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

Булычева А.Е

«\_\_\_»\_\_\_\_\_\_\_\_\_ 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

Булычева А.Е

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 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.** Участники поочередно заполняют пустые клетки игрового поля размером 3х3 своими знаками (один всегда ставит крестики, другой — нолики). Первым ходит тот, кто играет за крестики.

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

**Завершение игры.** Победителем становится тот, кто первым выстроит три свои фигуры подряд по вертикали, горизонтали или любой из двух диагоналей. Если же после заполнения всех девяти клеток ни у одного игрока не образуется такая линия, то партия заканчивается ничьей.

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 работать стабильно, без аварийных завершений и ошибок. При возникновении сбоев и неполадок, оно должно автоматически восстанавливать свою работоспособность, сохраняя при этом все данные.

**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/AleksandraBulycheva/public.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

Булычева А.Е

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

2024

**Введение**

Программа является приложением для игры в «Крестики-нолики» с графическим интерфейсом, разработанным для операционной системы Windows.

**2.Основания для разработки**

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

**3.Функциональное назначение**

* dismiss: эта функция принимает объект окна.
* game\_over\_popup:эта функция создает всплывающее окно, информирующее об окончании игры.
* options\_window.grab\_set(): функция создает окно настроек игры.
* сheck\_winner: метод, который проверяет, выиграл ли игрок.
* computer\_move: функция, отвечающая за выполнение хода компьютера.
* player\_move: эта функция обрабатывает ход игрока.
* start\_new\_game: эта функция запускает новую игру.

**4.Структура программы**

Классы:

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

**5. Процесс разработки**

Разработка проекта включала следующие этапы:

1. Определение задач: формулирование целей проекта и ключевых требований.

2. Проектирование архитектуры: создание схемы классов, модулей и методов.

3. Реализация функционала: написание кода для игровой логики и графического интерфейса.

4. Тестирование: проверка корректности всех функций.

5. Улучшение пользовательского опыта: добавление 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

Булычева А.Е

«\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 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, ничья или поражение)

**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 создании графического интерфейса программа формирует игровое поле. Пользователь взаимодействует с программой посредством мыши. В программе предусмотрено меню с кнопками: «Сброс», «Выбор режима» / «Режим: игра с компьютером» / «Режим: игра с другим игроком».

Программа создана с использованием класса TicTacToe, который включает в себя всю логику и интерфейс игры «Крестики-нолики». В этом классе реализованы следующие методы:

- \_\_init\_\_: конструктор, вызываемый при создании нового экземпляра класса TicTacToe.

- create\_widgets: метод, создающий и размещающий все графические компоненты.

- on\_button\_click: метод-обработчик нажатий на кнопки игрового поля.

- computer\_move: метод для выбора оптимального хода компьютера.

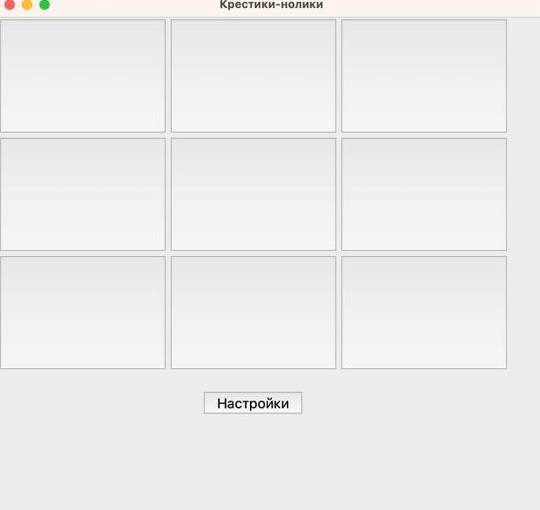
- minimax: функция, определяющая оптимальное решение на основе текущего состояния доски, глубины рекурсии и флага максимизации.

- check\_winner: метод проверки победы одного из игроков.

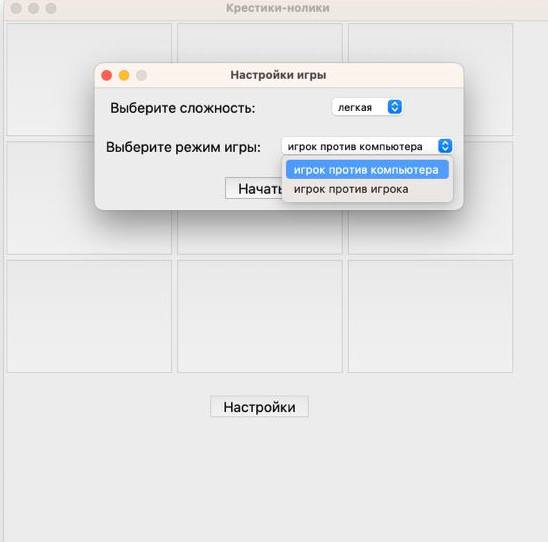
- reset\_game: метод сброса игры к начальному состоянию.

