МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по лабораторной работе №6

по дисциплине «Компьютерная графика»

Тема: Реализация трехмерного объекта с использованием библиотеки OpenGL

Студентка гр. 8383 _____ Максимова А.А.

Студент гр. 8383 _____ Ларин А.

Преподаватель _____ Герасимова Т.В.

Санкт-Петербург

2021

Задание

Разработать программу, реализующую представление разработанного вами трехмерного рисунка, используя предложенные функции библиотеки OpenGL (матрицы видового преобразования, проецирование) и язык GLSL.

Разработанная программа должна быть пополнена возможностями остановки интерактивно различных атрибутов через вызов соответствующих элементов интерфейса пользователя, замена типа проекции, управление преобразованиями, как с помощью мыши, так и с помощью диалоговых элементов.

Выполнение

Задание вполнено на языке C++ с использованием фреймворка Qt5.
Qt предоставляет возможность использование OpenGL посредством класса QGLWidget. От него отнаследован класс CustomGL.
`class CustomGL : public QGLWidget, в котором переопределены методы:`

- `void initializeGL();`
  Задает контекст рендеринга. Инициализирует шейдеры
- `void resizeGL(int nWidth, int nHeight);`
  `Для корректного масштабирования. Размер окна задается через glViewport.`
- `void paintGL();`
  Вызов этой функции происходит при рисовании.

В конструкторе класса CustomGL происходит создание шейдерной программы
В функции initializeGL происходит вызов initShaders() в которой загружается код шейдеров, затем линкуется и подгружается в шейдерную программу

```
shaderProgram->addShaderFromSourceCode(QGLShader::Fragment,
_fsh.c_str()
```

Затем происходит первичная генерация многогранника. Многогранник представляет из себя икосферу с заданным числом разбиений

При рисовании и генерации многогранника программа оперирует множеством параметров, принимаемых с графического интерфейса среди них степень

разбиения, интенсивность освещения, прорисовка граней, тип проекции, положение многогранника, а также его вращение и позиция.

Перемещение камеры происходит при помощи мыши

В функции paintGL происходит задание типа прорисовки и теста глубины, после чего управление передается в функцию scene.
В ней происходит настройка матрици трансформации(тип проекции, положение вращение и пр.), после чего последовательно отрисовываются оси координат и сам многогранник. Для их отрисовки используется шейдер.
В шейдер передаются вершины, нормали, цвета, матрица трансформации и источник света. Интенсивность света высчитывается с учетом коэффициента задаваемого с интерфейса и направлением нормали. На источник ярче, от источника тусклее. В коде шейдера это выглядит так:
gl_FragColor = vec4(vColor.xyz,1.0f) + vec4(val,val,val,1)*
dot(normalize(lightsource.xyz-position),normal);

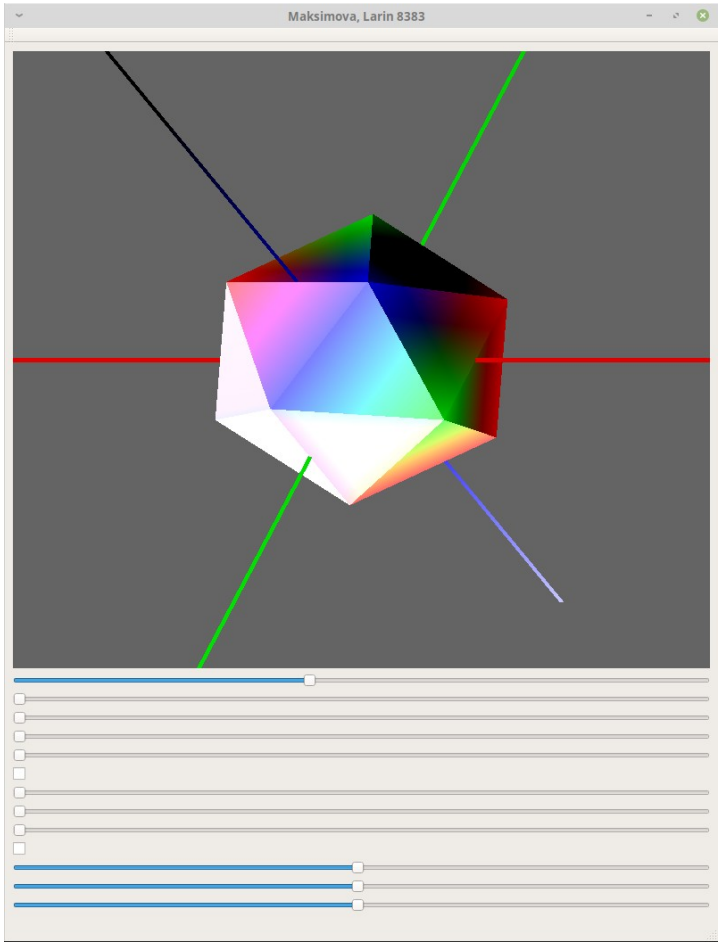
Здесь:
vColor — цвет вершины
val — интенсивность света с учетом расстояния и коэфициента
position — положение точки
lightsource.xyz — положение источника (это vec4, в четвертой компоненте передана интенсивность для val)
normal — нормаль плоскости, закладывается в каждую вершину

