**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное бюджетное образовательное учреждение

высшего образования

**«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

Кафедра «Измерительно-вычислительные комплексы»

Приложение <Го-Бан>

**РУКОВОДСТВО ПРОГРАММИСТА**

Р.02069337.<номер зачётки>-<2-зн. номер варианта> РП-<2-зн. номер редакции>

Листов <количество листов документа>

**Руководитель разработки**:

профессор каф. ИВК, к.т.н., доцент

*Шишкин Вадим Викторинович*

« » 2022 г.

**Исполнитель**:

студент гр. ИСТбд-21

*<ФИО>*

« » 2022 г.

**2022**

**1. Назначение и условия применения программы**

**1.1 Назначение и функции, выполняемые приложением**

Назначение приложения: Получение игрового опыта в игру Гобан.

Свод правил:

* Игра проходит на доске 15х15 клеток
* Чтобы победить в игре, нужно поставить 5 фишек в ряд, по горизонтали, вертикали или диагонали
* Ходы происходят по очереди
* Камни ставятся в угол клетки

Функциональный возможности:

* Игра в Гобан против компьютера
* Выбор цвета своих фишек
* Сохранение результатов игр

**1.2 Условия, необходимые для использования приложения**

Версия ОС: Windows 7 / 8 /10/11 (64bit)

Установленный интерпретатор языка Python > 3.7

Для поддержки, дальнейшего развития и отладки рекомендуется использовать среду разработки PyCharm 2022.3

Встроенные библиотеки языка Python, необходимые для работы приложения: sys, random, collections, os.

Устанавливаемые библиотеки языка Python, необходимые для работы приложения: pygame 2.1.3, pygame-menu 4.3.1

**2. Характеристики программы**

**2.1 Характеристики приложения**

Количество значимых строк программного кода, **≈** 600

Структуры данных: списки, двумерные списки, словари и кортежи.

Библиотеки:

* random предоставляет функции для генерации случайных чисел, букв, случайного выбора элементов последовательности.
* collections даёт доступ к специальным типам данных на основе коллекций python
* pygame набор модулей (библиотек) языка программирования Python, предназначенный для написания компьютерных игр и мультимедиа-приложений
* pygame-menu это библиотека python-pygame для создания меню и графических интерфейсов.

Средства контроля ввода:

* Для ввода имени есть ограничение, не больше 256 символов
* Для выбора цвета, необходимо использовать селектор

Внешний вид приложения:

