

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

**ОТЧЕТ**  
**по лабораторной работе №1**  
**по дисциплине «Компьютерная графика»**  
**Тема: Создание примитивов.**

Студент гр. 9304

\_\_\_\_\_

Попов Д.С.

Преподаватель

\_\_\_\_\_

Герасимова Т.В.

Санкт-Петербург

2022

### **Цель работы.**

Ознакомиться с примитивами и реализовать программу по их отображению.

### **Задание.**

Разработать программу, реализующую представление определенного набора примитивов из имеющихся в библиотеке OpenGL (GL\_POINT, GL\_LINES, GL\_LINE\_STRIP, GL\_LINE\_LOOP, GL\_TRIANGLES, GL\_TRIANGLE\_STRIP, GL\_TRIANGLE\_FAN, GL\_QUADS, GL\_QUAD\_STRIP, GL\_POLYGON).

Разработанная на базе шаблона программа должна быть пополнена возможностями остановки интерактивно различных атрибутов примитивов рисования через вызов соответствующих элементов интерфейса пользователя.

### **Общие сведения.**

В данной лабораторной работе были рассмотрены следующие примитивы:

**GL\_LINES** – каждая пара вершин рассматривается как независимый отрезок. Первые две вершины определяют первый отрезок, следующие две – второй отрезок и т.д., вершины  $(2n-1)$  и  $2n$  определяют отрезок  $n$ . Всего рисуется  $N/2$  линий. Если число вершин нечетно, то последняя просто игнорируется.

**GL\_TRIANGLES** – каждая тройка вершин рассматривается как независимый треугольник. Вершины  $(3n-2)$ ,  $(3n-1)$ ,  $3n$  (в таком порядке) определяют треугольник  $n$ . Если число вершин не кратно 3, то оставшиеся (одна или две) вершины игнорируются. Всего рисуется  $N/3$  треугольника.

**GL\_TRIANGLE\_FAN** - в этом режиме рисуется группа связанных треугольников, имеющих общие грани и одну общую вершину. Первые три

вершины определяют первый треугольник, первая, третья и четвертая – второй и т.д. Всего рисуется (N-2) треугольника.

**GL\_QUADS** – каждая группа из четырех вершин рассматривается как независимый четырехугольник. Вершины  $(4n-3)$ ,  $(4n-2)$ ,  $(4n-1)$  и  $4n$  определяют четырехугольник  $n$ . Если число вершин не кратно 4, то оставшиеся (одна, две или три) вершины игнорируются. Всего рисуется  $N/4$  четырехугольника.

**GL\_POLYGON** – задает многоугольник. При этом число вершин равно числу вершин рисуемого многоугольника.

### Выполнение работы.

В качестве среды разработки был выбран фреймворк Qt (версия 5.12.12) под операционную среду Linux(Ubuntu 20.04). Для создания каждого примитива был реализован свой класс:

Примитив	Класс
GL_LINES	Line
GL_TRIANGLES	Triangle
GL_TRIANGLE_FAN	Pyramide
GL_QUADS	Cube
GL_POLYGON	Polygon

В каждом классе имеется метод задающий вершины примитива. Набор вершин для создания треугольника представлен ниже:

```
glBegin(GL_TRIANGLES);  
    glColor4f(1.0f, 0.0f, 0.0f, 0.7f);  
    glVertex3f(0.0, size, 0.0);  
    glVertex3f(-size, -size, size);  
    glVertex3f(size, -size, size);  
glEnd();
```

Все классы наследуются от единого интерфейса, который реализует все общие методы отрисовки OpenGL. Внешний вид приложения представлен на рисунке 1.

