|  |  |
| --- | --- |
|  | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана**  **(национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

ФАКУЛЬТЕТ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

КАФЕДРА \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**ОТЧЕТ ПО ПРОИЗВОДСТВЕННОЙ ПРАКТИКЕ**

Студент\_\_\_\_\_\_\_Чепрасов Кирилл Михайлович\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*фамилия, имя, отчество*

Группа\_\_ИУ7-46Б\_\_\_\_\_\_\_\_\_\_

Тип практики \_\_\_Технологическая\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Название предприятия\_\_\_\_\_\_МГТУ им. Баумана\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*подпись, дата фамилия, и.о.*

Руководитель практики **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

*подпись, дата фамилия, и.о.*

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*2022 г.*

УТВЕРЖДАЮ

Заведующий кафедрой ИУ7

(Индекс)

\_\_\_\_\_\_\_\_\_\_\_\_\_ И.В.Рудаков

(И.О.Фамилия)

« \_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_ 2022 г.

**З А Д А Н И Е**

**на выполнение курсовой работы**

по дисциплине\_\_\_\_\_\_\_\_Компьютерная графика\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_Программа моделирования зеркального кубика Рубика методом трассировки лучей\_\_\_\_

(Тема курсового проекта)

Студент\_\_\_\_\_\_\_\_\_\_\_Чепрасов К.М. гр. ИУ7-56\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Фамилия, инициалы, индекс группы)

График выполнения проекта: 25% к \_4 нед., 50% к \_7\_ нед., 75% к \_11нед., 100% к 14 нед.

***1. Техническое задание***

Разработать программу моделирования зеркального кубика рубика, с возможностью вращения его гране методом трассировки лучей. Кубик Рубика должен находится в рамках некой сцены, представляющую собой набор простых трёхмерных фигур. Зеркальный кубик Рубика должен представлять собой набор из 27 параллелепипедов (по 9 с каждой грани), образующих куб, каждого ребро которого делится параллелепипедами в отношении 1:2:3. Предусмотреть возможность отрисовки сцены по нажатию кнопки, поворота камеры относительно кубика Рубика. Исследовать возможность учёта освещённости\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***2. Оформление курсовой работы***

2.1. Расчетно-пояснительная записка на 25-30 листах формата А4.

Расчетно-пояснительная записка должна содержать постановку введение, аналитическую часть, конструкторскую часть, технологическую часть, экспериментально-исследовательский раздел, заключение, список литературы, приложения.

2.2. Перечень графического материала (плакаты, схемы, чертежи и т.п.)\_\_На защиту проекта должна быть представлена презентация, состоящая из 15-20 слайдов. На слайдах должны быть отражены: постановка задачи, использованные методы и алгоритмы, расчетные соотношения, структура комплекса программ, диаграмма классов, интерфейс, характеристики разработанного ПО, результаты проведенных исследований.

Дата выдачи задания « \_\_\_\_ » \_\_\_\_\_\_\_\_\_\_\_\_\_ 20\_\_ г.

**Руководитель курсовой работы** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_Куров А.В.

(Подпись, дата) (И.О.Фамилия)

**Студент \_\_\_**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**\_\_**  **\_** Чепрасов К.М.

(Подпись,дата) (И.О.Фамилия)

Примечание:

1. Задание оформляется в двух экземплярах; один выдаётся студенту, второй хранится на кафедре.

# ОГЛАВЛЕНИЕ

# ВВЕДЕНИЕ

Цель практики: исследование обоснование .

  
*(Рис 1. внешний вид зеркального кубика Рубика в собранном (слева) и в разрбранном (справа) состояниях)*

Основной задачей является создание программного продукта, позволяющего взаимодействовать с гранями упомянутого кубика Рубика. Необходимо было так-же создать сцену, в рамках которой он будет находиться, и предоставить пользователю интерфейс, позволяющий производить вращение кубика.

# ОСНОВНАЯ ЧАСТЬ

Среди них:

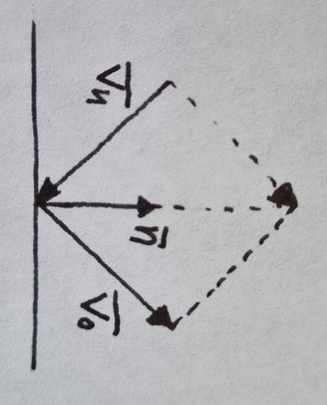
* трёхмерный (Vector3D) и двумерный (Vector2D) вектора,
* угол (Angle),
* тройка углов (EulerAngles),
* положение объекта на сцене (Transform),
* объект сцены (SceneObject),
* видимый объект сцены (VisibleObject),
* трёхмерная модель (Model3D),
* камера (Camera).

Были добавлены структуры, описывающие вершины (Vertex) и поверхности (Surface). Поверхность связывает тройку вершин, и задаёт коэффициент диффузии. При использовании технологии трассировки лучей, при каждом столкновении луча с поверхностью образуются два дополнительных луча: отражённый и преломлённый. Так как в данном проекте преломления света не происходит, луч только лишь отражается. Таким образом, необходимо совместить диффузию и отражение. Описать их соотношение я решил используя лишь одно число: коэффициент диффузии (). Так, например, если , то поверхность не является отражающей. Если , то поверхность является идеальным зеркалом. При любом промежуточном значении, итоговый цвет вычисляется как:

где: *C* – итоговый цвет, – цвет диффузии, – цвет отражения, – коэффициент диффузии. Цвет представляет собой структуру из трёх чисел: коэффициент красного, зелёного и синего оттенков. (далее r, g, b). Функция определена как цвет, со следующими компонентами:

Здесь r, g, b – компоненты итогового цвета, – компоненты цвета ; – компоненты цвета ; *k* – коэффициент смешения.

При расчёте цвета пикселя, для определения цвета отражения запускается ещё один луч, зеркально отражённый от поверхности столкновения.

  
*(Рис. 2: графическое представление вектора отражения)*

Используя Рис. 2, нетрудно увидеть, что вектор отражения можно получить, если из из исходного вектора направления вычесть две его проекции на нормаль. Получаем:

Где – отражённый вектор, – исходный вектор, – вектор нормали к поверхности. Вектор нормали нормируется на этапе инициализации модели, то есть мы можем гарантировать, что его модуль равен единице. Таким образом, формула (2) упрощается, получаем:

# ЗАКЛЮЧЕНИЕ

# ИСТОЧНИКИ

# ПРИЛОЖЕНИЯ