МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«УЛЬЯНОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра «Измерительно-вычислительные комплексы»

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | | |  | |  |  |
| |  |  | | --- | --- | | ***Подп. и дата*** |  | | ***Инв. № дубл.*** |  | | ***Взам. инв. №*** |  | | ***Подп. и дата*** |  | | ***Инв. № подл*** |  | |  | | **Пояснительная записка**  на лабораторную работу №8  по дисциплине «Алгоритмы и структуры данных»  **Тема** «Разработка компьютерной игры PVP Arena» | | | | | | |
|  | | |  | |  | | | |
|  | | Исполнитель  студент гр. ИСТбд-21  Казаров Д.С.  «\_\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 2024 г. | | | |
|  | | 2024 | | | | | | |

**Введение**

Разрабатываемое приложение представляет собой пошаговую тактическую игру "PvP Арена", разработанную с использованием языка программирования Python и библиотеки Pygame для графического интерфейса. Целью приложения является моделирование пошаговых сражений между командами персонажей на игровом поле. Программа демонстрирует навыки программирования, создания пользовательских интерфейсов и реализации игровой логики.

**Основания для разработки:**

Проект разработан в рамках учебного задания по курсу "Разработка игр на Python".

**Постановка задачи**

Цель разработки заключается в создании пошаговой тактической игры, в которой игрок управляет командой персонажей, сражающихся против команды компьютера на игровом поле, разделенном на клетки.

**Структура программы**

**Классы:**

1. Character: класс для создания и управления персонажами.

**Основные функции:**

1. create\_team - создание команды персонажей.

2. check\_winner - проверка условий победы.

3. ai\_move - логика хода компьютера.

4. draw\_character\_info - отображение информации о персонаже.

5. draw\_grid - отрисовка игрового поля.

6. main\_menu - отображение главного меню.

7. main\_game - основной игровой цикл.

**Функциональное назначение:**

1. Создание и управление персонажами.

2. Реализация пошаговой механики боя.

3. Отображение игрового поля и информации о персонажах.

4. Реализация искусственного интеллекта для команды противника.

5. Проверка условий победы и завершения игры.

**Проектная часть**

**1.1 Математические методы**

Приложение использует систему координат для управления перемещением персонажей по игровому полю. Для расчета дистанций атаки и перемещения применяются простые геометрические методы, такие как расчет евклидова расстояния между клетками.

**1.2 Архитектура и алгоритмы**

**1.2.1 Архитектура**

Архитектура приложения включает несколько основных компонентов:

• Игровое поле – область экрана, разделенная на клетки, где происходит сражение.

• Персонажи – объекты, представляющие игровых юнитов с различными характеристиками.

• Команды – группы персонажей, управляемые игроком и компьютером.

• Панель информации – область экрана для отображения данных о выбранном персонаже.

• Система ходов – механизм, управляющий очередностью действий игрока и компьютера.

**1.2.2 Алгоритмы**

• Алгоритм перемещения персонажей: Персонажи перемещаются по клеткам в пределах своего радиуса движения, избегая столкновений с другими персонажами.

• Алгоритм атаки: Персонажи могут атаковать противников в пределах своего радиуса атаки, нанося урон по здоровью цели.

• Алгоритм ИИ: Компьютер выбирает действия для своих персонажей, основываясь на их положении относительно противников и своих характеристиках.

**Стадии и этапы разработки**

1. Определение требований: постановка задач и определение функциональности игры.

2. Проектирование: разработка структуры классов и функций, планирование игровой механики.

3. Реализация: написание кода программы, создание графического интерфейса.

4. Тестирование: проверка работоспособности всех игровых механик и взаимодействий.

5. Оптимизация: улучшение производительности и баланса игры.

6. Документирование: создание пояснительной записки и комментариев к коду.

**Тестирование**

**Цель тестирования**

Цель тестирования заключается в проверке корректности работы всех игровых механик, взаимодействия персонажей и баланса игры.

**Методика тестирования**

Тестирование проводилось методом "черного ящика", где проверялись функциональные возможности игры при различных сценариях.

**Тестирование включало:**

1. Модульное тестирование: проверка работы отдельных классов и функций.

2. Интеграционное тестирование: проверка взаимодействия между различными компонентами игры.

3. Функциональное тестирование: проверка соответствия игры заданным требованиям.

4. Тестирование пользовательского интерфейса: проверка удобства использования и отзывчивости интерфейса.

5. Тестирование баланса: проверка справедливости игрового процесса при различных стратегиях.

**Результаты тестирования**

Были выявлены и исправлены ошибки в логике перемещения персонажей, механике атаки и работе ИИ противника.

**Заключение**

Разработанная игра "PvP Арена" полностью соответствует поставленным задачам и обеспечивает увлекательный игровой процесс в жанре пошаговой тактики. Игра имеет интуитивно понятный пользовательский интерфейс и предоставляет возможность для развития стратегического мышления игрока.

**Источники, использованные при разработке**

1. Свейгарт, Эл. Разработка игр на Python. – М.: Эксмо, 2021.

2. "Документация Pygame" [Электронный ресурс] // Официальный сайт библиотеки Pygame: https://www.pygame.org/docs/ – Режим доступа: свободный.

3. Макконнелл, С. Совершенный код. Мастер-класс. – СПб.: БХВ-Петербург, 2017.

4. Рассел, С., Норвиг, П. Искусственный интеллект: современный подход. – М.: Вильямс, 2020.