

## Лабораторные работы 3-4-5-6

**Тема: Основы построения фотореалистичных изображений. Ознакомление с технологией OpenGL. Создание шейдерных анимационных эффектов в OpenGL 2.1**

### Материалы

Методические указания к лабораторным работам по компьютерной графике.

### Условие

Используя результаты Л.Р. #2 и #3, аппроксимировать заданное тело выпуклым многогранником, создать графическое приложение с использованием OpenGL, изобразить заданное тело с использованием средств OpenGL 2.1. Точность аппроксимации задается пользователем. Обеспечить возможность вращения и масштабирования многогранника и удаление невидимых линий и поверхностей. Реализовать простую модель закраски для случая одного источника света. Параметры освещения и отражающие свойства материала задаются пользователем в диалоговом режиме. Реализовать простую модель освещения на GLSL. Параметры освещения и отражающие свойства материала задаются пользователем в диалоговом режиме.

Вариант 12: Прямой круговой цилиндр. Создание анимации по формуле:  $y = y * \cos(t + y)$

### Описание программы

Программа написана на JavaScript . С использованием библиотеки p5 и WebGL (вместо OpenGL).


### Структура программы

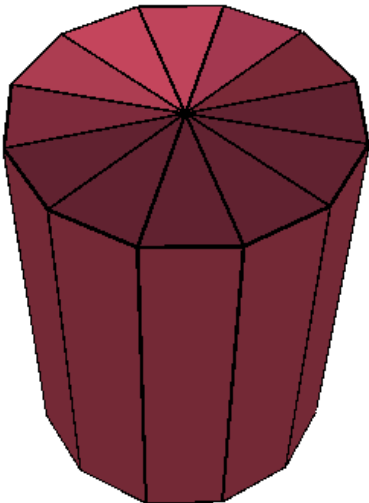
Программа состоит из 1 файла: main.js - описание работы приложения.

Стадии инициализации приложения:

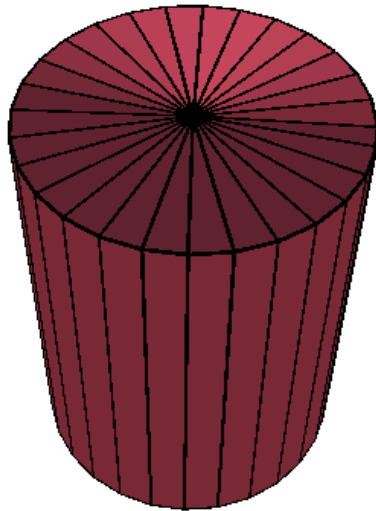
- Цилиндр рисуется за два шага в функции drawCylinder - создаются боковые стороны и основания. Отрисовка происходит полигонами по кругу. Основания состояются из треугольников, которые находятся на заданном расстоянии. Так же для построения оснований используются окружности заданного радиуса.
- Метод beginShape() - начинает записывать вершины фигуры, а endShape() - заканчивает. Далее следуют методы vertex() с тремя параметрами, которые задают позицию вершины в 3D-пространстве. Цвет контура каждой фигуры задается с помощью метода stroke() .
- Функция setup() :
  - Метод createCanvas(WIDTH, HEIGHT, WEBGL) - создает поле для изображения фигуры, заданы размеры поля и renderer = WebGL , что включают 3D-рендеринг, вводя третье измерение: Z.
  - debugMode() - помогает визуализировать 3D-пространство, добавляя сетку, добавляющую к эскизу значок осей, указывающих направления X, Y и Z.
- Функция draw() :
- ambientLight(100) - создает свет
- directionalLight - создает направленный свет с цветом и направлением
- orbitControl - позволяет перемещаться по 3D эскизу с помощью мыши или трекпада

Демонстрация работы

 n - number of sides



 n - number of sides



## Выводы

С P5.js можно легко создавать бесконечно большое количество вариантов отображения информации: от веб-формата, статичных картинок для печати, до анимированных 3d фигур. В целом P5.js - это хороший инструмент для рисования. Также вместо технологии OpenGL была использована технология WebGL для настройки положения камеры, света. Данную опцию 3D-рендеринга смело можно назвать аналогом OpenGL, но лишь для веб-приложений.