**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»

**Техническое задание**

Листов 4

**Руководитель разработки**:

профессор каф. ИВК, к.т.н., доцент

*Шишкин Вадим Викторинович*

« » 2022 г.

**Исполнитель**:

студент гр. ИСТбд-22

*Бартецкий Павел Витальевич*

« » 2022

**2022**

**Введение**

Правила игры в Го-бан схожи с правилами в крестики нолики. Вся игра происходит на клетчатой доске от 10х10 до 19х19

Каждый из двух игроков размещает фишки на поле, главная цель это собрать пять фишек в ряд. Собрать пять в ряд, можно как по горизонтали и вертикале так и по диагонали.

Для реализации доски, необходимо взять двумерный список. Двумерный список идеально повторяет доску для игры, для имитации пустых клеток используется 0, для фишек первого игрока 1 а для фишек второго игрока 2.

Для загрузки настроек игры, статистики и сохранений игры используется файлы в формате json. Для распаковки данных файлов в программном коде использовать словарь ключ-значение.

**1. Основания для разработки**

В качестве оснований для разработки указываются учебный план направления 09.03.02 «Алгоритмы и структуры данных» и методические указания к выполнению курсовой работы.

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

**2.1 Общая постановка задачи**

Определяется общей постановкой задачи в задании на курсовую работу

**2.2 Требования к функциональным характеристикам**

2.2.1 Требования к структуре приложения

* RC5 класс реализующий шифрование и дешифрование файлов
* AI класс отвечающий за ходы компьютера
* Checkerboard класс отвечающий за проверку состояния доски

Дополнительные библиотеки: pygame, pygame\_menu, collections и os

* pygame библиотека для реализации самой игры
* pygame\_menu расширение для pygame, для реализации игрового меню
* collections библиотека с коллекциями пайтон
* os библиотека для взаимодействия с ОС

2.2.2 Требования к организации входных и выходных данных

Имя пользователя, минимальная длина 1 символ, максимальная 256.

Сохранённые настройки хранятся в файле в формате json

Входные и выходные данные для функций:

* take\_setting: входные данные name: str; выходные None
* get\_direction\_score: входные данные point: Point, x\_offset: int, y:offset: int выходные int.
* Функции выводящие меню и саму игру: входные данные screen: pygame.display

**2.3 Требования к надежности**

Для хранения данных пользователей используются файлы в формате json.

**2.4 Требования к информационной и программной совместимости**

Для работы приложения необходима версия Windows 10/11, Python версии 3.11 и основная библиотека pygame 2.1.3.

Для разработки использовалась следующая IDE: Pycharm community 2022.3

**2.5 Требования к маркировке и упаковке**

Определяются заданием на курсовую работу

**2.6 Требования к транспортированию и хранению**

2.6.1 Условия транспортирования

Требования к условиям транспортирования не предъявляются

2.6 2 Условия хранения

Идеальные условия хранения:

* Температура от +10 до +20 градусов по цельсию
* Относительная влажность от 20 до 65 процентов
* Расстояние от источников тепла, влаги не менее 0,5 метра

Допустимые условия хранения:

* Температура от -55 до +55 градусов по цельсию
* Относительная влажность от 5 до 90 процентов
* Расстояние от источников тепла, влаги не менее 0,5 метра

2.6 3 Сроки хранения

Срок хранения – до июля 2023 года

**3. Требования к программной документации**

Определяются заданием на курсовую работу

**4. Стадии и этапы разработки**

Ограничиться фразой «Определяются заданием на курсовую работу».

**5. Порядок контроля и приёмки**

Ограничиться фразой «Определяются заданием на курсовую работу».