

Редактор трехмерных моделей

Студент:

Калашников С. Д.
ИУ7-53Б

Научный руководитель:

Павельев А. А.

Москва 2022

Цель и задачи работы

Цель данной работы – разработать программу которая позволит редактировать трехмерные модели на уровне вершин, ребер, полигонов.

1. Описать используемые структуры.
2. Описать алгоритмы визуализации трехмерной сцены.
3. Реализовать данные алгоритмы и структуры.
4. Провести исследования на основе написанной программы.

Описание сцены

- Камера
- Модель
- Источник света

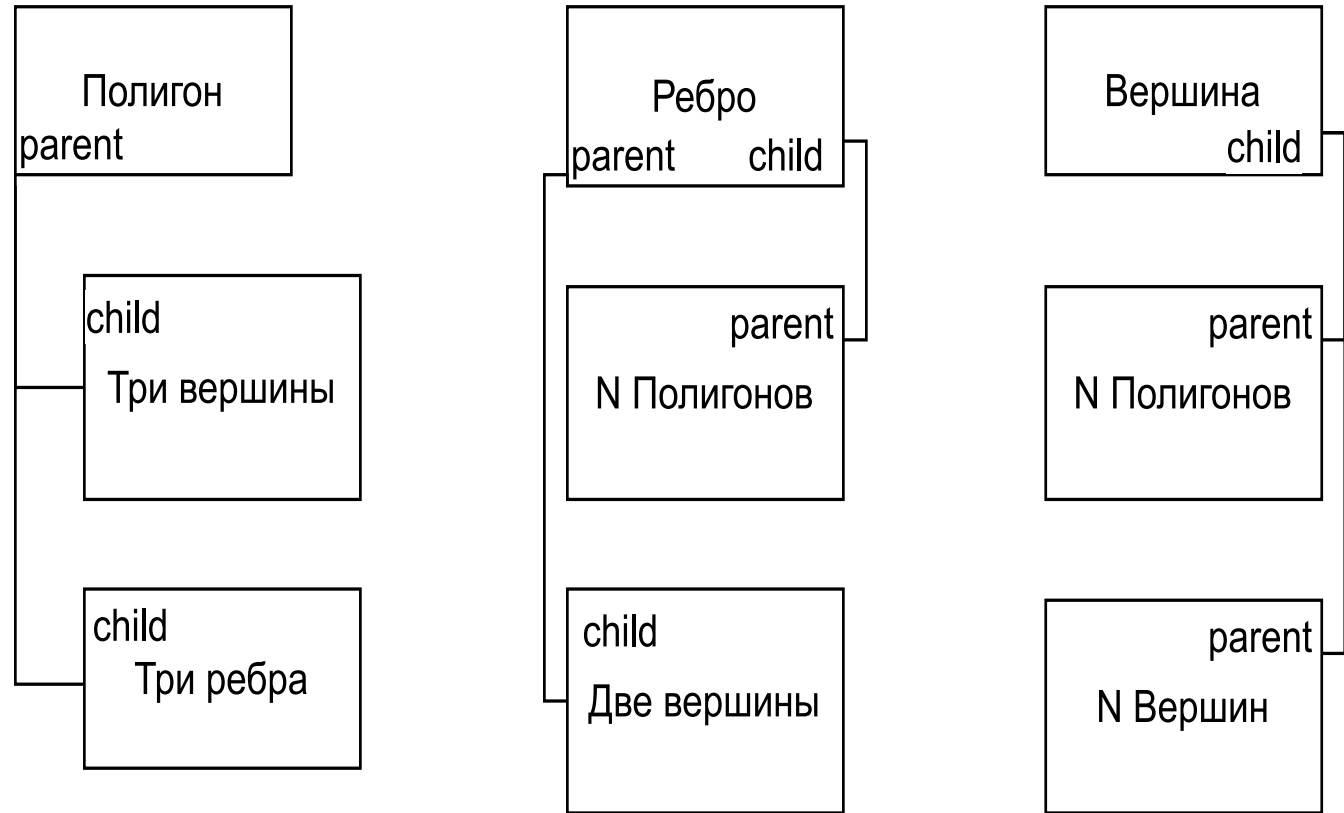
Описание модели

Полигональная модель состоит из:

- трех контейнеров с элементами;
- контейнера с активными элементами.

Особенности:

- отсутствие повторяющихся элементов;
- удобство работы с каждым элементом по отдельности и моделью целиком.



