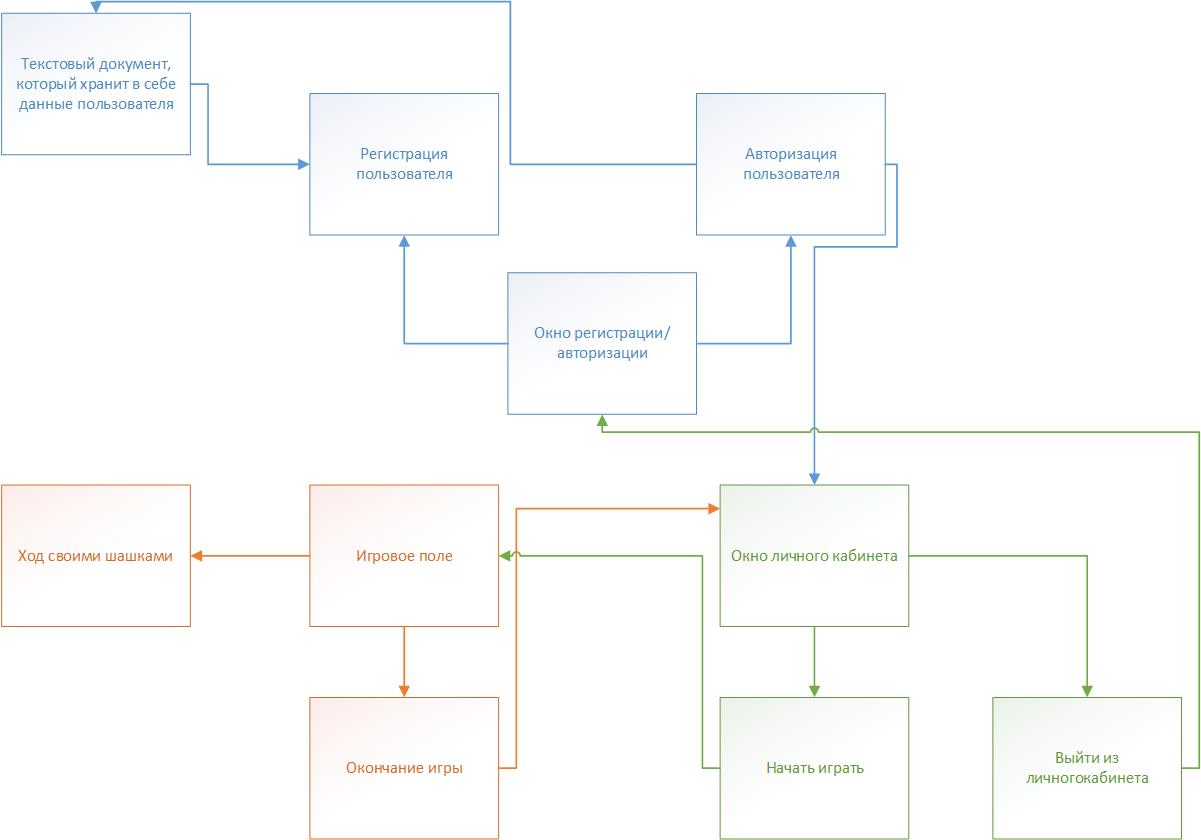
Определяется общей постановкой задачи в задании на курсовую работу. Детализируется в разработанном техническом задании (приложение 1)

**1.2 Математические методы**

Математические методы не применялись.

**1.3 Алгоритмы**

1.3.1. Архитектура



1.3.2 Алгоритм шифрования

Данный криптографический алгоритм предназначен для защиты логина и пароля пользователя. Каждый символ введённой строки сначала преобразовывается в число из таблицы Unicode, а потом шифруется с помощью дополнительной функции. Зашифрованные данные записываются в текстовый документ.