- toggle\_mode: метод переключения режимов игры («игрок против компьютера» и «игрок против игрока»).

**2. 2. Описание интерфейса программы**

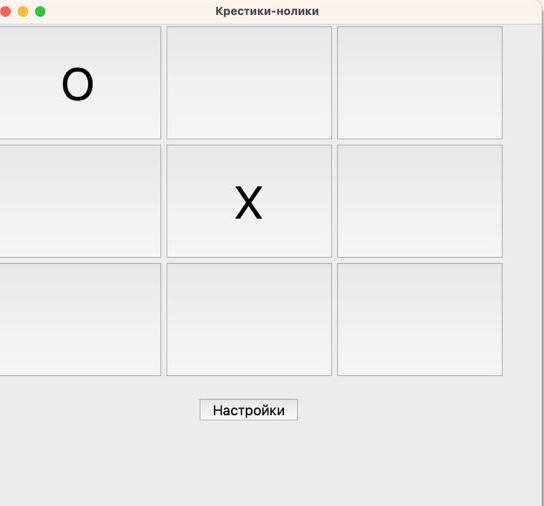
****

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



«Режим: игра против компьютера» или Режим: игра против игрока»

Далее необходимо сделать ход нажав на любую пустую клетку:



**3. Обращение к программе**

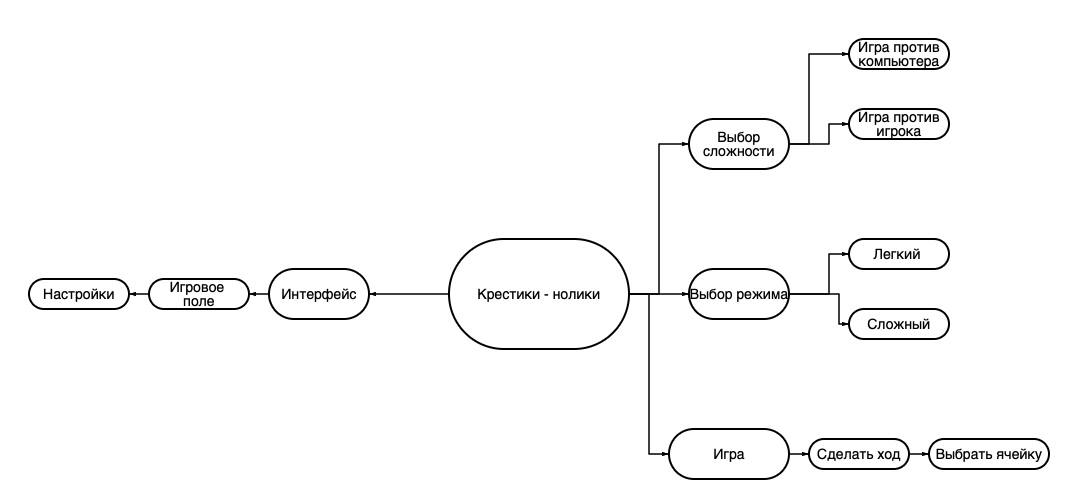
Tkinter — это кросс-платформенная библиотека Python для создания графических интерфейсов, основанная на библиотеке Tk. Она предоставляет инструменты для разработки GUI-приложений, включая различные компоненты пользовательского интерфейса.

**4. Сообщения**

Приложение выдает следующие сообщения:

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

**Mind map**

****

**Чек-лист**

|  |  |  |
| --- | --- | --- |
| Описание | Пример | Результат |
| Запуск игры | Запуск программы | Открытие игрового поля |
| Выбор режима игры | Нажать на кнопку «Настойки» | Выбрать режим |
| 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 «Компьютер выиграл!» |
| Ничья | Все клетки заполнены, но условие выигрыша не выполнено | Появляется окно «Ничья!» |
| Смена режима | Нажать на кнопку «Настройки» | Режим игры меняется на «Режим: игра против игрока» |
| 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 кнопка «Настройки»

ТК-2: Выбор режима игры

(игра запущена)

Шаги:

1. Нажать на кнопку «Настройки»

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 игрока»

(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 «Компьютер выиграл!» или «Игрок 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!»