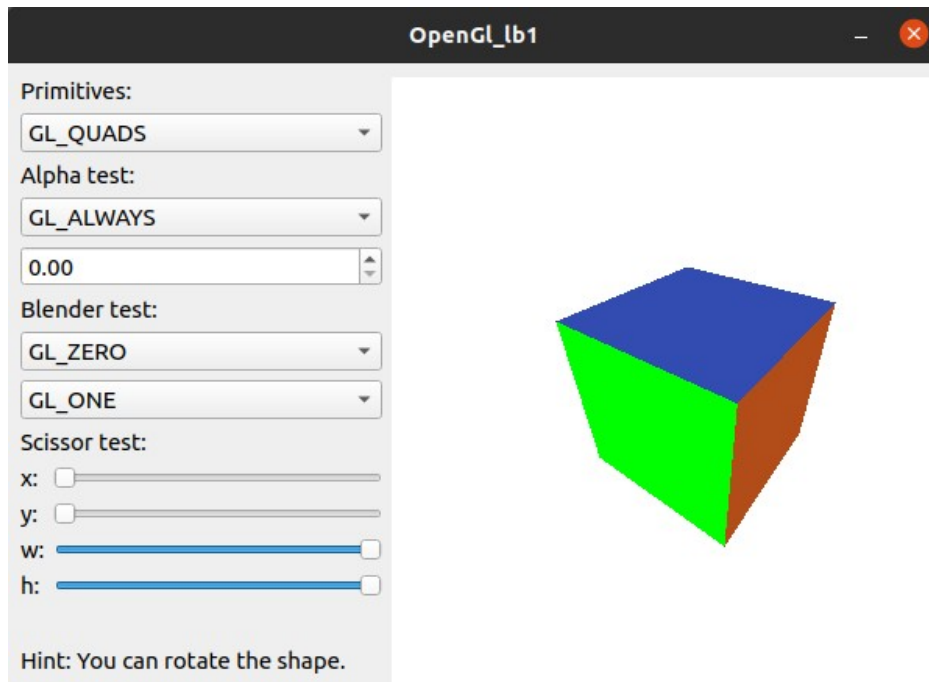


Рисунок 1 — Графический интерфейс программы.

Изображения примитивов представленный на рисунках 2-6.

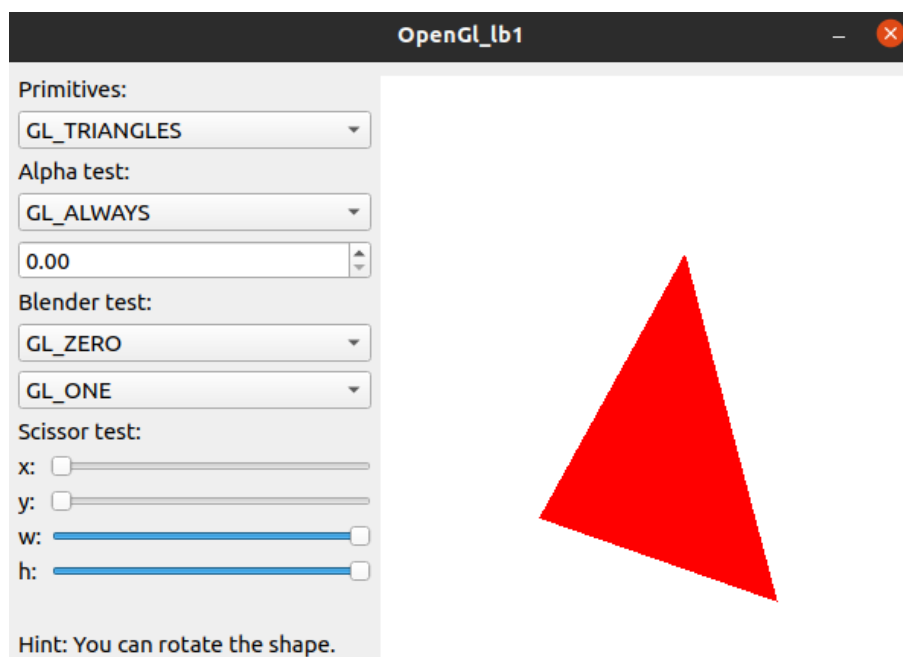


Рисунок 2 — Примитив GL\_TRIANGLES.