Главный экран представлен на рисунке 1

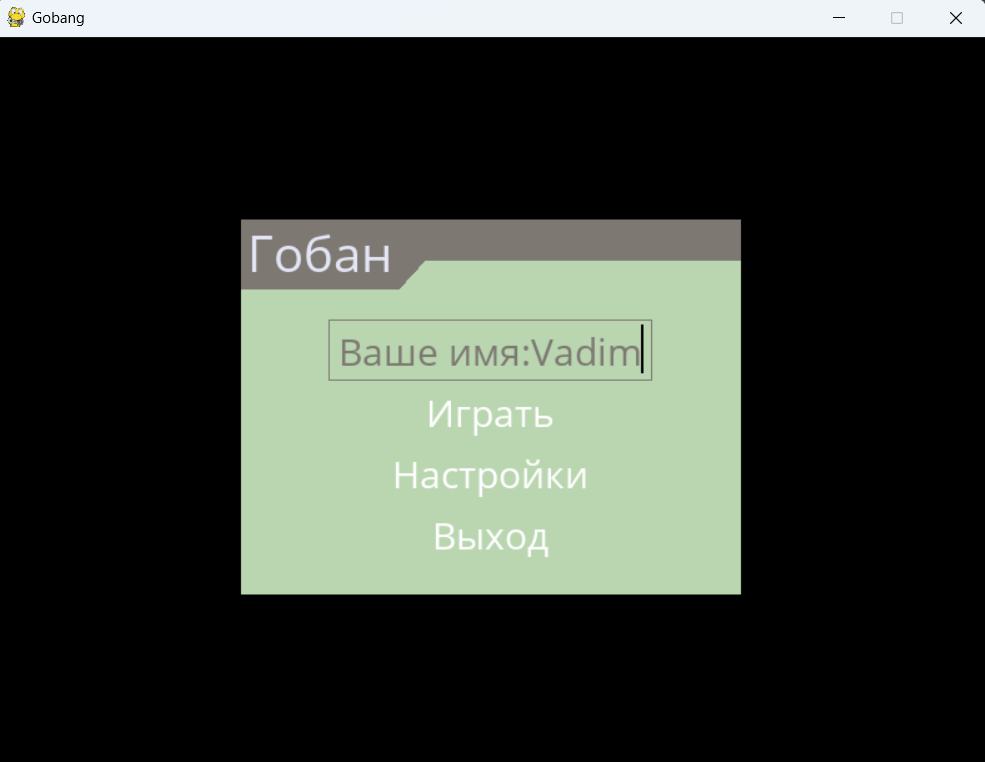


Рисунок 1 - Главное меню

Экраны с игровым полем представлены на рисунке 2 и 3

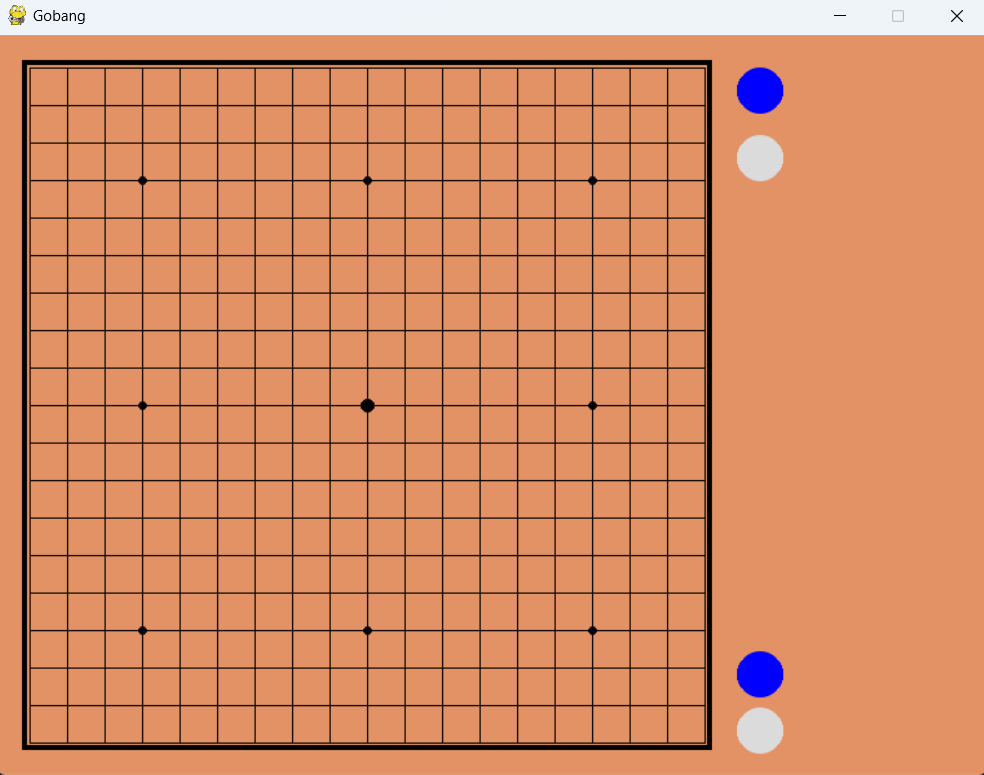


Рисунок 2 - Начало игры

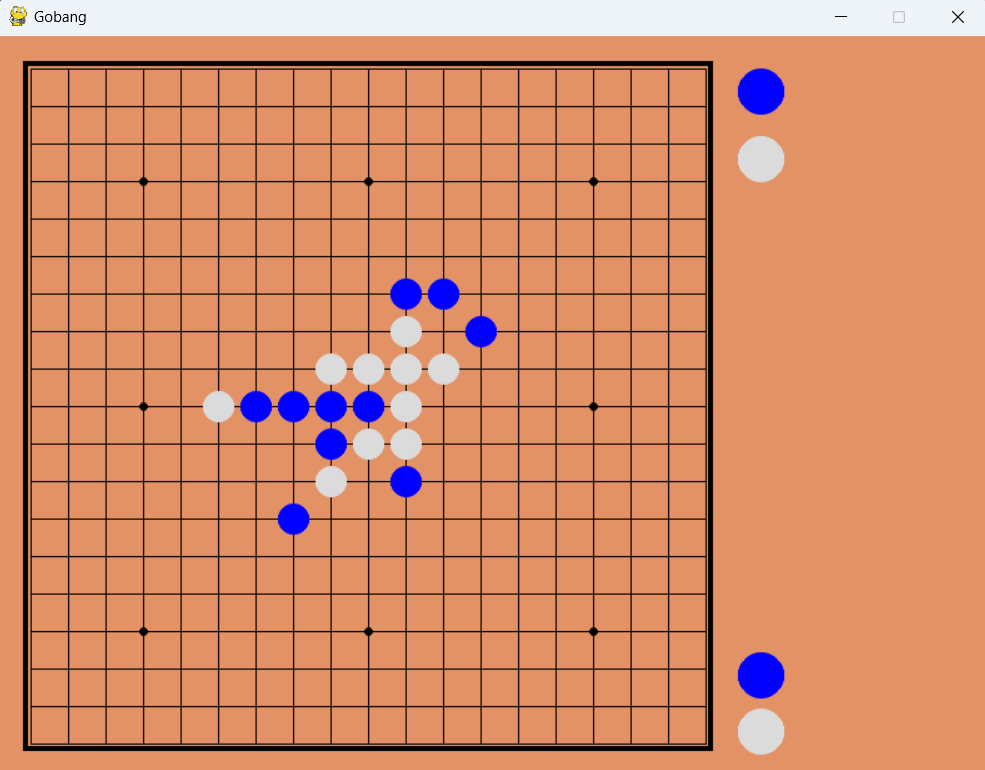


Рисунок 3 - Игра после нескольких ходов

Экран окончания игры представлен на рисунке 4