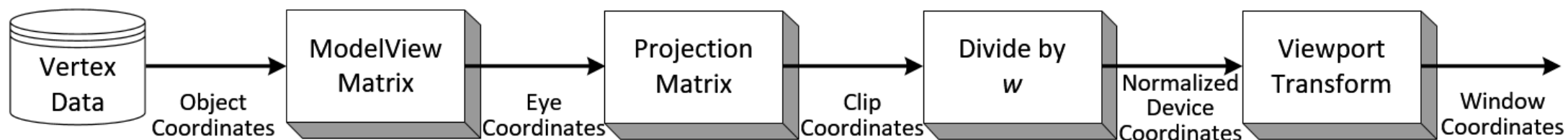
Отношения между элементами

Алгоритмы визуализации

	Плюсы	Минусы
Растеризация	<ul style="list-style-type: none">• Скорость работы;• большой выбор алгоритмов при реализации.	<ul style="list-style-type: none">• Качество изображения;• объем занимаемой памяти.
Рейкастинг	<ul style="list-style-type: none">• Реалистичность изображения;• занимает меньший объем памяти.	<ul style="list-style-type: none">• Время работы.

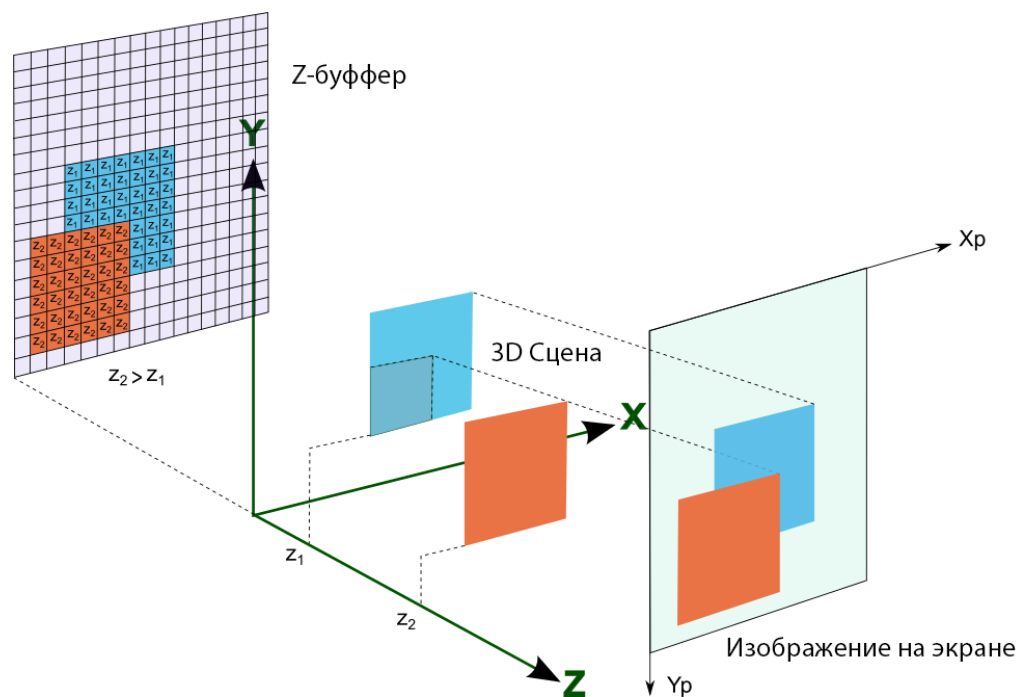
Растеризатор

1)

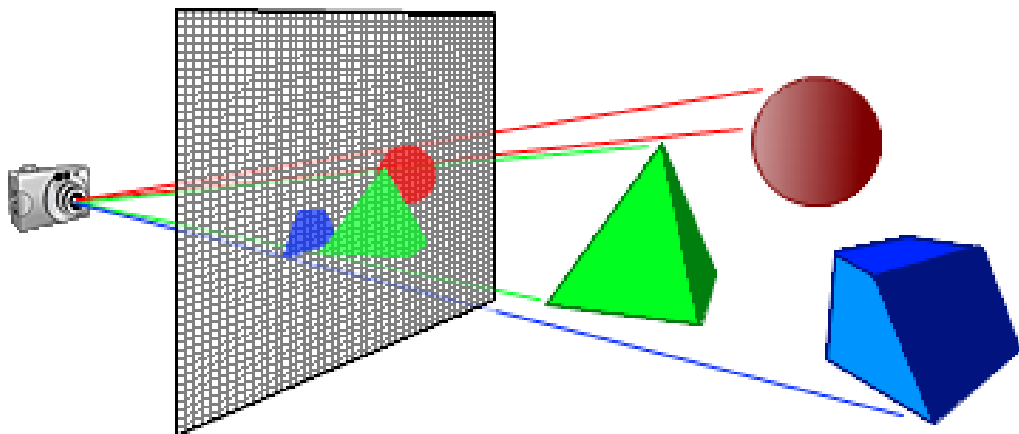


2) Интерполяция или барицентрические координаты

3)



