**Создание игры – платформера на базе pygame**

**Hero of middleearth**

Подготовили: Дубич Евгений,

Мамонтов Олег,

Спеваков Владимир

**Яндекс Лицей 2021**

**Введение**

В настоящее время распространена сфера написания игр. Эта сфера стремительно развивается и требует соответствующего программного обеспечения и других дополнительных модулей. Обучаясь в Яндекс Лицее необходимо было изучить программное обеспечение на языке программирования, связанное с библиотекой pygame. И в качестве контроля знаний, была введена интересная форма теста – проект-игра.

**Цель:** Создание игры в среде программирования Python с использованием библиотеки pygame.

**Задачи**:

1. Создать план игры
2. Создание основных методов героя
3. Создания дополнительных методов (ии для определения и управления персонажем, реализация полета пули)
4. Создания графики, меню и медиа.
5. Сборка игры

**Создание плана игры**

В среде программирования Python доступно взаимодействие с пользователем (Считывание нажатий клавиш и кликов мыши).

Изначально было решено производить загрузку карты с помощью списка, в котором хранилась бы информация о необходимом блоке. Далее производился проход по нему и добавление изображений каждого блока.

**Создание основных методов героя**

Главный герой (Player) – изначально должен уметь передвигаться и иметь простое взаимодействие объектов. Был написан класс героя. Персонаж представлял собой прямоугольник, который получал координаты нахождения на карте и мог передвигаться за счет нажатия на необходимые клавиши. Далее были написаны классы стен. Объект (Wall) представляет собой квадрат – координаты которого берутся из карты. Далее они были добавлены в группу спрайтов, чтобы обеспечить коллизию и предотвратить прохождение сквозь стены.

Следующий этап – создание класса камеры, которая обеспечивала отображение карты относительно координат героя.

**Создания дополнительных методов (ии для определения и управления персонажем, реализация полета пули)**

Для создания карты был найден более удобный способ для создания и отрисовки карты (Tiled Map). С помощью редактора можно быстро и качественно создавать карты, заранее создавая карту блоков (тайлов).

Создание искусственного интеллекта для персонажа производилось при помощи написания класса (Mobs). Координаты появления считывались с карты. Далее находились пользовательские координаты, чтобы определить направление движения и построить направленный вектор. Для обеспечения четкого перемещения героя с помощью векторной математики и написания функции с формулой для вычисления угла, объект мог передвигаться в определенном направлении, зависимо от координат главного героя.

Позже для добавления функций игры был создан класс (Bullet). При нажатии правой клавиши мыши, считывались координаты нажатия, и строился прямоугольный треугольник, гипотенуза которого представляла прямую соединяющая пользовательские координаты и точку нажатия, которая представляла собой сумму координаты нажатия и положения камеры. Далее вычислялся угол полета с помощью тригонометрических функций (arcsin, sin). Далее строится прямая, соединяющая координаты объекта и координат нажатия. Далее была написана функция для вычисления координат данной прямой. Движение объекта класса (Bullet) представляет собой проход по списку координат и перемещение, шаг которого зависит параметров (скорость и дальность полета).

**Создания графики, меню и медиа.**

Для прорисовки анимаций было решено использовать пиксельную 2D графику. Это обеспечивает простоту отрисовки и простоту передвижения, и низкие требования от пользовательского компьютера. В приложении для создания изображений (Pixel Art) были созданы все персонажи и анимации передвижения.

Также для своевременной отрисовки анимаций персонажей были написаны функции, которые вызывались в зависимости от угла передвижения. В функции производилось прохождение по списку анимаций и присвоение их изображений персонажу.

Для удобного взаимодействия с пользователем было создано меню, с возможностью выбора (начало игры, загрузка игры или выход). Кнопки представляют собой объекты, которые считывают положение мыши, при нажатии на определенную кнопку производится вызов определенной функции.

Также была добавлена музыка, написанная на гитаре. Музыкальные эффекты позволяют понять пользователю о состоянии игры в настоящее время (нахождение в меню, прохождение уровня и проигрыш).

**Сборка игры**

Далее были соединены основной код игры и меню. Добавлены анимации и звуковые эффекты.

В итоге была получена готовая игра, удобная для пользователя, написанная в языке Python с помощью библиотеки pygame, имеющая звуковые и анимационные эффекты.