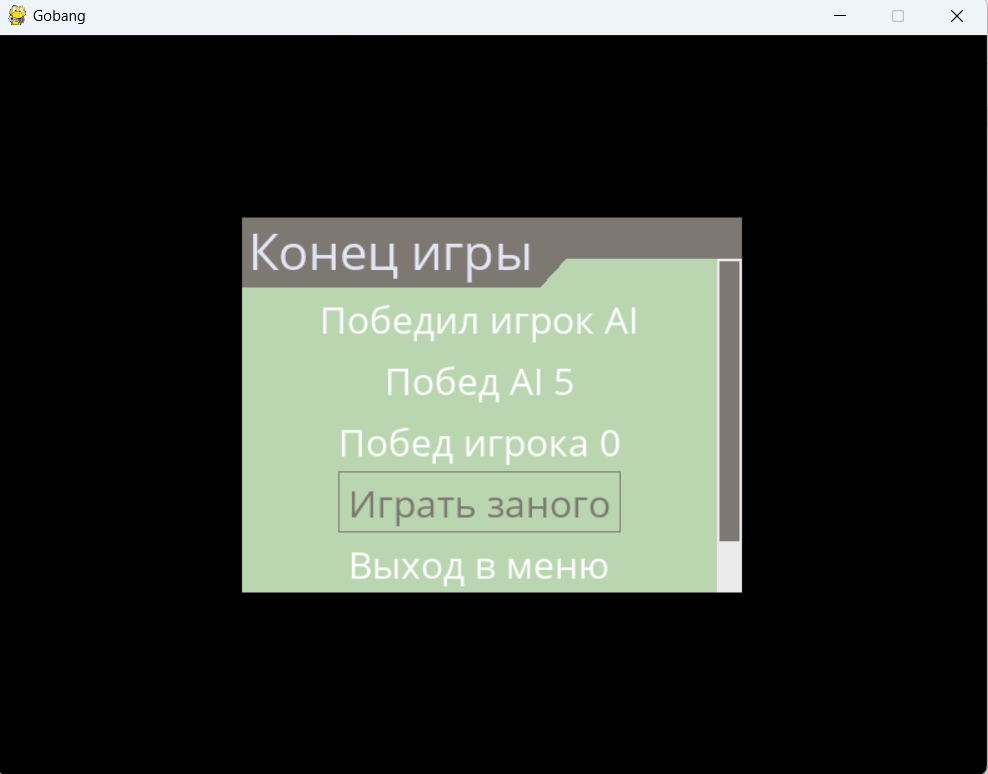


Рисунок 4 - Результат игр

Экран с настройками на рисунке 5

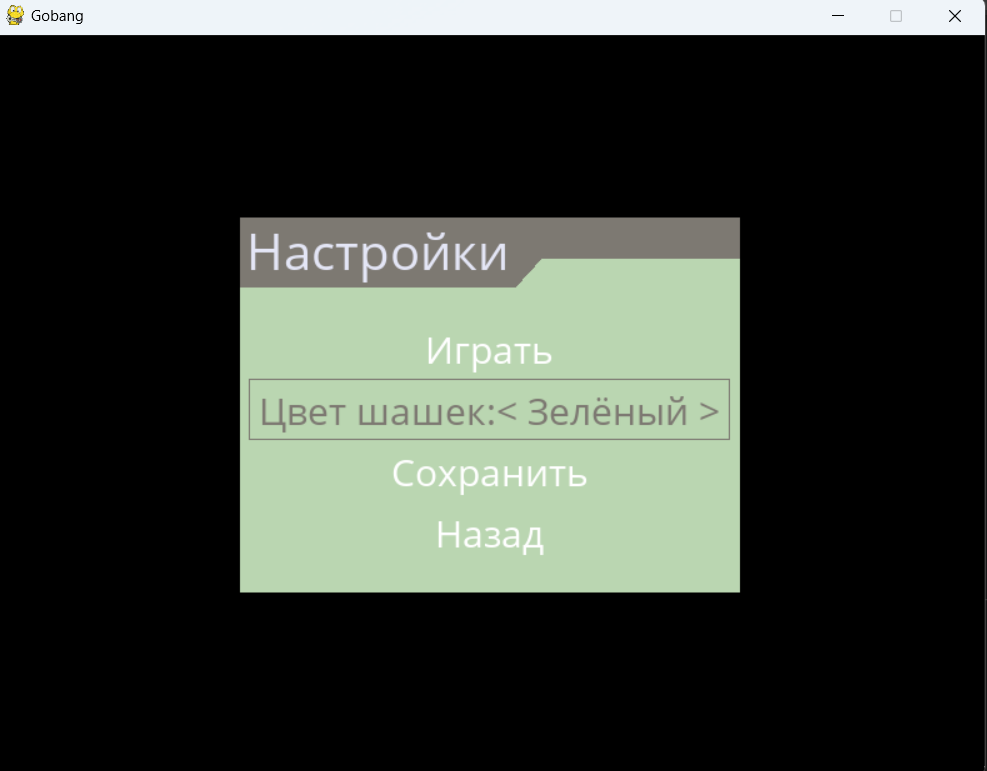


Рисунок 5 - Настройки игры

**2.2 Особенности реализации приложения**

Для реализации поля выбран двумерный список. На практике двумерный список, это список который в каждом своем элементе хранит список.

Для реализации данных о игроке, был выбран словарь. Словарь выбран для наглядности и читабельности кода. Так как ключ называется так же, как и то что хранит. Name хранит имя и т.д.

Для реализации игрока и точки на карте, был выбран кортеж с именем. Его большой плюс, что данные в нем не изменяемы и во время игры нельзя будет поменять игрока.

