**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

# Лабораторная работа

**По дисциплине «Алгоритмы и структуры данных»**

**Тема** «Игра Крестики - нолики»

|  |  |
| --- | --- |
| **Подп. и** |  |
| **Инв** |  |
| **Вза** |  |
| **Подп. и** |  |
| **Инв.** |  |

# Пояснительная записка

**Исполнитель**: студент гр. ИСТбд-21

*Голиков Олег Дмитриевич*

2024

**Введение**

Программа разработана для игры в Крестики-нолики с графическим интерфейсом и возможностью выбора режима игры: против компьютера или другого игрока.

**Основания для разработки:**

Задание для лабораторной работы по учебному плану направления «Информационные системы и технологии».

**Функциональное назначение:**

* reset\_game – сброс текущей игры, обнуление игрового поля и установка текущего игрока.
* set\_first\_player – установка первого игрока (человек или компьютер) перед началом игры.
* play\_vs\_computer – начало игры против компьютера, автоматически выполняющий первый ход, если он играет первым.
* play\_vs\_human – начало игры против другого игрока.
* click – обработка нажатия кнопки на игровом поле, обновление состояния игры и проверка на победителя.
* computer\_move – осуществление хода компьютера с использованием алгоритма минимакс.
* minimax – реализация алгоритма минимакс для выбора наилучшего хода для компьютера.
* check\_winner – проверка наличия победителя на игровом поле.

**Структура программы**

**Классы:**

**TicTacToe** – основной класс, отвечающий за логику игры, управление состоянием игрового поля, ходами игроков и взаимодействием с графическим интерфейсом.

**Стадии и этапы разработки**

Проект разработан по следующим этапам:

1. **Определение задач и требований** – установление ключевых целей и функциональных возможностей проекта.
2. **Архитектурное проектирование** – выделение необходимых классов и методов для построения программы.
3. **Кодирование** – реализация логики игры, алгоритма минимакс для компьютера и создание графического интерфейса.
4. **Тестирование** – проверка программы на наличие ошибок и соответствие заданным целям.
5. **Оптимизация интерфейса** – создание удобного и интуитивного игрового поля и элементов управления.
6. **Документация** – описание работы программы для последующего сопровождения и использования.

**Тестирование**

1. **Подготовка тестовых сценариев** – создание игровых ситуаций для проверки всех возможных исходов игры.
2. **Модульное тестирование** – тестирование каждого метода класса TicTacToe отдельно.
3. **Функциональное тестирование** – проверка работы игрового поля, корректности ходов игрока и компьютера, определение победителя.
4. **Интеграционное тестирование** – проверка взаимодействия методов класса TicTacToe и корректности отображения игрового процесса.