МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Измерительно-вычислительные комплексы»

ДОКУМЕНТАЦИЯ

на лабораторную работу

по дисциплине «Алгоритмы и структуры данных»

Тема «Игра Крестики-нолики»

Исполнитель

студентка гр. ИСТбд-21

Феофанова П.А.

«\_\_\_»\_\_\_\_\_\_\_\_\_ 2024 г.

2024

Содержание

Техническое задание...............................................................................................3

Пояснительная записка...........................................................................................7

Руководство программиста...................................................................................10

Mind map.................................................................................................................15

Чек-лист..................................................................................................................16

Набор тест-кейсов..................................................................................................17

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

по дисциплине «Алгоритмы и структуры данных»

Тема «Игра Крестики-нолики»

Исполнитель

студентка гр. ИСТбд-21

Феофанова П.А.

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

2024

**Введение**

Разрабатываемое приложение «Крестики-нолики» реализует игру крестики-нолики. Крестики-нолики – логическая игра между двумя противниками на квадратном поле 3 на 3 клетки. Один из игроков играет «крестиками», второй - «ноликами».

**1. Основания для разработки**

Основание для разработки является учебный план направления 09.03.02 "Информационные системы и технологии".

**2. Требования к программе или программному изделию**

**2.1. Функциональное назначение**

**Функциональное назначение**

Требуется разработать приложение для игры в «крестики-нолики» с графическим интерфейсом в среде Windows.

**2.2 Требования к функциональным характеристикам**

2.2.1 Требования к структуре приложения

Приложение должно соответствовать следующим правилам игры.

**Поле и игроки.** Игроки по очереди ставят на свободные клетки поля 3х3 знаки (один всегда крестики, другой всегда нолики). Первый ход делает игрок, ставящий крестики.

**Цель игры.** Поставить в один ряд три крестика или нолика и не дать сопернику сделать то же самое.

**Завершение игры.** Первый, выстроивший в ряд 3 своих фигуры по вертикали, горизонтали или большой диагонали, выигрывает. Если игроки заполнили все 9 ячеек и оказалось, что ни в одной вертикали, горизонтали или большей диагонали нет трех одинаковых знаков, партия считается закончившейся вничью.

2.2.2 Требования к составу функций приложения

В приложении должны быть реализованы в графическом режиме следующие основные функции:

- отрисовка игрового поля;

- интерактивные прием, проверка правильности и отрисовка хода пользователя;

- проверка окончания игры;

- вычисление, проверка правильности и отрисовка хода компьютера;

- информирование пользователя о победителе.

**2.3 Требования к надёжности**

Приложение должно обеспечивать стабильную работу без выходов из строя и ошибок. В случае возникновения сбоев и ошибок, оно должно автоматически восстанавливаться с сохранением всех данных.

**2.4 Требования к информационной и программной совместимости**

Версия операционной системы: Windows 7, 8, 8.1, 10, 11

Язык программирования: Python

Используемые библиотеки: графическая библиотека Tkinter

**2.5 Требования к маркировке и упаковке**

Определяются заданием на лабораторную работу.

**2.6 Требования к транспортированию и хранению**

2.6.1 Условия транспортирования

Требования к условиям транспортирования не предъявляются.

2.6. 2 Условия хранения

Проект будет храниться в репозитории на сайте github.com по ссылке https://github.com/pollinaff/-1.git

2.6. 3 Сроки хранения

Срок хранения – не ограничен.

**3. Требования к программной документации**

В состав программной документации должны входить:

* Техническое задание
* Руководство программиста
* Пояснительная записка

**4. Стадии и этапы разработки**

1.Анализ игры «Крестики-нолики».

2.Создание интерфейса.

3.Реализация игры.

4.Тестирование и отладка.

5.Сдача лабораторной работы.

**5. Порядок контроля и приёмки**

Определяются заданием на лабораторную работу.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Измерительно-вычислительные комплексы»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

на лабораторную работу

по дисциплине «Алгоритмы и структуры данных»

Тема «Игра Крестики-нолики»

Исполнитель

студентка гр. ИСТбд-21

Феофанова П.А.

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

2024

**Введение**

Программа представляет из себя приложение по игре «Крестики-нолики» с графическим интерфейсом в среде Windows.

**2.Основания для разработки**

Основание для разработки является учебный план направления 09.03.02 "Информационные системы и технологии".

**3.Функциональное назначение**

* create\_widgets: метод, который создает и размещает все графические элементы.
* сomputer\_move: метод, отвечающий за выбор наилучшего хода для компьютера.
* сheck\_winner: метод, который проверяет, выиграл ли игрок.
* reset\_game: метод, который сбрасывает игру до начального состояния.
* toggle\_mode: метод, который переключает игру между «игрок против компьютера» и «игрок против игрока».