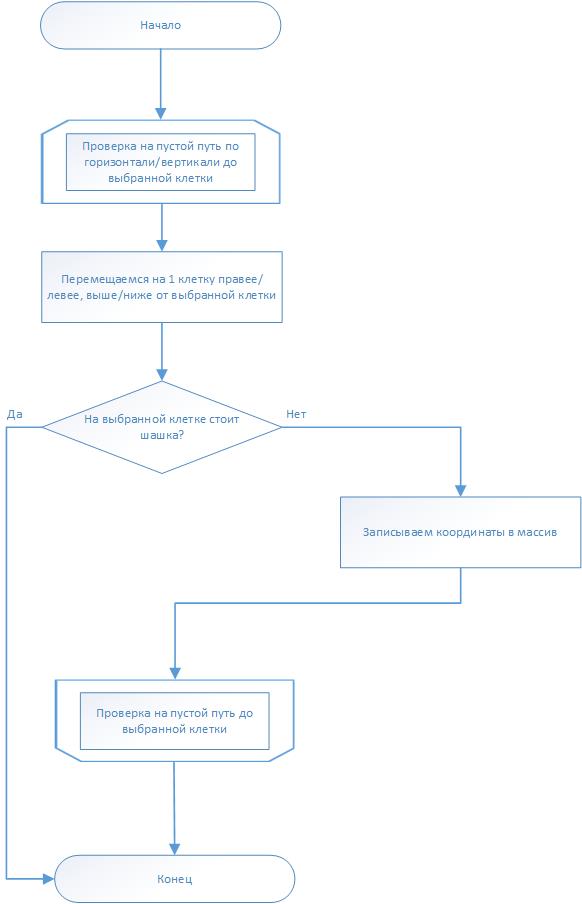
1.3.3 Алгоритм дешифрования

Данный алгоритм осуществляет расшифровку логина и пароля пользователя. Каждый элемент сначала дешифруется с помощью дополнительноой функции, а потом преобразовывается в символ из таблицы Unicode. После алгоритм сравнивает расшифрованные данные с введёнными данными пользователем.



1.3.4 Алгоритм проверки возможности хода

Данный алгоритм проверяет возможность сходить шашкой в соответствии с правилами игры. Проверяются клетки от выбранной до текущей. Если они пустые, то ход возможен.



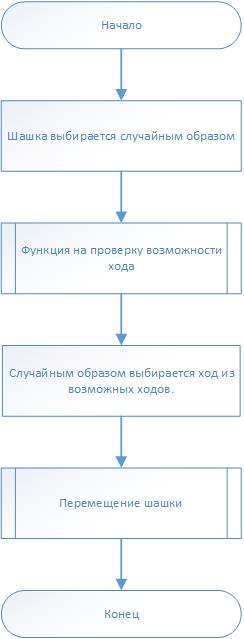
1.3.5 Алгоритм хода игрока

Данный алгоритм совершает ход игрока. Для начала идёт проверка возможности хода, затем они подсвечиаются. После выбора хода игрока, шашка переририсовывается.



1.3.6 Алгоритм хода компьютера

Данный алгоритм осуществляет ход противника(компьютера). Данный алгоритм совершает ход случайной шашкой, соответсвуя правилам игры.



1.3.7 Алгоритм регистрации

Данный алгоритм осуществляет регистрацию пользователя, зашифровывая данные с помощью алгоритма шифрования и записывая данные в текстовый документ .txt

1.3.8 Алгоритм авторизации

Алгоритм осуществляет авторизацию пользователя. Пользователь вводит логин и пароль, а затем алгоритм проверяет зарегистрирован ли такой пользователь.

1.3.9 Алгоритм выхода из игры

Алгоритм осуществляет выход пользователя из игры по нажатию на крестик в верхнем правом углу.

1.3.10 Алгоритм определения победителя

Данный алгоритм определяет победителя в сыгранной игре. Результат выводится в личном кабинете.

1.3.11 Алгоритм проверки окончания игры

Алгоритм проверяет по правилам игры закончилась ли игра.

1.3.12 Алгоритм подсвечивания возможных ходов

Алгоритм используя данные, полученные из алгоритма возможности хода, подсвечивает голубым все возможные ходы.

1.3.13 Алгоритм проверки ходов на правильность

Данный алгоритм отслеживает движение шашки и удаляет съеденные шашки противника.

1.3.14 Алгоритм выхода из личного кабинета

Алгоритм осуществляется выход из личного кабинета при нажатии на кнопку «Выйти». После выхода, у пользователя открывается окно авторизации.

**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, личного кабинета 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 быть**максимально рабочей и пригодной для использования.** На данном этапе проводится тестирование установки программы.

**2. Источники, использованные при разработке**

1. Wikipedia [Электронный ресурс]: Апит Содок (дата обращения 07.11.2022);
2. Wikipedia [Электронный ресурс]: RSA (дата обращения 17.11.2022);
3. Постолит А.В. Основы искусственного интеллекта в примерах на Python. Самоучитель. [Электронный ресурс]: книга / СПб.: БХВ-Петербург, 2021. -448 с.: ил. -(Самоучитель) (дата обращения 19.10.2022).