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 лабораторную работу №9  по дисциплине «Алгоритмы и структуры данных»  Тема «Игра крестики-нолики» | | | | | | |
|  | | |  | |  | | | |
|  | | Исполнитель  студент гр. ИСТбд-21 Мишин А.В.  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2024 г. | | | |
|  | | 2024 | | | | | | |

**1. Назначение и условия применения программы**

**1.1. Назначение программы**

Программа представляет собой приложение для игры в "Крестики-нолики" с графическим интерфейсом. Она позволяет двум пользователям или одному игроку против компьютера сыграть в эту классическую игру на поле 3x3.

**1.2. Функции программы**

Программа выполняет следующие функции:

• Создание новых задач с указанием названия, категории, дедлайна и времени

• Отображение списка задач в виде таблицы

• Редактирование существующих задач

• Удаление отдельных задач или всех задач сразу

• Отметка задач как выполненных

• Сортировка задач по различным параметрам

• Управление категориями задач

• Сохранение и загрузка задач из файла

**1.3. Условия применения**

Для использования программы необходимы:

* Python версии 3.x.
* Библиотека tkinter для создания графического интерфейса.
* Возможность работы на операционных системах Windows, Linux, macOS.

**2. Характеристика программы**

**2.1. Общие сведения**

Программа реализована с использованием объектно-ориентированного подхода. Основной класс представляет игровую логику и интерфейс, организующий взаимодействие между пользователем и игрой. Программа использует стандартные библиотеки Python для работы с графическим интерфейсом и временными метками.

**2.2. Основные характеристики**

Программа использует графический интерфейс на основе tkinter. Игровое поле отображается с помощью кнопок, каждая из которых представляет одну клетку. В зависимости от состояния игры, отображаются символы "❌" и "⚫". Программа реализует логику переключения ходов между игроками, а также использования алгоритма Minimax для игры против компьютера.

**3. Обращение к программе**

3.1. Класс Game

3.1.1. Метод \_\_init\_\_(self, master)

Назначение: инициализация основного окна приложения и всех виджетов.

3.1.2. Метод create\_widgets(self)

Назначение: создание и размещение всех элементов интерфейса.

3.1.3. Метод start\_game(self)

Назначение: запуск игры в выбранном режиме: против игрока или компьютера.

3.1.4. Метод button\_click(self, index)

Назначение: обработка нажатия на клетку игрового поля и обновление состояния игры.

3.1.5. Метод check\_winner(self)

Назначение: проверка наличия победителя после каждого хода.

3.1.6. Метод computer\_turn(self)

Назначение: выполнение хода компьютера с использованием алгоритма Minimax.

3.1.7. Метод switch\_turn(self)

Назначение: переключение хода между игроками.

3.1.8. Метод reset\_game(self)

Назначение: перезапуск игры, очистка поля и сброс всех переменных.

3.1.9. Метод update\_timer(self)

Назначение: обновление таймера, отслеживание времени игры.

3.2. Методы для графического интерфейса

3.2.1. Метод show\_main\_menu(self)

Назначение: отображение главного меню с выбором режима игры.

3.2.2. Метод show\_game\_screen(self)

Назначение: отображение экрана игры, скрытие главного меню.

**4. Входные и выходные данные**

**4.1. Входные данные**

* Входные данные включают действия пользователя (выбор режима игры, нажатия на кнопки для выполнения ходов).
* Выбор режима игры: "Игрок против игрока" или "Игрок против компьютера".

**4.2. Выходные данные**

* Отображение на экране текущего состояния игры (кто ходит, победитель или ничья).
* Вывод сообщений о победителе или ничьей через окна messagebox.
* Время игры отображается на экране.

**5. Сообщения**

Программа выводит следующие сообщения:

* "Ходят ❌/⚫" — для отображения текущего игрока.
* "Игра закончилась в ничью!" — если ничья.
* "Вы выиграли!" или "Компьютер выиграл!" — при победе игрока или компьютера.

**6. Используемые технические средства**

Для работы программы используются следующие библиотеки Python:

* tkinter: создание графического интерфейса.
* time: отслеживание времени игры.
* messagebox: отображение сообщений о результатах игры.

**7. Особенности реализации**

* Программа использует объектно-ориентированное проектирование.
* Интерфейс разделен на несколько функциональных блоков: главное меню, экран игры, панель управления.
* Реализована логика игры против компьютера с использованием алгоритма Minimax.
* Данные игры (победители, состояние поля) не сохраняются после завершения игры, так как программа предназначена для однократных запусков с перезапуском.