**1 слайд (титульный): ~15 секунд**

Здравствуйте, уважаемые члены комиссии, темой курсового проекта, в соответствии с техническим заданием, является разработка редактора композиций трехмерных графических примитивов.

**2 слайд (цели и задачи): ~30 секунд**

В ходе выполнения курсового проекта были выделены следующие задачи:

* Изучение и анализ существующих алгоритмов компьютерной графики, использующихся для создания реалистичной модели взаимно перекрывающихся объектов, и выбор наиболее подходящих для решения поставленной задачи.
* Проектирование архитектуры программы и ее интерфейса, с учетом возможностей ее дальнейшей модификации.
* Реализация выбранных алгоритмов и структур данных.
* Проведение исследования временных характеристик разработанной программы.

**3 слайд (сравнение алгоритмов удаления невидимых линий): ~ 1 минута**

При построении реалистичного изображения основной задачей является задача удаления объектов или их частей, которые перекрываются другими объектами, то есть являются невидимыми с точки зрения наблюдателя.

В ходе работы над курсовым проектом был проведен сравнительный анализ основных алгоритмов, решающих эту задачу. В качестве критериев были использованы зависимость трудоемкости алгоритма от числа объектов, использование рекурсивных вызовов, простота реализации и распространённость в современном ПО. В результате данного сравнения для реализации был выбран алгоритм Z-буфера, подходящий по всем приведенным критериям. Его идея заключается в использовании двух буферов: буфера кадра для хранения интенсивностей пикселей и Z-буфера, для хранения глубин пикселей. При растеризации полигона вычисляется глубина каждого пикселя, в случае если она меньше значения в Z-буфере, то интенсивность пикселя вычисляется и заносится в буфер кадра, а значение в Z-буфере корректируется.

**4 слайд (общий алгоритм программы): ~ 1 минута**

Сцена в данном курсовом проекте состоит из набора объектов, камеры и источника освещения. Ее визуализация происходит по следующему алгоритму:

Изначально координаты объекта находятся в пространстве модели. Для перехода в мировое пространство они умножаются на матрицу аффинных преобразований, которые представлены поворотом и переносом относительно координатных осей.

Затем осуществляется переход в пространство камеры с помощью матрицы вида. Для корректного пространственного восприятия объектов проводится перспективное проецирование с помощью матрицы проекции, все точки переводятся в однородное пространство отсечения.

Для улучшения быстродействия, в пространстве отсечения происходят отбрасывание всех нелицевых по отношению к камере граней и отсечение по пирамиде видимости.

Если полигон не был отброшен, то происходит его растеризация по алгоритму Z-буфера. При расчете интенсивностей пикселей расчет теней осуществляется с помощью кубических карт теней, а вычисления для освещенных частей выполняются в соответствии с моделью Фонга.

**5 слайд (структура классов):~ 1 минута**

Для реализации данного курсового проекта был использован язык С++, поддерживающий объектно-ориентированную модель разработки, за счет которой достигаются четкая структуризация программы и легкая модифицируемость ее отдельных частей. При разработке были спроектированы следующие классы:

* математические классы векторов и матриц,
* классы описывающие модели объектов,
* классы камеры и источника освещения,
* управляющие классы для хранения объектов сцены, приведения координат объектов к экранным и визуализации сцены,
* классы шейдеров для попиксельной обработки полигонов,
* класс-посредник, реализующий взаимодействие между классами графического интерфейса и управляющими классами,
* классы графического интерфейса.

В участке программы с попиксельными вычислениями глубины и интенсивности выполняется наибольшее количество вычислений, поэтому он был распараллелен с помощью библиотеки thread, что увеличило быстродействие программы.

**6 слайд (примеры работы): ~1 минута**

Интерфейс разработанного программного продукта состоит из виджета сцены и меню управления. Меню управления делится на 3 раздела. Раздел Модель содержит параметры для создания и редактирования объектов. Модель задается радиусами верхнего и нижнего оснований, высотой, количеством боковых граней, спектральными характеристиками и параметрами для перемещения по сцене. В разделе Камера находятся настройки пирамиды видимости, позиции и ориентации камеры в пространстве. Также перемещение камеры может осуществляться с помощью клавиатуры, а вращение – с помощью мыши. С помощью раздела Источник можно изменить цвет и позицию источника освещения в пространстве.

На виджете сцены отображаются все созданные объекты, система координат и источник освещения, изображаемый пирамидой.

Как видно из примеров, сцены с пересекающимися и заслоняющими друг друга объектами визуализируются корректно, тени генерируются четко в соответствии с взаимоположением источника освещения и объектов.

**7 слайд (графики): ~1 минута**

При исследовании временных характеристик разработанной программы использовался компьютер на базе 4-х ядерного процессора Intel Core i5 частотой 2300 MГц с 8 ГБ оперативной памяти типа LPDDR3 частотой 2133 МГц. Для замеров времени была использована библиотека chrono.

Было проведено 2 эксперимента:

В 1-ом изучалась зависимость времени рендеринга сцены от количества объектов на ней при фиксированном числе боковых граней и как видно из графика, эта зависимость линейная для объектов с любым числом боковых граней. Максимальное время рендеринга составило 1339 миллисекунд в случае 1000 9-гранных объектов, минимальное время рендеринга составило 64 миллисекунды в случае 100 3-гранных объектов.

Целью 2-го эксперимента было исследовать зависимость времени отрисовки сцены от количества боковых граней объектов. Для исследования использовались 2 объекта, в 1-ом случае количество боковых граней менялось у обоих объектов, во 2-ом и 3-ем случаях оно было фиксировано у одного из них. На графике наблюдаются пересечения 1-ой и 2-ой кривых и 1-ой и 3-ой. Объясняется это тем, что изначально в 1-ом случае суммарное число граней меньше, чем во 2-ом и 3-ем, однако в 1-ом случае оно растет в 2 раза быстрее, поэтому в точках, где в 1-ом и 2-ом и в 1-ом и 3-ем случаях число граней становится равным наблюдаются точки пересечения кривых. Максимальное время рендеринга составило 99 миллисекунд в случае 2-х 500-гранных объектов, минимальное время рендеринга составило 13 миллисекунд в случае 2-х 50-тигранных объектов.

В результате данных исследований была выявлена линейная зависимость времени рендеринга сцены от количества объектов и от числа их боковых граней.

**8 слайд (заключение): ~40 секунд**

В ходе выполнения курсового проекта:

* Был разработан программный продукт, позволяющий создавать и редактировать композиции из трехмерных графических примитивов.
* Рассмотрены, проанализированы и реализованы основные алгоритмы построения реалистичного трехмерного изображения.
* Спроектирован пользовательский интерфейс, предоставляющий широкие возможности настройки геометрических и спектральных характеристик объектов, положения камеры и положения и цвета источника освещения.
* Проведены и проанализированы экспериментальные исследования временных характеристик разработанного программного продукта.

Программа курсовой работы полностью соответствует поставленному техническому заданию.