Рэйкастинг



Процесс работы рейкастинга

Уравнение луча:

$$\text{PointOnRay} = \text{Raystart} + t * \text{Raydirection}$$

Представление плоскости:

$$X_n \cdot X = d$$

Нахождение точки пересечения:

$$t = (\text{Xn} \cdot (\text{PointOnRay} - \text{Raystart})) / (\text{Xn} \cdot \text{Raydirection})$$

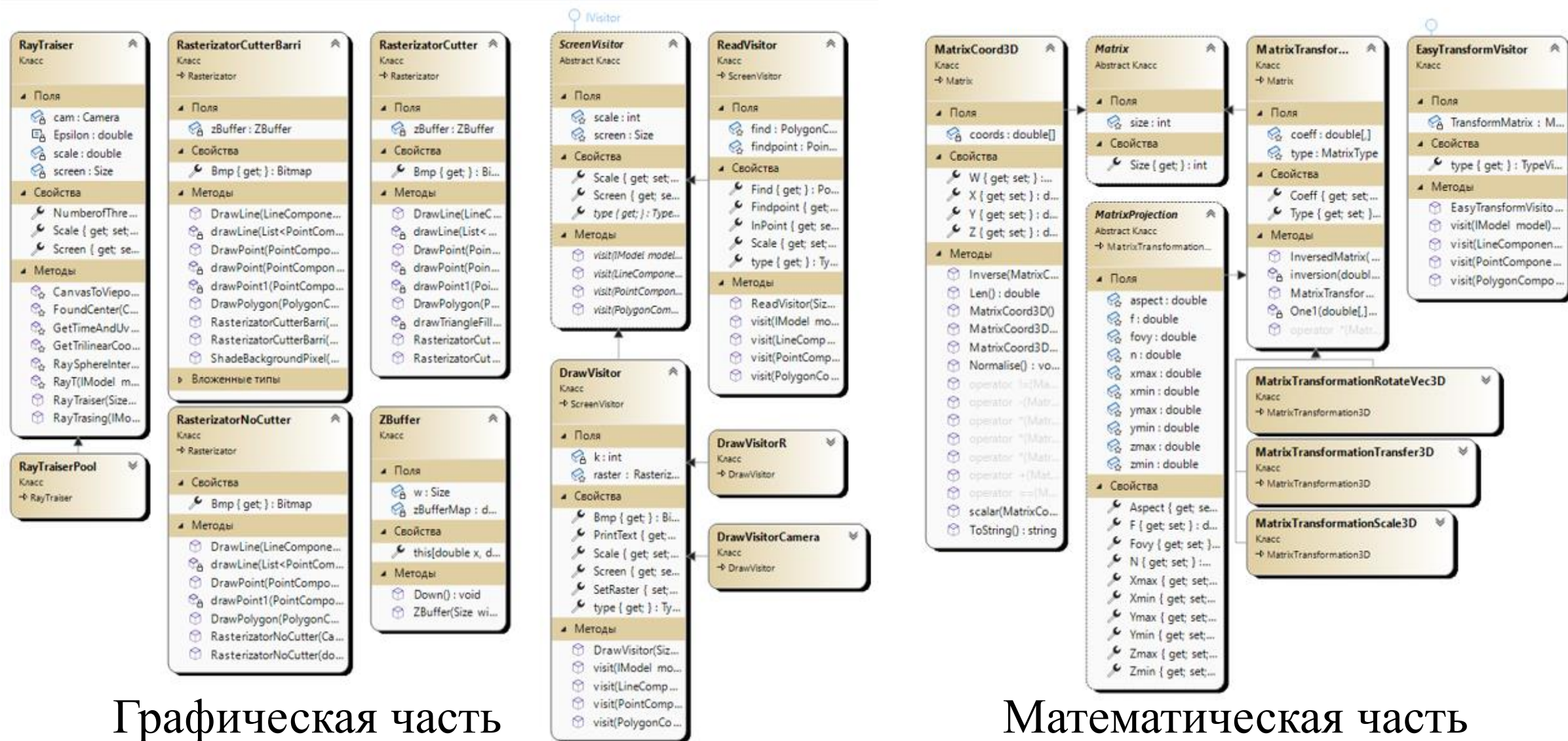
Выбор языка программирования и среды разработки

