|  |
| --- |
| Слайд 2  **Целью** данной работы является разработка программного обеспечения для визуализации шахматных фигур на шахматной доске.  Для достижения поставленной цели требуется решить следующие **задачи**:  1) формализовать задачу;  2) выбрать алгоритм построения реалистичного изображения;  3) разработать функциональную модель программного обеспечения;  4) выбрать средства реализации и реализовать программное обеспечение для визуализации сцены;  5) исследовать зависимость времени генерации кадра от числа полигонов на сцене для варьируемого числа рабочих потоков. |
| Слайд 3  Для визуализации сцены в приложении должны быть заданы: **источник света,** который задается в программе константанами, **камера**, она же наблюдатель, которая определяется вектором взгляда и положением в пространстве и модели фигур. На выходе мы получаем кадр. |
| Слайд 4  На сцене могут быть изображены:  - шахматная доска, 8х8 клеток, с деревянным основанием, клетками черного и белого цветов  - шахматные фигуры, перечисленные на слайде номер 4, представленные в двух цветах.  Форма и размер шахматных фигур и шахматной доски соответствует стандарту шахматного оборудования и игровых площадок, предназначенных для проведения турниров Международной шахматной федерации (ФИДЕ, Fédération Internationale des Échecs). Пример шахматных фигур, приведенный в качестве возможно используемых моделей в соревнованиях ФИДЕ приведен на слайде 4. Данные фигуры оформлены в стиле Стаунтон, которые рекомендуется для применения. |
| Слайд 5  В программе была использована глобальная модель освещения. Она учитывает взаимное расположение объектов и описывается законом, который изображен на слайде номер 5. |
| Слайд 6  Одним из этапов построении реалистичного изображения является удаление невидимых линий. Сравнение алгоритмов, решающих данную задачу представлено на слайде номер 6.  Для генерации кадра был выбран алгоритм обратной трассировки лучей в связи с целями задачи. И так как этот метод, для закраски сцены и создания теней не требует дополнительных алгоритмов, он также является и решением задачи построения реалистичного изображения. |
| Слайд 7  Функциональная модель программного обеспечения с учетом выбранного алгоритма в формате idef0 нулевого уровня представлена на слайде номер 7 |
| Слайд 8  Декомпозированное программное обеспечение представлено на слайде номер 8. В программе присутствует рекурсивный цикл, ограниченный глубиной трассировки равной двум рекурсивным вызовам, так |
| Слайд 9  На слайде номер 9 приведен пример алгоритма трассировки лучей, который определяет цвет пикселя для определенного луча используя глобальную модель освещения без преломления света. |
| Слайд 10  В качестве средств реализации был выбрал язык С++, модульное тестирование было реализовано с помощью фреймворка GoogleTest и покрытие кода вычислялось с помощью утилиты gcov. |
| Слайд 11  На слайде номер 11 вы можете видеть диаграмму классов. При реализации программного обеспечения использовались такие паттерны как фасад (FacadeScene), стратегия (TransformAction), команда, он представлен здесь одним базовым классом BaseCommand, его реализации были вынесены с целью сохранения читабельности диаграммы, так же представлены паттерны директор и компоновщик, для создания классов моделей фигур. |
| Слайд 12  Программа была покрыта модульными тестами. Были созданы наборы тестов для основных вычислительных конструкций, для методов класса Camera, RayTracing, Ray, Triangle.  В качестве метрики успешности модульного тестирования было выбрано покрытие кода. Были покрыты тестами 35% процентов строк кода. |
| Слайд 13  Было проведено функциональное тестирование, которое заключалось в рассмотрении нескольких положений камеры для различной расстановки фигур относительно друг друга при использовании глянцевой и матовой поверхности шахматной доски. На слайде номер 13 приведены наиболее показательные ракурсы. |
| Слайд 14  В качестве исследования была выбрана зависимости времени генерации кадра от числа полигонов на сцене для варьируемого числа рабочих потоков, поскольку количество полигонов, которыми задаются модели шахматных фигур определяет точность их визуального восприятия.  Результат вы можете видеть на графике, представленном на слайде номер 14. По данному графику были сделаны выводы что при количестве полигонов на сцене большем 200, наиболее эффективными для машины на которой производились тесты является использования 32 потоков. |
| Слайд 15  Все поставленные задачи были решены, и поставленная цель достигнута. Спасибо за внимание, готова продемонстрировать ПО. |