МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Измерительно-вычислительные комплексы»

**ТЕХНИЧЕСКОЕ ЗАДАНИЕ**

на лабораторную работу №10

по дисциплине «Алгоритмы и структуры данных»

**Разработка компьютерной игры «Крестики-нолики» с режимами игры против компьютера и другого игрока**

Р.02069337. №23/711-Вариант 26

Листов – 5

Исполнитель:

студент гр. ИСТбд-23

Фокин Даниил Сергеевич

«\_\_» \_\_\_\_\_\_\_\_ 2024 г.

Ульяновск 2024 г.

**Введение**

Проект "Разработка компьютерной игры «Крестики-нолики»" представляет собой приложение, предназначенное для игры в классическую настольную игру "Крестики-нолики". Приложение позволяет пользователю выбрать один из двух режимов игры:

1. Игра против другого игрока на одном устройстве;
2. Игра против компьютера с использованием алгоритма Minimax для выбора оптимальных ходов.

Цель игры: добиться заполнения трех ячеек подряд своим символом ("X" или "O") горизонтально, вертикально или по диагонали.

Приложение предоставляет простой и интуитивно понятный графический интерфейс.

1. **Основания для разработки**

В качестве оснований для разработки указывается учебный план направления 09.03.02 «Информационные системы и технологии» и распоряжение по факультету.

1. **Требования к программе**
   1. **Функциональное назначение**

Программа "Крестики-нолики" предназначена для автоматизации игрового процесса, предоставляя пользователям возможность выбора между двумя режимами игры: игра против другого человека или игра против компьютера. Основная цель игры — заполнить три клетки подряд своим символом ("X" или "O") горизонтально, вертикально или по диагонали, при этом соперник должен быть заблокирован от выполнения этой же задачи. Пользовательский интерфейс разработан так, чтобы игра была интуитивно понятной и доступной для пользователей разного возраста. Игровой процесс предполагает использование поля размером 3x3, на котором участники поочередно совершают ходы, выбирая свободные клетки. В случае заполнения всех клеток без определения победителя игра заканчивается ничьей. Реализация режима игры против компьютера основана на алгоритме Minimax, который обеспечивает оптимальное решение, делая игру сложной и увлекательной. Приложение ориентировано на использование на персональных компьютерах с установленной библиотекой tkinter, и его функционал не требует от пользователей сложных технических навыков.

* 1. **Требования к функциональным характеристикам**
     1. Требования к структуре программы

Структура приложения организована модульно для обеспечения легкости обновления и модификации кода. Приложение включает три основных модуля: главный модуль, модуль игры "человек против человека" и модуль игры "человек против компьютера". Главный модуль отвечает за отображение начального меню и выбор игрового режима. Модуль игры "человек против человека" управляет процессом двухпользовательской игры, обеспечивая поочередное выполнение ходов и проверку условий окончания партии. Модуль игры "человек против компьютера" реализует стратегию хода компьютера на основе алгоритма Minimax, что обеспечивает расчет оптимального решения для каждой игровой ситуации. Каждый модуль изолирован и взаимодействует с другими только через строго определенные интерфейсы, что способствует упрощению тестирования и сопровождения.

* + 1. Требования к составу функций

Основными функциями приложения являются отображение интерфейса, обработка пользовательских действий и управление логикой игры. При запуске программы пользователь видит меню, где может выбрать режим игры. В режиме "человек против человека" программа обрабатывает нажатия на клетки игрового поля, обновляет состояние игры, проверяет условия победы или ничьей и уведомляет игроков о результатах. В режиме "человек против компьютера" дополнительно выполняется расчет оптимального хода компьютера после каждого хода игрока. Компьютер действует за символ "O" и выбирает оптимальную клетку для хода, чтобы либо обеспечить победу, либо предотвратить поражение. После завершения игры пользователю предлагается перезапустить сессию. Пользовательский интерфейс адаптирован для визуального отображения текущего состояния поля, что позволяет игрокам легко понимать ход игры.

* + 1. Требования к организации информационного обеспечения

Программа принимает пользовательские действия в виде нажатий на клетки игрового поля, определяя координаты выбранной ячейки. На выходе приложение отображает обновленное состояние игрового поля, включая занятые клетки, и выводит сообщения о победе, ничьей или необходимости совершить следующий ход. Все операции отображаются графически с помощью библиотеки tkinter. Структура данных игрового поля организована в виде двумерного массива размером 3x3, где каждая ячейка может содержать символ "X", "O" или быть пустой. Взаимодействие между модулями осуществляется через прямой вызов функций, что обеспечивает простоту обмена данными. Информация о текущем состоянии игры сохраняется в оперативной памяти и очищается при сбросе или завершении сессии.

* 1. **Требования к надежности**

Программа должна быть устойчивой к ошибкам, вызванным некорректными действиями пользователя, такими как попытка сделать ход в занятую клетку или продолжить игру после завершения партии. Она должна обеспечивать корректную работу даже при интенсивных действиях пользователей, включая быстрое нажатие на кнопки или многократное перезапуск игры. Для восстановления игры после завершения партии предусмотрен сброс состояния игрового поля, что исключает любые сбои или некорректные данные в новой сессии. Приложение должно работать стабильно в течение длительного времени, корректно освобождая ресурсы системы после завершения работы.

* 1. **Требования к информационной и программной совместимости**

Программа должна работать в операционных системах, поддерживающих библиотеку tkinter.

Необходимое программное обеспечение:

* Python 3.8 и выше.
* Стандартная библиотека python (tkinter)
  1. **Требования к маркировке и упаковке**

Определяются задание на лабораторную работу.

* 1. **Требования к транспортированию и хранению**
     1. Условия транспортирования

Требования к условиям транспортирования не предъявляются.

* + 1. Условия хранения

Обеспечение свободного доступа к проекту в репозитории до окончания срока учебы.

* + 1. Сроки хранения

Срок хранения – до окончания срока учебы.

1. **Требования к программной документации**

Определяются заданием на лабораторную работу.

1. **Стадии и этапы разработки**

Определяются заданием на лабораторную работу.

1. **Порядок контроля и приемки**

Определяются заданием на лабораторную работу.