МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Измерительно-вычислительные комплексы»

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | |  | |  |  |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл*** |  | |  | | Руководство программиста  на курсовую работу  по дисциплине «Алгоритмы и структуры данных»  Тема «Разработка компьютерной игры крестики нолики» | | | | | | |
|  | | |  | |  | | | |
|  | | Исполнитель  студент гр. ИСТбд-21  Клянченков М.А.  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2024 г. | | | |
|  | | 2024 | | | | | | |

**1. Введение**

Это руководство предназначено для программистов, которые будут поддерживать или расширять функциональность игры, разработанной с использованием Python и Tkinter. В руководстве описываются основные архитектурные решения, структура проекта и порядок его запуска.

**2. Среда разработки**

Для работы с проектом требуется установить следующие инструменты:

* **Python 3.x**: основной язык программирования.
* **Tkinter**: библиотека для создания графических окон. Обычно включена в дистрибутив Python.

Для запуска игры достаточно поддерживать актуальные версии Python и указанных библиотек.

**3. Структура проекта**

Проект состоит из одного файла:

1. Содержит весь код игры, включая логику игры, интерфейс и искусственный интеллект.

**4. Описание основных классов и модулей**

1. Класс TicTacToe

Описание: Этот класс представляет собой основную логику игры "Крестики-нолики".

Основные методы:

* \_\_init\_\_(self): инициализирует игру, создает окно и устанавливает начальные параметры.
* initialize\_ui(self): создает пользовательский интерфейс игры.
* initialize\_board(self): инициализирует игровое поле.
* create\_mode\_buttons(self): создает кнопки выбора режима игры.
* on\_click(self, row, col): обрабатывает клик по ячейке игрового поля.
* make\_move(self, row, col, player): выполняет ход игрока.
* ai\_move(self): выполняет ход компьютера.
* minimax(self, depth, is\_maximizing): реализует алгоритм минимакс для ИИ.
* check\_win(self, player): проверяет, выиграл ли игрок.
* check\_draw(self): проверяет, закончилась ли игра вничью.
* reset\_game(self): сбрасывает игру к начальному состоянию.
* start\_game(self): запускает игровой цикл.

**5. Детали реализации**

1. Пользовательский интерфейс

* Игровое поле реализовано в виде сетки кнопок 3x3.
* Кнопки "Рестарт", "Игрок против Игрока" и "Игрок против Компьютера" расположены под игровым полем.

1. Логика игры

* Игроки ходят по очереди, ставя "X" или "O" на свободные клетки поля.
* Победа достигается при заполнении трех клеток в ряд, столбец или по диагонали.
* Если все клетки заполнены и нет победителя, объявляется ничья.

1. Искусственный интеллект

* ИИ использует алгоритм минимакс для выбора оптимального хода.
* Алгоритм просчитывает все возможные ходы и выбирает наилучший.

**6. Расширение функциональности**

1. Добавление новых режимов игры

* Можно добавить новые режимы игры, создав дополнительные методы в классе TicTacToe и соответствующие кнопки в методе create\_mode\_buttons.

1. Улучшение ИИ

* Можно оптимизировать алгоритм минимакс, добавив альфа-бета отсечение для увеличения скорости работы.

1. Добавление статистики

* Можно добавить систему подсчета очков и сохранения статистики игр.

**7. Тестирование и отладка**

1. **Основные тесты:**

* Проверка корректности определения победы для всех возможных комбинаций.
* Тестирование ИИ на различных сценариях игры.
* Проверка корректности работы кнопки "Рестарт" и смены режимов игры.

1. **Отладка:**

* Для отладки рекомендуется использовать встроенный отладчик Python в IDE (например, PyCharm или VS Code).
* Можно добавить логирование ходов и состояний игры для анализа работы программы.

**8. Заключение**

Данное руководство описывает основные аспекты проекта "Крестики-нолики" и предоставляет необходимую информацию для понимания структуры кода, его функциональности и методов расширения. При возникновении вопросов или необходимости внесения изменений, рекомендуется обращаться к данному руководству и исходному коду проекта.