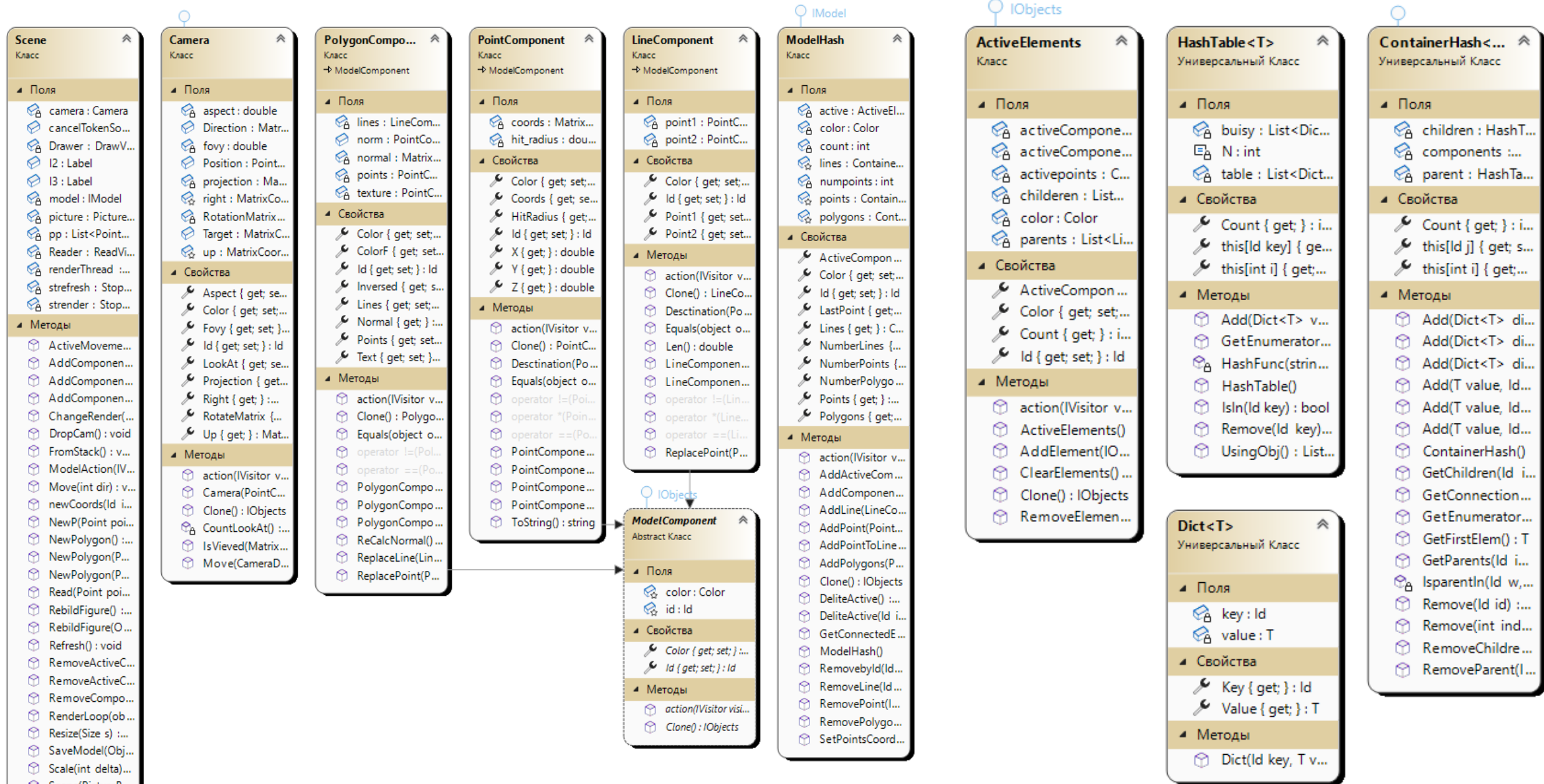
В качестве языка программирования был выбран C#:

- реализована многопоточность;
- объектно-ориентированный язык;
- высокая эффективность и большой набор стандартных классов и процедур.

