**1. Назначение и условия применения программы**

**1.1 Назначение и функции, выполняемые приложением**

Приложение "Крестики-нолики" предназначено для игры в классическую игру крестики-нолики на двумерной сетке 3x3.

Основные функциональные возможности приложения:

* Игра против компьютера или другого игрока
* Интерактивный игровой интерфейс с возможностью выбора ячеек
* Автоматическое определение победителя или ничьей
* Отображение текущего счета игры
* Возможность начать новую игру

**1.2 Условия, необходимые для использования приложения**

Для использования приложения необходимы следующие условия:

Операционная система: Windows 7, Windows 10, Linux

Python 3.8 и выше

Библиотеки: tkinter

Рекомендуемая среда разработки: PyCharm или любая другая IDE с поддержкой Python

**2. Характеристики программы**

**2.1 Характеристики приложения**

* Количество значимых строк кода: около 400
* Количество основных классов: 7 (Window, TicTacToe, Board, Cell, Score, ResultDisplay)
* Количество основных алгоритмов: 4 (проверка победителя, ход компьютера, Минимакс, оценка состояния игры)

Используемые библиотеки:

tkinter: для создания графического интерфейса

Работа приложения:

Интерфейс игры представлен на рисунке 1. При запуске открывается главное окно с игровым полем, панелью счета и кнопками управления. Игрок может выбирать режим игры (против компьютера или против другого игрока) и делать ходы, кликая на ячейки поля. Игра автоматически определяет победителя или ничью и обновляет счет.

Интерфейс игры Крестики-Нолики


Рисунок 1 - Интерфейс игры Крестики-Нолики

2.2 **Особенности реализации приложения**

Основные классы:

* Window: основное окно приложения
* TicTacToe: основной класс игры, управляющий логикой и интерфейсом
* Board: представляет игровое поле
* Cell: представляет отдельную ячейку игрового поля
* Score: отображает счет игры
* ResultDisplay: отображает результат игры

Выбор этих классов обусловлен логическим разделением функциональности и обеспечением модульности кода.

**3. Обращение к программе**

Основные методы и алгоритмы:

1. Метод minimax: реализует алгоритм минимакса для компьютера

def minimax(board, is\_max, player, opponent):

score = evaluate(board, player, opponent)

if score == 1:

return score

if score == -1:

return score

if not is\_moves\_left(board):

return 0

size = len(board)

if is\_max:

value = -1

for row in range(size):

for col in range(size):

if board[row][col] == ' ':

board[row][col] = player

value = max(value, minimax(board, not is\_max, player, opponent))

board[row][col] = ' '

return value

else:

value = 1

for row in range(size):

for col in range(size):

if board[row][col] == ' ':

board[row][col] = opponent

value = min(value, minimax(board, not is\_max, player, opponent))

board[row][col] = ' '

return value

Метод find\_best\_move: находит лучший ход для компьютера

def find\_best\_move(board, player):

opponent = 'X' if player == 'O' else 'O'

best\_val = -1

best\_move = (-1, -1)

size = len(board)

for i in range(size):

for j in range(size):

if board[i][j] == ' ':

board[i][j] = player

move\_val = minimax(board, False, player, opponent)

board[i][j] = ' '

if move\_val > best\_val:

best\_move = (i, j)

best\_val = move\_val

return best\_move

Метод check\_winner: проверяет наличие победителя

def check\_winner(self):

if self.board.check\_winner(self.current\_player.get()):

self.winner = self.current\_player.get()

self.game\_over = True

elif self.board.check\_draw():

self.game\_over = True

Метод on\_cell\_click: обрабатывает клик по ячейке

def on\_cell\_click(self, row, col):

if self.game\_over or not self.board.is\_cell\_empty(row, col):

return

if self.game\_mode.get() == "Против игрока":

self.board.set\_player(self.current\_player.get(), row, col, self.current\_player\_color)

self.change\_player()

else:

self.board.set\_player(self.current\_player.get(), row, col, self.current\_player\_color)

self.change\_player()

self.ai\_move()

self.change\_player()

**4. Сообщения**

Приложение использует визуальные индикаторы для отображения состояния игры:

* Текущий игрок отображается в верхней части экрана
* Победитель или ничья отображаются на экране результатов
* Счет игры обновляется автоматически