

Московский авиационный институт
(национальный исследовательский университет)
Факультет информационных технологий и прикладной математики
Кафедра вычислительной математики и программирования

Лабораторная работа №6 по курсу «Компьютерная графика»

Студент: Е.Ю. Юрков
Группа: М80-312Б-22
Дата:
Оценка:
Подпись:

Москва, 2024

Цель лабораторной работы

Цель лабораторной работы: В этой лабораторной работе вы будете разрабатывать основу для собственного игрового движка. Это завершающая работа, в которой вы объедините все изученные техники работы с 2D и 3D графикой, освещением, шейдерами и трассировкой лучей. В результате выполнения лабораторной работы у вас должен получиться простой движок, который может отрисовывать сцены в реальном времени, с возможностью работы с камерой, объектами и освещением. Основные задачи:

1. Создать базовую архитектуру игрового движка с поддержкой 3D-сцен.
2. Реализовать систему рендеринга с использованием изученных подходов: рендеринг через растризацию и трассировку лучей.
3. Настроить камеру и систему управления ею.
4. Реализовать поддержку базовых игровых объектов и взаимодействие с ними.
5. Обеспечить работу с несколькими типами источников света.
6. Оптимизировать производительность движка (по возможности).

Метод решения

Движок я решил реализовать в качестве библиотеки. Она имеет следующую структуру:

- `camera` - директория, содержащая классы для работы с камерой, а именно для отправления матрицы камеры в шейдер.
 - `Camera.hpp` - основной класс для работы с камерой. Позволяет задавать позицию камеры и позицию цели камеры. Также в этом файле содержится структура, позволяющая задавать настройки камеры, например, проекцию, плоскости отсечения, угол обзора.
 - `FPCamera.hpp` - класс для камеры от первого лица, наследует класс `Camera`, а также имеет методы для обработки мыши и привязки к объекту.
 - `TPCamera.hpp` - класс для камеры от третьего лица, наследует класс `Camera`, а также имеет методы для обработки мыши и привязки к объекту.
- `objects` - директория, содержащая классы основных объектов и класс модели (`.obj`).
 - `ObjectBase.hpp` - базовый класс для всех фигур, задаёт интерфейс взаимодействия с объектами, а также определяет операции для перемещения объектов через матрицу модели.
 - `Cube.hpp` - класс куба, позволяет также накладывать текстуру на куб.
 - `Flat.hpp` - класс плоскости, с текстурой.

- Model.hpp - класс для обработки фигуры из файла .obj, берёт текстуру для модели из файлов указанных в файле .mtl.
- shader - директория, содержащая класс Shader для удобной компиляции шейдеров и работы с ними.
- texture - директория, содержащая класс Texture для установки текстур и работы с ними.
- light - директория, содержащая классы источников света.
 - PointLight.hpp - класс точечного источника света, позволяет установить цвет освещения. В шейдере учитывается расстояние от источника для затухания.
 - DirectLight.hpp - класс направленного света.
- logic - директория, содержащая классы событий, которые обрабатывает движок.
 - Event.hpp - базовая структура события, содержит всего два поля: condition и action - условие, по которому возникает событие и действие, которое должно быть совершено.
 - FrameEvent.hpp - событие, срабатывающее на каждом кадре.
 - EveryNSec.hpp - событие, срабатывающее каждые N секунд.
 - KeyboardEvent.hpp - событие, срабатывающее при нажатии определённой кнопки.
 - MouseEvent.hpp - событие, срабатывающее при движении мыши, action для этого события должен принимать offsetX, offsetY и scroll.
 - IntersectEvent.hpp - событие, проверяющее пересечение bounding box объектов.
- engine - директория, содержащая основной класс движка - сцену.
 - Scene.hpp - класс сцены. Позволяет привязывать к сцене объекты, источники света, камеру и назначать события.

В итоге на разработанном движке была создана игра, где снеговик, перемещаясь по ограниченному полю, должен был ловить снежинки и вырасти благодаря им до определённого размера.

Результат работы программы



Выводы

В ходе выполнения этой лабораторной работы я укрепил знания, полученные в курсе компьютерной графики. Я изучил устройство и структуру obj файлов, а также написал класс для загрузки объектов из файла. Также эта лабораторная позволила углубить знания в области разработки игр.