Алгоритм выбора хода AI, это алгоритм minimax. Правило принятия решений, используемое в теории игр, теории принятия решений, исследовании операций, статистике и философии для минимизации возможных потерь из тех, которые лицо, принимающее решение, не может предотвратить при развитии событий по наихудшему для него сценарию.

Альтернативные варианты, это алгоритмы negamax, negaScout и MTDF.

Алгоритм для шифрования RC5, это [блочный шифр](https://ru.wikipedia.org/wiki/%D0%91%D0%BB%D0%BE%D1%87%D0%BD%D1%8B%D0%B9_%D1%88%D0%B8%D1%84%D1%80), разработанный [Роном Ривестом](https://ru.wikipedia.org/wiki/%D0%A0%D0%B8%D0%B2%D0%B5%D1%81%D1%82,_%D0%A0%D0%BE%D0%BD%D0%B0%D0%BB%D1%8C%D0%B4_%D0%9B%D0%B8%D0%BD%D0%BD) с переменным количеством [раундов](https://ru.wikipedia.org/wiki/%D0%A0%D0%B0%D1%83%D0%BD%D0%B4_(%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D1%8F)), длиной блока и длиной [ключа](https://ru.wikipedia.org/wiki/%D0%9A%D0%BB%D1%8E%D1%87_(%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B3%D1%80%D0%B0%D1%84%D0%B8%D1%8F)). Это расширяет сферу использования и упрощает переход на более сильный вариант алгоритма.

Алгоритм представляет собой сеть Фейсстеля, в каждом раунде происходит следующее, кроме нулевого:



где:

r это номер текущего раунда, начиная с 1

S это фрагмент расширенного ключа

<<<n операция циклического сдвига на n битов влево

В нулевом раунде выполняется операции наложения двух первых фрагментов расширенного ключа на шифруемые данные

Алгоритм дешифрования происходит в обратном порядке шифрования:



>>> n - операция циклического сдвига вправо

r - номер раунда в обратном порядке, т.е. начиная с R и заканчивая единицей.

В конце выполняется обратные операции и для нулевого раунда.

Альтернативные варианты, это алгоритмы шифрования HS256, RS256 и ES256.

**3. Обращение к программе**

Приводятся наименование и полное описание методов, алгоритмов.

Методы класса Ai:

1. \_init\_ инициализация класса
2. get\_oponent\_drop получить точку куда сходим противник
3. Ai\_drop выбор клетки для хода, на основе очков клетки
4. get\_point\_score подсчет общего числа очков переданной клетки
5. get\_direction\_score получить счет выбранного направления
6. get\_stone\_color получение цвета камушка

Методы класса Checkerboard:

1. get\_checkerboard вернуть игровую доску
2. can\_drop проверка доступности хода на клетку
3. drop метод ставящий фишку на клетку
4. win возвращает True если победитель обнаружен
5. get\_count\_on\_direction посчитать количество фишек на направлении

Методы класса RC5:

1. \_init\_ инициализация класса, установка размера блока, количество раундов и ключевое слово
2. lshift побитовый сдвиг влево

Функции:

1. take\_settings получить настройки из файла
2. save\_settings сохранить настройки в файл
3. start\_game начать игру
4. set\_color установить цвет фишек
5. get\_color получить текущий цвет фишек
6. finish\_game\_windows вывод экрана окончания игры
7. draw\_checkerboard отрисовка игровой доски
8. draw\_chessman отрисовка шашек

Библиотеки:

1. Pygame, библиотека для вывода графики. В приложении используется для создания графической части. Внутри приложения используются следующие методы: display, draw, gfxdraw.
2. Pygame-menu, библиотека для легкого создания GUI в приложениях написанных с помощью библиотеки Pygame. В приложении используется класс menu и его методы: add(кнопки, текст, поле для ввода текста и селектор)

**4. Сообщения**

При вводе имени больше 256 символов, выводиться всплывающее окно с ошибкой. Имя может быть меньше 256 символов.