Генерация икосферы происходит следующим образом:
Начала создается три плоскости лежащие в плоскостях кординат, с отношением сторон равному золотому сечению. Далее углы этих плокостей соединяются по три, образуя икосферу

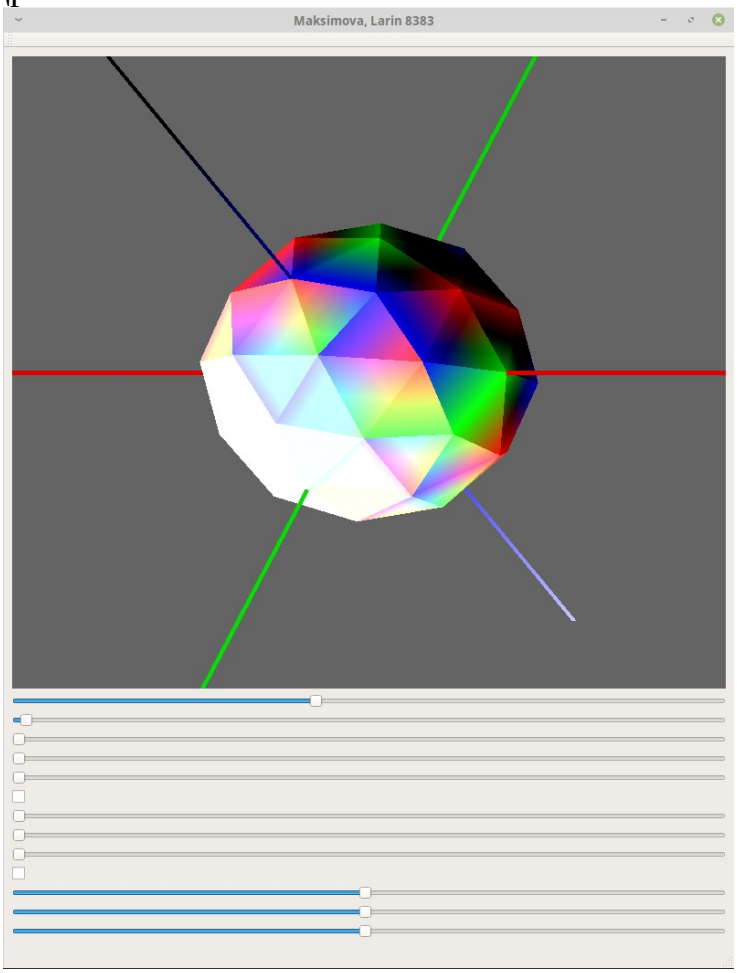
Далее каждая треугольная грань делится на четыре, на манер итерации треугольника серпинского. Каждая новая вершина выносится на поверхность сферы. При увеличекии числа дроблений многогранник начинает все больше походить на сферу.