**4.Структура программы**

Классы:

1. TicTacToe: содержит всю логику и интерфейс игры «Крестики-нолики».

**5. Процесс разработки**

Разработка проекта включала следующие этапы:

1. Постановка задачи: определение целей проекта и основных требований.
2. Проектирование структуры: определение классов, модулей, методов.
3. Реализация процесса: разработка программного кода для игровой логики и графического интерфейса.
4. Тестирование: проверка всех действий на правильность.
5. Улучшение интерфейса: реализация главного меню.

**6. Тестирование**

Тестирование проекта было проведено в несколько этапов:

* **Функциональное тестирование:** проверка основных функций.
* **Тестирование на устойчивость:** запуск проекта в различных сценариях для проверки возможных ошибок и неправильного поведения.
* **Ручное тестирование интерфейса:** проверка работы главного меню.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Измерительно-вычислительные комплексы»

РУКОВОДСТВО ПРОГРАММИСТА

на лабораторную работу

по дисциплине «Алгоритмы и структуры данных»

Тема «Игра Крестики-нолики»

Исполнитель

студентка гр. ИСТбд-21

Феофанова П.А.

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2024 г.

2024

**1. Назначение и условия применения программы**

**1.1 Назначение и функции, выполняемые приложением**

Программа представляет собой проложение с графическим интерфейсом для игры в «Крестики-нолики».

Основные функциональные возможности приложения:

* Визуализация игрового поля
* Реализация игры «игрок против компьютера»
* Реализация игры «пользователь против пользователя»
* Определение конца игры (победа или ничья)

**1.2 Условия, необходимые для использования приложения**

Версия операционной системы: Windows 7, 8, 8.1, 10, 11

Язык программирования: Python

Используемые библиотеки: графическая библиотека Tkinter

**2. Характеристики программы**

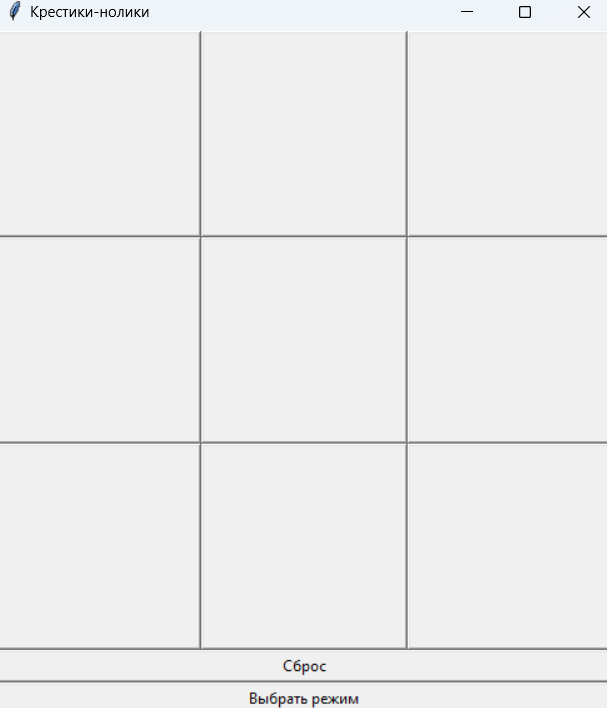
**2.1 Характеристики приложения**

При реализации графического интерфейса программа создает игровое поле. Взаимодействие с пользователем происходит при помощи мыши. В программе реализовано меню с кнопками: «Сброс», «Выбрать режим» / «Режим: игра против компьютера» / «Режим: игра против пользователя».

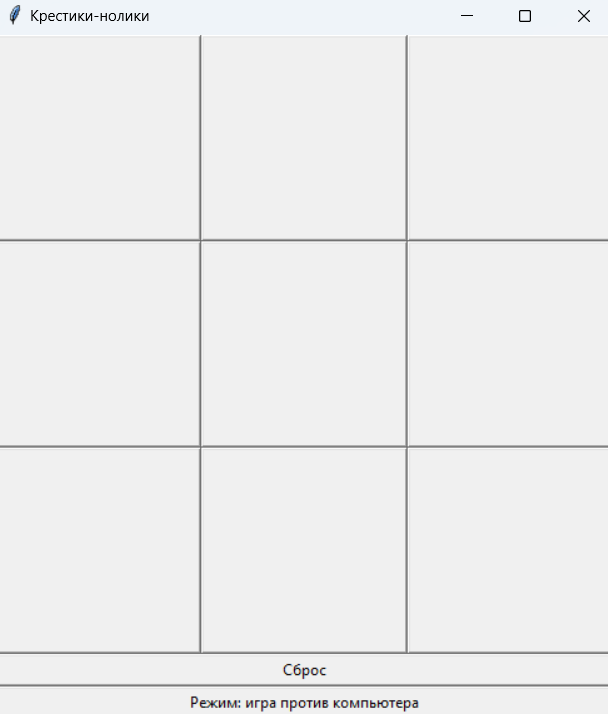
Программа реализована при помощи класса TicTacToe, который содержит всю логику и интерфейс игры «Крестики-нолики», в нем реализовано:

* \_\_init\_\_: метод, который вызывается при создании экземпляра класса TicTacToe.
* create\_widgets: метод, который создает и размещает все графические элементы.
* on\_button\_click: метод, который вызывается при нажатие на любую кнопку игрового поля.
* сomputer\_move: метод, отвечающий за выбор наилучшего хода для компьютера.
* minimax: определяет параметры: текущее состояние доски, глубина рекурсии, флаг максимизации.
* сheck\_winner: метод, который проверяет, выиграл ли игрок.
* reset\_game: метод, который сбрасывает игру до начального состояния.
* toggle\_mode: метод, который переключает игру между «игрок против компьютера» и «игрок против игрока».

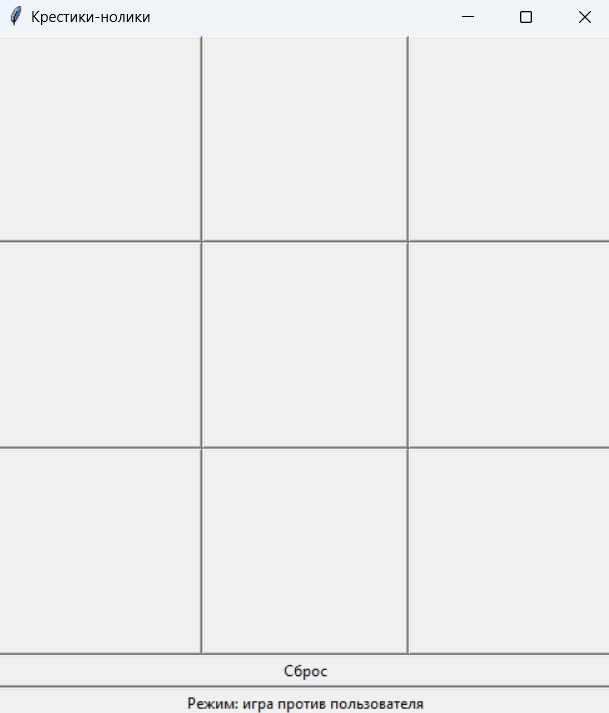
**2. 2. Описание интерфейса программы**



Для того, чтобы начать пользоваться программой необходимо выбрать режим игры, нажав на кнопку «Выбрать режим»:

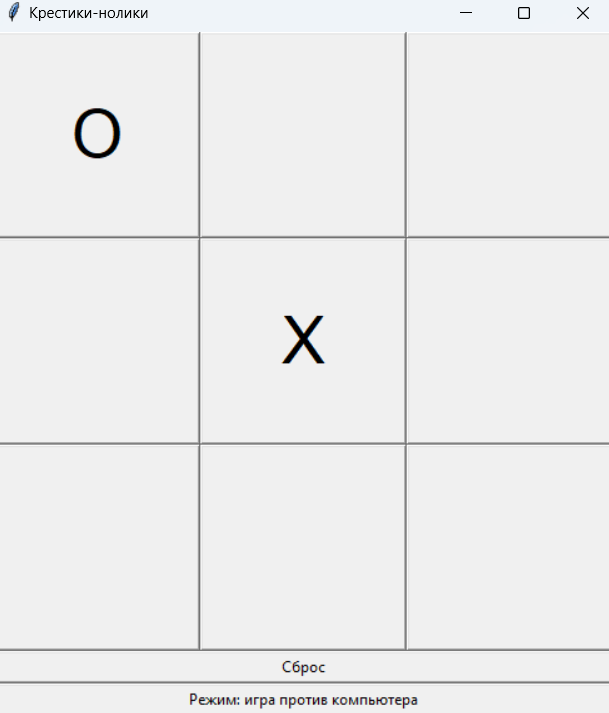


Либо «Режим: игра против компьютера»



Либо «Режим: игра против пользователя»

Далее необходимо сделать ход нажав на любую пустую клетку:



Для того, чтобы начать игру заново необходимо нажать на кнопку «Сброс».

