**Приложение 1**

**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 курсовую работу**

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

Тема: Приложение «Го-Бан»

**Инв. № подл.**

**Подп. и дата**

**Взам. инв. №**

**Инв. № дубл.**

**Подп. и дата**

Р.02069337. 21/298-06 ТЗ-01

Листов 3

**Исполнитель**:

студент гр. ИСТбд-22

*Бартецкий Павел Витальевич*

« » 2022 г.

**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 вертикали так и по диагонали.

Игровая доска реализуется с помощью двумерного списка. 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 распоряжение по факультету.

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

**2.1 Функциональное назначение**

Научиться играть в игру Го-Бан. Правила игры указаны выше.

**2.2 Требования к функциональным характеристикам**

2.2.1 Требования к структуре приложения

Используемые классы:

1. Шифрование и дешифрование
2. Ходы компьютера
3. Проверка состояния доски

Используемые библиотеки:

1. Библиотека для разработки игр на языке Python
2. Библиотека для создания GUI в играх на языке Python
3. Дополнительные коллекции языка Python

Используемые виджеты:

1. Кнопки(Button)
2. Списки с выбором(Selector)
3. Текст(Text)
4. Поля для ввода данных(Entry)

Используемые структуры данных:

1. Списки
2. Словари
3. Кортежи

2.2.2 Требования к составу функций приложения

Главные функции приложения: начать игру.

В процессе самой игры: выбор хода, ход бота с помощью оценочного алгоритма.

2.2.3 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:

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

**2.3 Требования к надежности**

Работоспособность стабильна

**2.4 Требования к информационной и программной совместимости**

Средство разработки: Pycharm community 2022.3

Операционная система: Windows 10/11

Язык программирования Python 3.11

**2.5 Требования к маркировке и2 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

Допустимые условия хранения:

* Температура от -55 до +55 градусов по цельсию

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**

Определяются заданием на курсовую работу.

**5. Порядок контроля и приёмки**

Определяются заданием на курсовую работу.