**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

**Лабораторная работа**

**По дисциплине «Алгоритмы и структуры данных»**

**Тема Лабораторная работа №10**

**Руководство программиста**

|  |  |
| --- | --- |
| **Подп. и** |  |
| **Инв** |  |
| **Вза** |  |
| **Подп. и** |  |
| **Инв.** |  |

Р.02069337. <23/710>-<04> РП-<2-зн. номер

**дата**

редакции> Листов 4

**. № дубл.**

Исполнитель:

**м. инв. №**

студент гр. ИСТбд-23

*Калашников А. А.*

« 20 » мая 2025 г.

**дата**

2025

**№ подл.**

**1. Назначение и условия применения программы**

**1.1. Назначение и функции, выполняемые приложением**

Приложение "Крестики-Нолики" предназначено для реализации классической игры, позволяя пользователям взаимодействовать через графический интерфейс. Основные функции приложения включают:

* Игровое поле: Приложение предоставляет 3x3 ячейки, где пользователи могут делать свои ходы поочередно.
* Определение победителя: Программа автоматически проверяет наличие выигрышной комбинации после каждого хода.
* Ход ИИ: Игра включает искусственный интеллект, который принимает ход.
* Уведомления: При завершении игры пользователи получают уведомление о победителе или ничьей.
* Сброс игры: После завершения партии пользователи могут начать новую игру с очисткой поля.

**1.2. Условия, необходимые для использования приложения**

Для корректного запуска приложения требуются следующие условия:

* Операционная система: Windows, macOS или Linux.
* Платформа: Python 3.6 и выше.
* Инструментальная среда: Рекомендуется использовать IDE, такую как PyCharm, VS Code или любой текстовый редактор.
* Библиотеки: Необходимы библиотеки:
  + Tkinter для создания графического интерфейса.
  + MessageBox для отображения уведомлений о результате игры.

**2. Характеристики программы**

**2.1. Характеристики приложения**

Приложение содержит 105 строк кода и использует:

* Структуры данных: 2.
* Алгоритмов: 6.

Применяемые библиотеки:

* tkinter: для создания графического интерфейса.

Внешний вид:

Изображение выглядит как снимок экрана, прямоугольный, текст, число

Автоматически созданное описание

**2.2. Особенности реализации приложения**

**Ключевые функции**

* Обработка кликов: Пользователь может нажимать на ячейки игрового поля. Каждый клик обновляет состояние поля, и если ячейка была пустой, в неё помещается знак текущего игрока ("X" или "O") с помощью метода on\_button\_click().
* Определение победителя: После каждого хода приложение проверяет наличие выигрышной комбинации с помощью метода check\_winner(). Если победитель найден, выводится уведомление.
* Ход искусственного интеллекта (AI): При переходе ко второму игроку (AI), программа производит ход автоматически с использованием метода computer\_move().
* Уведомления об окончании игры: Приложение использует messagebox для отображения уведомлений о результатах игры, таких как победа одного из игроков или ничья.
* Сброс игры: После завершения игры пользователи могут начать новую с помощью метода reset\_board(), который очищает игровое поле и восстанавливает состояние текущего игрока.

**Возможные улучшения**

* Оптимизация AI: Усовершенствовать алгоритм искусственного интеллекта для разработки более сложной стратегии, добавив дополнительные условия и варианты ходов.
* Улучшение интерфейса: Рассмотреть возможность добавления настройки уровня сложности в игру, а также опции для игры с другим человеком.
* Документация: Улучшить читаемость кода, добавив комментарии и документацию к методам и классам для обеспечения лучшего понимания структуры программы.

**Тестирование**

* Реализовать тесты: Провести тестирование основных функциональных возможностей приложения, особенно для методов проверки победителей и обработки ходов, чтобы убедиться в их корректной работе.

**3. Обращение к программе**

Основные методы

* + create\_board(): создает игровое поле
  + on\_button\_click(): обработка нажатия на кнопку игрового поля(Ход игрока).
  + minimax(): Вычисление наилучшего хода для ИИ
  + computer\_move(): выполнение хода ИИ.
  + check\_winner(): определение победителя игры.
  + reset\_board(): сброс состояния игры для новой партии.

**4. Сообщения**

Приложение отображает следующие сообщения в результате контроля игры:

* "Победа! Игрок «Х или О» выиграл!": Если игрок обыграл компьютера.
* "Ничья!": Сообщение о ничьей.
* " Победа! Компьютер «Х или О» выиграл!": Если компьютер обыграл игрока.