**3. Обращение к программе**

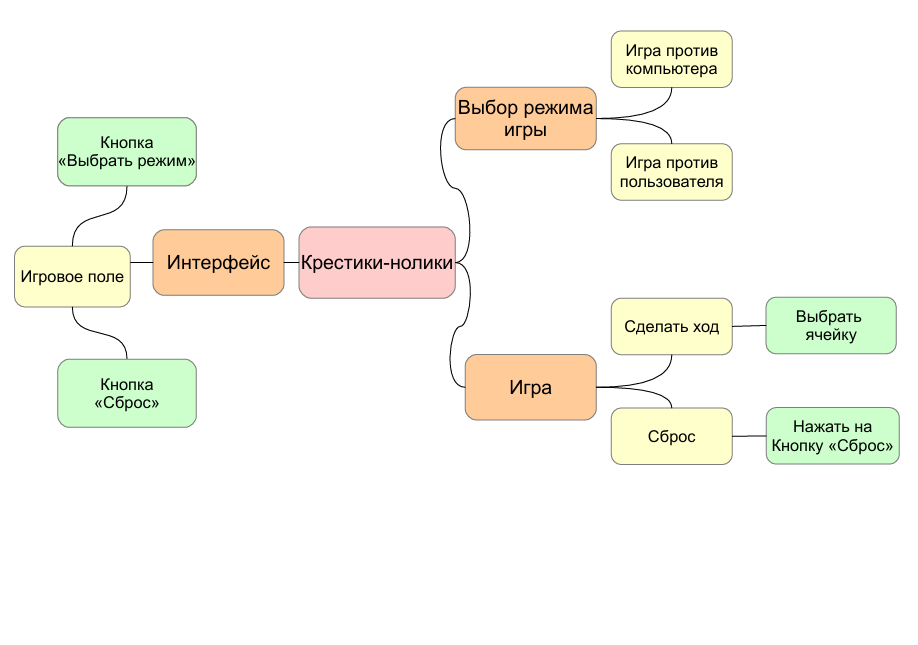
Tkinter – это кросс-платформенная событийно-ориентированная графическая библиотека Python, предназначенная для работы с библиотекой Tk. Библиотека Tk содержит компоненты графического интерфейса пользователя (GUI).

**4. Сообщения**

Приложение выдает следующие сообщения:

При ошибке загрузки файла: «Игрок О выиграл!», «Игрок X выиграл!» и «Ничья!».

**Mind map**



**Чек-лист**

|  |  |  |
| --- | --- | --- |
| Описание | Пример | Результат |
| Запуск игры | Запуск программы | Открытие игрового поля |
| Выбор режима игры | Нажать на кнопку «Выбрать режим» | Режим игры меняется на «Режим: игра против компьютера» |
| Ход игрока | Нажатие на любую пустую ячейку | В ячейке появляется «X» |
| Ход компьютера | После хода игрока | В определенной ячейке появляется «O» |
| Победа игрока | Три «Х» в ряд | Появляется окно «Игрок X выиграл!» |
| Победа компьютера | Три «О» в ряд | Появляется окно «Игрок О выиграл!» |
| Ничья | Все клетки заполнены, но условие выигрыша не выполнено | Появляется окно «Ничья!» |
| Сброс игры | Нажать на кнопку «Сброс» | Игровое поле очищается, игра начинается заново |
| Смена режима | Нажать на кнопку «Выбрать режим» | Режим игры меняется на «Режим: игра против пользователя» |
| Ход второго игрока | Нажатие на любую пустую ячейку | В ячейке появляется символ второго игрока |

**Набор тест-кейсов**

**1. Начало игры и интерфейс**

ТК-1: Запуск игры

Шаги:

1. Запустить приложение

Ожидаемый результат:

* Отображается пустая игровая доска 3x3
* В нижней части экрана отображаются кнопки «Сброс» и «Выбрать режим»

ТК-2: Выбор режима игры

(игра запущена)

Шаги:

1. Нажать на кнопку «Выбрать режим»

2. Выбрать режим «Режим: игра против компьютера»

Ожидаемый результат:

* Режим игры меняется на «против компьютера»

**2. Игровой процесс**

ТК-3: Ход игрока в режиме «Против пользователя»

(выбран режим «Режим: игра против пользователя»)

Шаги:

1. Нажать на пустую клетку игрового поля

Ожидаемый результат:

* В клетке появляется символ игрока (X или O)
* Ход переходит к другому игроку

ТК-4: Ход игрока в режиме "Против компьютера"

(выбран режим «Режим: игра против компьютера»)

Шаги:

1. Кликнуть на пустую клетку игрового поля

Ожидаемый результат:

* В выбранной клетке появляется символ игрока «Х»
* Компьютер делает ход, ставя «O» в одну из пустых клеток

ТК-5: Победа игрока

(игра заканчивается)

Шаги:

1. Сделать последний ход для победы

Ожидаемый результат:

* Игра заканчивается
* Появляется окно «Игрок X выиграл!» или «Игрок O выиграл!».

ТК-6: Ничья

(осталась одна свободная клетка, но она не приведет к победе)

Шаги:

1. Заполнить последнюю клетку

Ожидаемый результат:

* Игра заканчивается
* Появляется окно «Ничья!»

ТК-7: Сброс игры

(игра завершена)

Шаги:

1. Нажать кнопку "Сброс"

Ожидаемый результат:

* Игровое поле очищается