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 №10  по дисциплине «Алгоритмы и структуры данных»  Тема «Разработка компьютерной игры Крестики-нолики» | | | | | | |
|  | | |  | |  | | | |
|  | | Исполнитель  студент гр. ИСТбд-22  Самборский Д.И.  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2024 г. | | | |
|  | | 2024 | | | | | | |

**Руководство программиста.**

**Руководство программиста на разработку программы "Крестики-нолики".**

1. **Общее описание:**

Данное руководство предназначено для программистов, работающих с кодом игры "Крестики-нолики", реализованной на языке Python и написанный на базе библиотеки Tkinter. Руководство содержит информацию о ключевых функциях, структуре программы и алгоритмах, используемых в ней.

1. **Структура программы:**
2. **Основной класс: TicTacToe**

Инициализация **-\_\_init\_\_**

self.root: Главное окно приложения.

self.center\_window: Центрирует окно на экране.

self.game\_board: Логическое игровое поле 3×3, где хранятся ходы.

self.current\_player: Указывает, чей сейчас ход (начинается с "X").

self.buttons: Матрица кнопок интерфейса, связанных с игровым полем.

self.create\_board: Создает графическое представление игрового поля.

1. **Методы для работы с графическим интерфейсом**

center\_window: Центрирует главное окно на экране с заданной шириной и высотой

create\_board: Создает кнопки для каждой ячейки поля 3×3, каждая кнопка привязывается к функции player\_move.

1. **Логика игры**

**player\_move()**

Отвечает за ход игрока:

Проверяет, можно ли сделать ход в указанной ячейке.

Помечает ячейку "X" и отключает кнопку.

Проверяет победу игрока или ничью.

Передает ход компьютеру, если игра продолжается

**ai\_move**

Отвечает за ход компьютера:

Вызывает find\_best\_move для определения оптимального хода.

Проверяет победу или ничью

**find\_best\_move**

Логика для определения лучшего хода компьютера:

Проверяет возможность своей победы за текущий ход.

Блокирует потенциальную победу игрока.

Следует приоритету углов или центра.

Выбирает первую доступную ячейку, если других вариантов нет

**check\_winner()**

Проверяет, выиграл ли указанный игрок (строки, столбцы, диагонали)

**check\_tie**

Проверяет, заполнены ли все клетки без победителя (ничья).

**reset\_board**

Сбрасывает игровое поле и активирует все кнопки.

1. **Основные свойства игры**

"X" — игрок, начинает первым.

"O" — компьютер с продуманной стратегией:

Выигрывает при возможности.

Блокирует игрока.

Делает приоритетный ход (углы, центр).

Эта структура организует программу так, чтобы четко разделять логику игры и графический интерфейс.

1. **Основные алгоритмы:**
2. **Алгоритм хода игрока (**player\_move**):**
3. Проверить, свободна ли выбранная ячейка:

Если да, пометить её "X".

Заблокировать кнопку в графическом интерфейсе.

1. Проверить:

Победил ли игрок с помощью check\_winner.

Если да, вывести сообщение о победе и сбросить игровое поле.

Если ничья (check\_tie):

Вывести сообщение о ничьей и сбросить игровое поле.

1. Если игра продолжается, передать ход компьютеру (ai\_move).
2. **Алгоритм хода компьютера (ai\_move):**
3. Использовать метод find\_best\_move, чтобы определить оптимальный ход:

Попробовать победить, сделав свой ход.

Если это невозможно, блокировать возможную победу игрока.

Если и это не требуется, сделать приоритетный ход (углы или центр).

Если углы и центр заняты, выбрать любую свободную клетку.

1. Сделать ход компьютера:

Пометить ячейку "O".

Заблокировать кнопку.

1. Проверить:

Победил ли компьютер с помощью check\_winner.

Если да, вывести сообщение о поражении игрока и сбросить игровое поле.

Если ничья (check\_tie):

Вывести сообщение о ничьей и сбросить игровое поле.

1. **Алгоритм проверки победы (check\_winner):**

Входные данные: Игрок (player), которого нужно проверить ("X" или "O").

Проверка победы:

1. Проверить строки:

Все клетки строки принадлежат игроку.

1. Проверить столбцы:

Все клетки столбца принадлежат игроку.

1. Проверить диагонали:

Все клетки главной диагонали принадлежат игроку.

Все клетки побочной диагонали принадлежат игроку.

Если хотя бы одно из условий выполнено, вернуть True.

1. **Алгоритм проверки ничьей (check\_tie):**
2. Проверить каждую клетку игрового поля:

Если хотя бы одна клетка пустая (None), игра продолжается, в озвращается False.

1. Если все клетки заполнены, вернуть True (ничья).
2. **Алгоритм определения лучшего хода компьютера (find\_best\_move):**
3. Проверка на возможность выигрыша:

Перебрать все свободные клетки.

Симулировать ход компьютера ("O") в каждую клетку.

Если после хода компьютера check\_winner("O") возвращает True, выбрать эту клетку.

Если нет, вернуть состояние клетки в исходное.

1. Проверка на необходимость блокировки игрока:

Перебрать все свободные клетки.

Симулировать ход игрока ("X") в каждую клетку.

Если после хода игрока check\_winner("X") возвращает True, выбрать эту клетку.

Если нет, вернуть состояние клетки в исходное.

1. Приоритетный ход (стратегия):

Попробовать занять углы или центр:

Проверить клетки [(0, 0), (0, 2), (2, 0), (2, 2), (1, 1)].

Если клетка свободна, выбрать её.

1. Если нет других вариантов:

Сделать любой доступный ход (перебрать поле и выбрать первую свободную клетку).

1. **Рекомендации программисту:**

Программа хорошо структурирована и документирована, что упрощает ее модификацию и отладку.

Графический интерфейс простой и интуитивно понятный, что делает программу удобной в использовании.

**Заключение:**

Данная программа предоставляет пользователям возможность сыграть в классическую игру "Крестики-нолики", и благодаря использованию алгоритма find\_best\_move, компьютер будет соперником на уровне с человеком. Программа может быть расширена дополнительными функциями.