При отрисовке многогранника матрица сохраняется, дополнительно изменяется в соответствие с вращением, положением и масштабированием самого многогранника(камера и оси координат при этом остаются на месте)

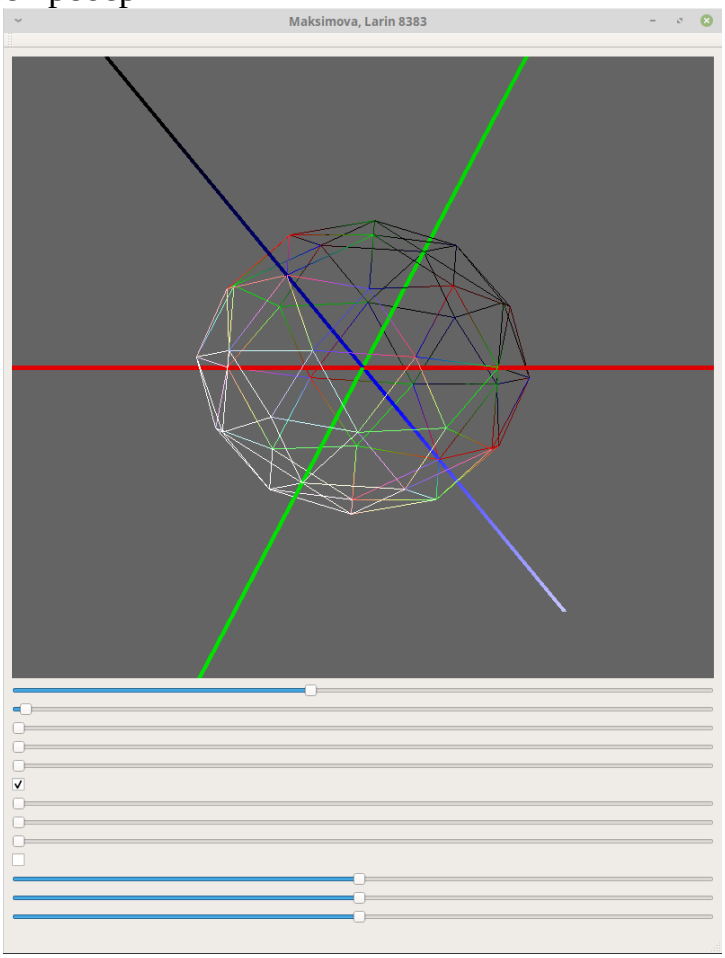
Вид многогранника без подразбиения:



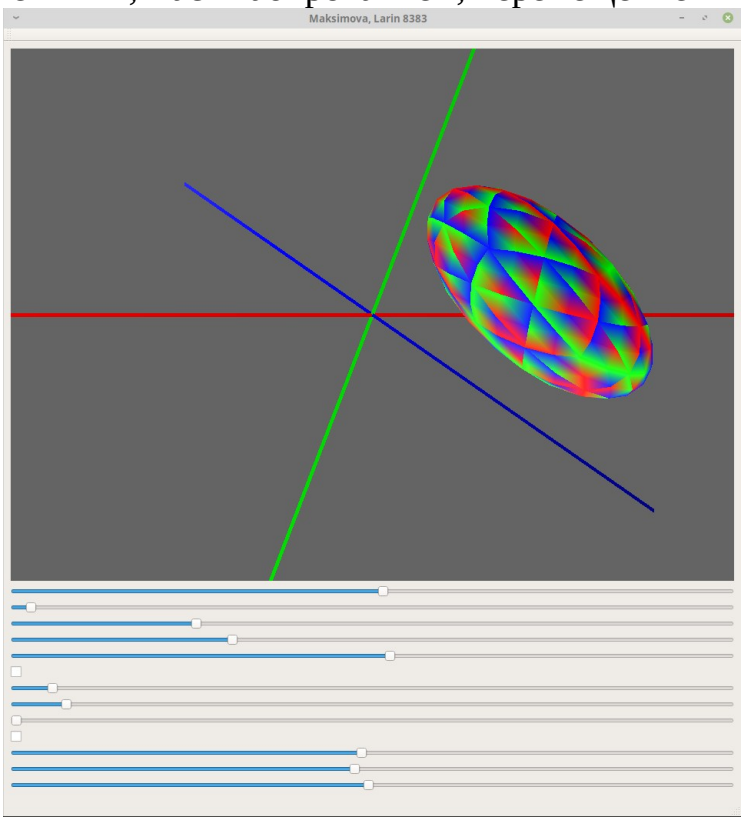
После одного подразбиения:



С прорисовкой всех ребер



С тремя подразбиениями, масштабированием, перемещением и поворотом:



Выводы.

В результате выполнения лабораторной работы была написана программа, реализующая подключение и использование шейдеров. Был сгенерирован трехмерный объект и реализована интерактивная работа с ним.