Dark Maze

1. Основная идея.

Основная идея проекта – создание хоррор-игры с использованием языка программирования Python и библиотеки PyGame.

1. Механика игры.

С самого начала игроку предлагается зайти в аккаунт, т.к. для каждого аккаунта сохраняется лучшее время прохождения. Далее игрок попадает в главное меню, где он может выбрать один из двух уровней и посмотреть правила игры. В каждом из уровней ему надо собрать два ключа и вернуться в начальное положение, не наткнувшись на двух врагов, которые постоянно бегут за ним, находя самый короткий путь. Также на обоих уровнях есть монетки, за которые после прохождения можно купить улучшения: увеличение скорости или дополнительную жизнь. Изначально у игрока есть 3 жизни. Если игрок собрал оба ключа, количество жизней автоматически уменьшается до одной.

Если игрок потратит все свои жизни, на экране появится надпись “Вы проиграли”. Если же он успешно пройдёт уровень, на экране будет написано “Вы прошли уровень”. Также будет написано время прохождения и лучшее время прохождения выбранного уровня, данные о котором хранятся в базе данных. После этого можно будет поставить оценку игре. Данные о поставленной оценке записываются в текстовый файл, после чего вычисляется общая оценка путём вычисления среднего арифметического из всех поставленных.

1. Основные классы и функции.

Техническая часть

Код состоит из пяти классов (Player, Drawing, Sprites, SpriteObject и Interaction), константных переменных и функции main, в которой запускается программа.

1. Класс Player.

Он состоит из пяти функций, в которых прописаны логика игрока (движение и проверка на столкновения со стенами). Функция pos возвращает координаты игрока.

2. Класс Drawing.

В инициализации – загрузка картинок стен, в функции ray\_casting – отображение стен, в функции time – отображение времени прохождения, в функции life – отображение количества жизней, в функции mini\_map – отображение уменьшенной карты лабиринта и монет.

3. Класс Sprites.

В инициализации происходит загрузка картинок монет, ключей и врагов с их местоположением.

4. Класс SpriteObject.

В нём происходит проверка попадания спрайтов в поле видимости игрока. Если спрайт ближе к игроку, чем стена, то функция object\_locate возвращает координаты спрайта, если не попадает, то возвращает False.

5. Класс Interaction.

В нём прописана логика врага (нахождение самого короткого пути до игрока и его поимка)

6. Функция main.

В ней происходит загрузка всех необходимых файлов, запуск главного цикла и музыки. main запускает 14 окон, которые включаются в зависимости от действий пользователя:

Первое – стартовое окно

Второе – правила игры

Третье – ввод имени

Десятое – меню

Четвёртое и седьмое – первый и второй уровни

Пятый и восьмой – запись результата прохождения в базу данных и вывода его и самого лучшего на экран.

Девятый - конечное окно

Двадцать второй – магазин

Двенадцатый и тринадцатый – видео поимки игрока врагом и окно проигрыша

Шестой и одиннадцатый – отображения номеров уровней

1. Библиотеки.

Главная библиотека – PyGame. Также была задействована библиотека moviepy.editor для воспроизведения видео скримера. Sqlite3 используется для сохранения данных о времени прохождения. Встроенный модуль math используется для получения 3d изображения.

1. Скриншоты.

Стартовое окно

Изображение выглядит как текст, снимок экрана, искусство, дизайн

Автоматически созданное описание

Процесс игры

Изображение выглядит как снимок экрана, мультфильм, текст, Графика

Автоматически созданное описание