В качестве среды разработки была выбрана Visual Studio 2022

Структура и состав классов





Объектная часть

Контейнерная часть

Интерфейс

Интерфейс позволяет:

- добавлять новые точки и ребра правой кнопкой мыши;
- удалять элементы;
- трансформировать элементы как вместе так и по отдельности;
- создавать новые полигоны;
- изменять позицию существующих точек;
- просматривать модель с любых ракурсов.

FormTransform

Перенос Масштаб Поворот

0	▲▼	Перенос по X
0	▲▼	Перенос по Y
0	▲▼	Перенос по Z

Подтвердить

☐ Операция относительно точки

0	▲▼	X
0	▲▼	Y
0	▲▼	Z

ActiveElementsForm

Polygon 148	Remove from Model	Delit fror
Point 5	Remove from Model	Delit fror

Форма активных элементов

Форма трансформирования

Главная форма

Form1

Файл Метод рендера Выбор элементов

Добавить полигон Изменить позицию точки

Добавить **ок** / Добавить существующую

Добавить **ок** / Добавить существующую

Добавить **ок** / Добавить существующую

Добавить полигон

Возврат камеры

Возврат фигуры

Выбрать трансформирование

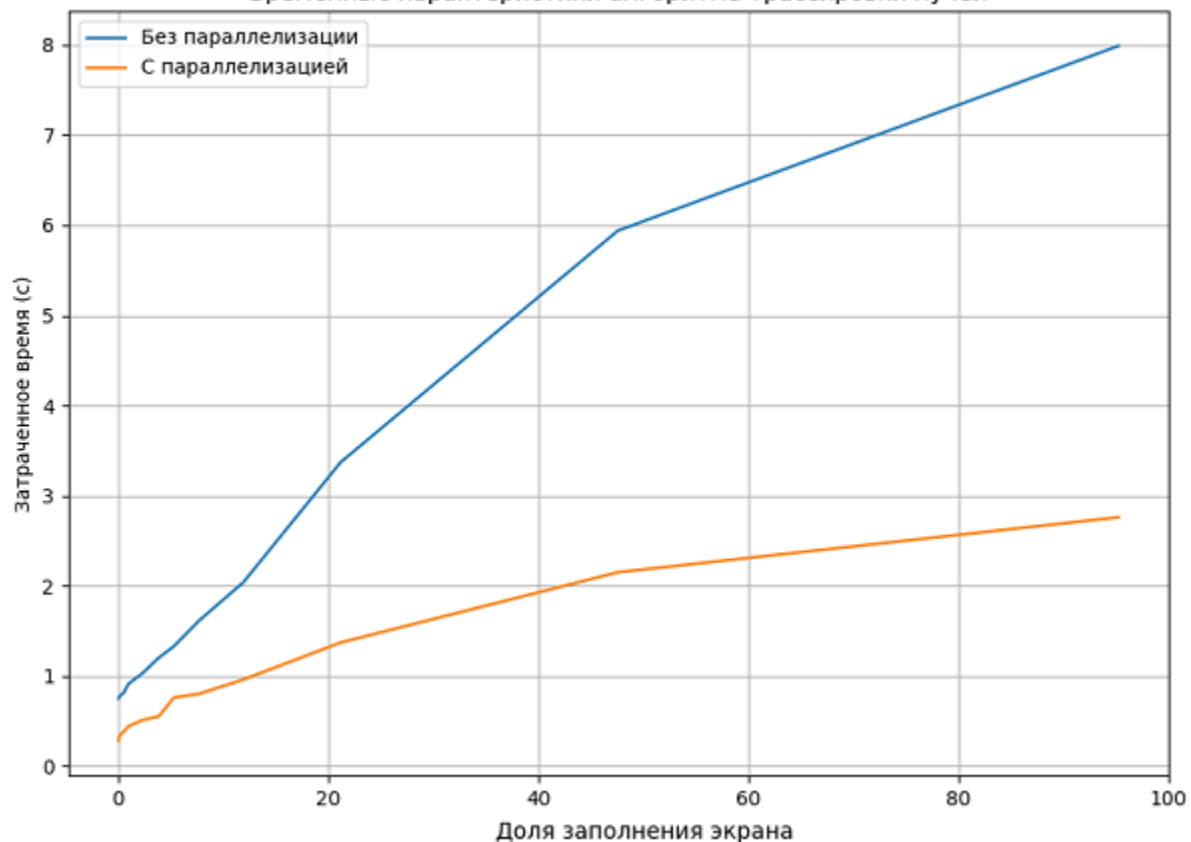
▼ Сделать активным

Время отрисовки предыдущей сцены: 00:00:00.00 Точка попадания: 0 0 0

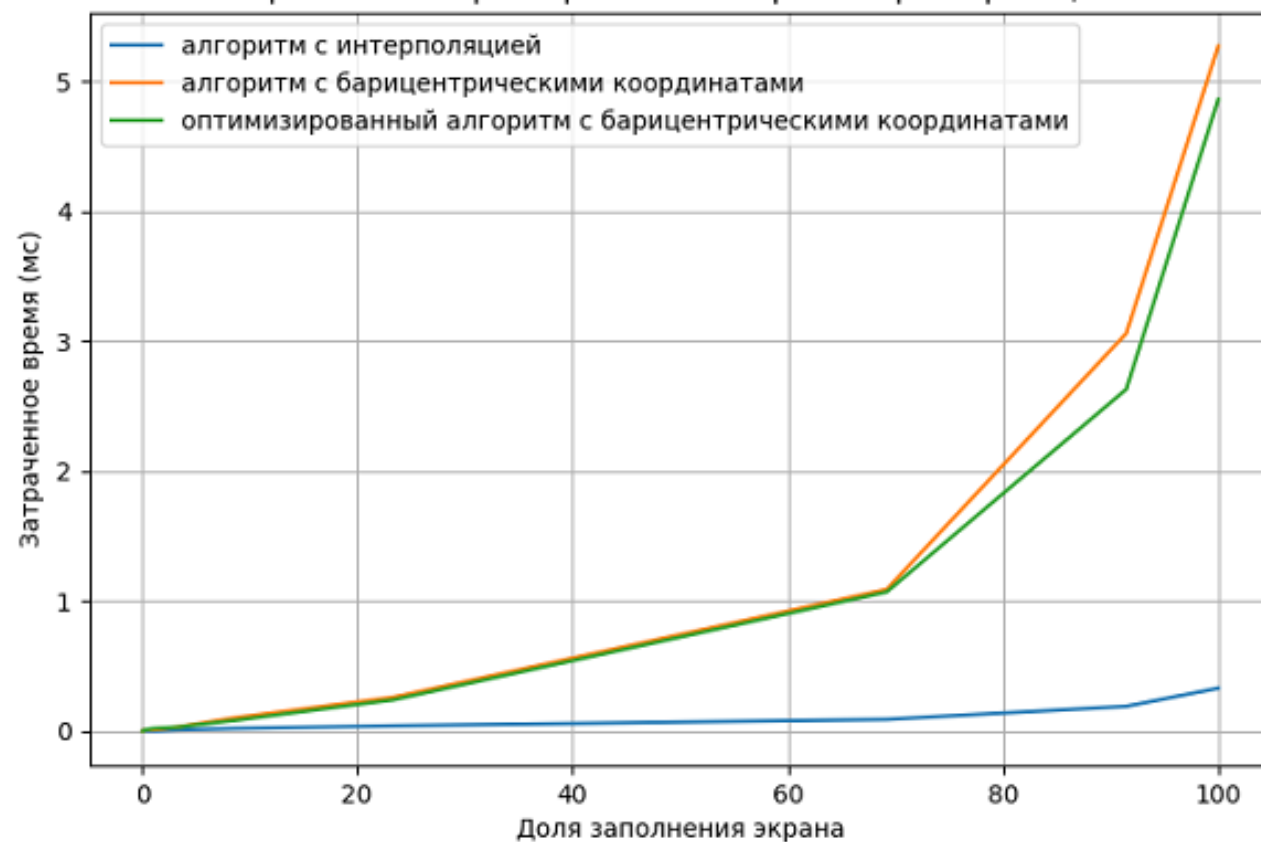
Остановить отрисовку Возобновить отрисовку

Исследования

Временные характеристики алгоритма трассировки лучей



Временные характеристики алгоритмов растеризации



Итоги

Выполнены следующие задачи:

1. Описаны используемые структуры.
2. Описаны алгоритмы визуализации трехмерной сцены.
3. Реализованы данные алгоритмы и структуры.
4. Проведено исследования на основе написанной программы.