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*** |  | |  | | ТЕХНИЧЕСКОЕ ЗАДАНИЕ  на лабораторную 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**

Целью лабораторной работы является разработка графического приложения "Крестики-нолики" на языке Python с использованием библиотеки tkinter. Программа должна предоставлять пользователю возможность играть в игру "Крестики-нолики" с другим игроком или компьютером, используя искусственный интеллект для хода компьютера.

1. **Заказчик**

Преподаватель Шишкин В.В.

1. **Основания для разработки**

Разработка осуществляется на основании учебного плана направления «Информационные системы и технологии».

1. **Основные требования к функциональности**

**4.1. Основные функции программы:**

Игра "Крестики-нолики" с двумя игровыми режимами:

Игрок против игрока

Игрок против компьютера (с использованием алгоритма Minimax для оптимального выбора ходов)

Возможность сброса игры в любой момент для начала новой партии.

Индикация текущего игрока ("X" или "O").

Сообщение о победе одного из игроков или о ничьей.

**4.2. Организация пользовательского интерфейса:**

Поле 3x3 с кнопками, представляющими ячейки игрового поля.

Кнопки для выбора режима игры: "Игрок против Игрока" и "Игрок против Компьютера".

Кнопка для сброса игры и начала новой партии.

Оповещение пользователя о результате игры через диалоговые окна (победа, ничья).

**4.3. Взаимодействие с пользователем:**

Кнопки игрового поля должны быть интерактивными, позволяя игрокам делать ходы, кликая по ячейкам.

При нажатии кнопки игрового поля срабатывает функция хода игрока.

При игре против компьютера программа выполняет ход от имени компьютера сразу после хода игрока.

1. **Требования к программной реализации**

**5.1. Язык программирования и библиотеки**

Программа должна быть реализована на языке Python (версии 3.8 и выше).

Для графического интерфейса необходимо использовать библиотеку tkinter, встроенную в стандартную библиотеку Python.

**5.2. Логика игры**

Программа должна проверять состояние игрового поля после каждого хода для определения победителя или ничьей.

Алгоритм для хода компьютера должен использовать метод Minimax для оптимального выбора позиции.

1. **Требования к надежности**

Программа должна корректно обрабатывать невалидные действия и предотвращать ошибки (например, попытку сделать ход в занятую ячейку).

Программа должна завершать игровую сессию после определения победителя или ничьей, блокируя дальнейшие ходы до начала новой игры.

1. **Состав и параметры технических средств**

Программа должна корректно работать на компьютерах, поддерживающих Python версии 3.8 и выше.

1. **Сроки выполнения**

Срок выполнения: 2 недели с момента выдачи задания.

1. **Этапы разработки**

Проектирование интерфейса: Определение структуры окна и расположения элементов.

Разработка основной логики игры: Реализация функций для обработки ходов игроков и проверки условий победы или ничьей.

Реализация алгоритма Minimax: Создание функции для оптимального выбора хода компьютера.

Тестирование: Проверка работоспособности всех функций, включая взаимодействие с интерфейсом и логику игры.

1. **Порядок контроля и приемки**

Контроль осуществляется путем тестирования всех функций программы, проверки интерфейса и корректности работы логики игры.