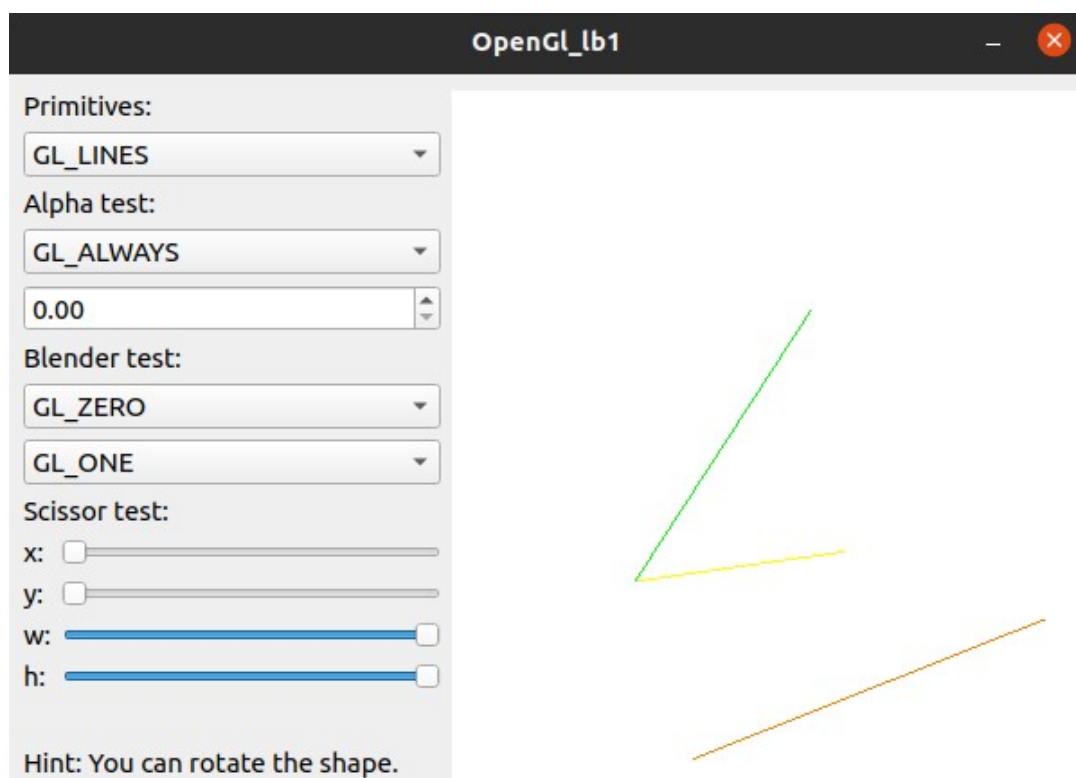


Рисунок 3 — Примитив GL\_LINES.

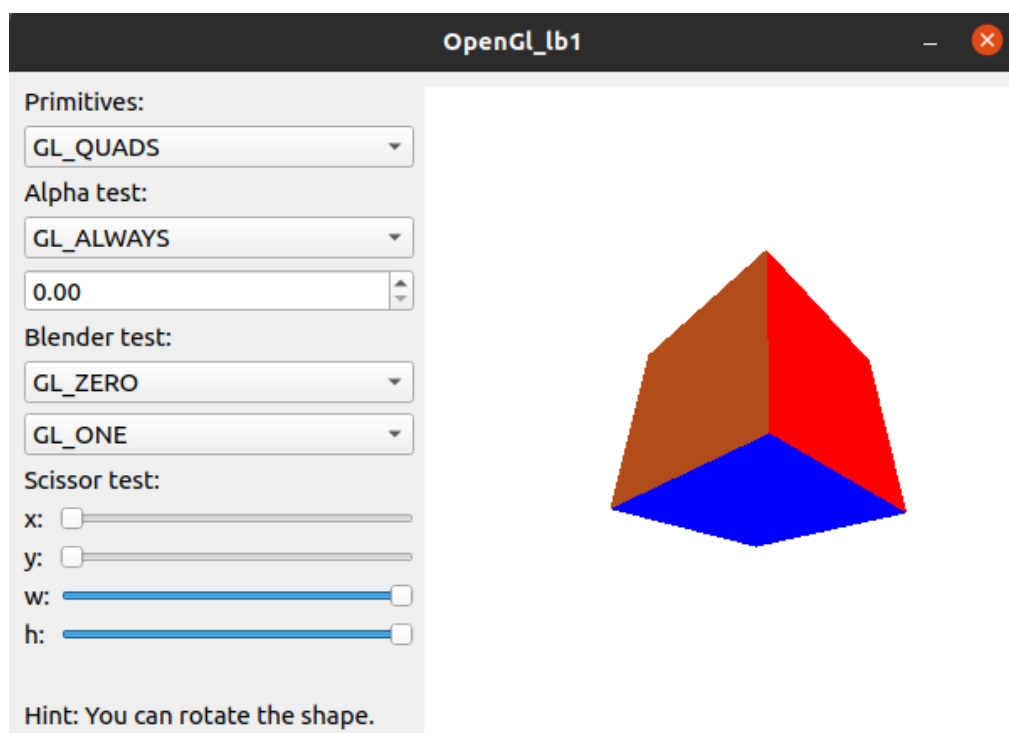


Рисунок 4 — Примитив GL\_QUADS.

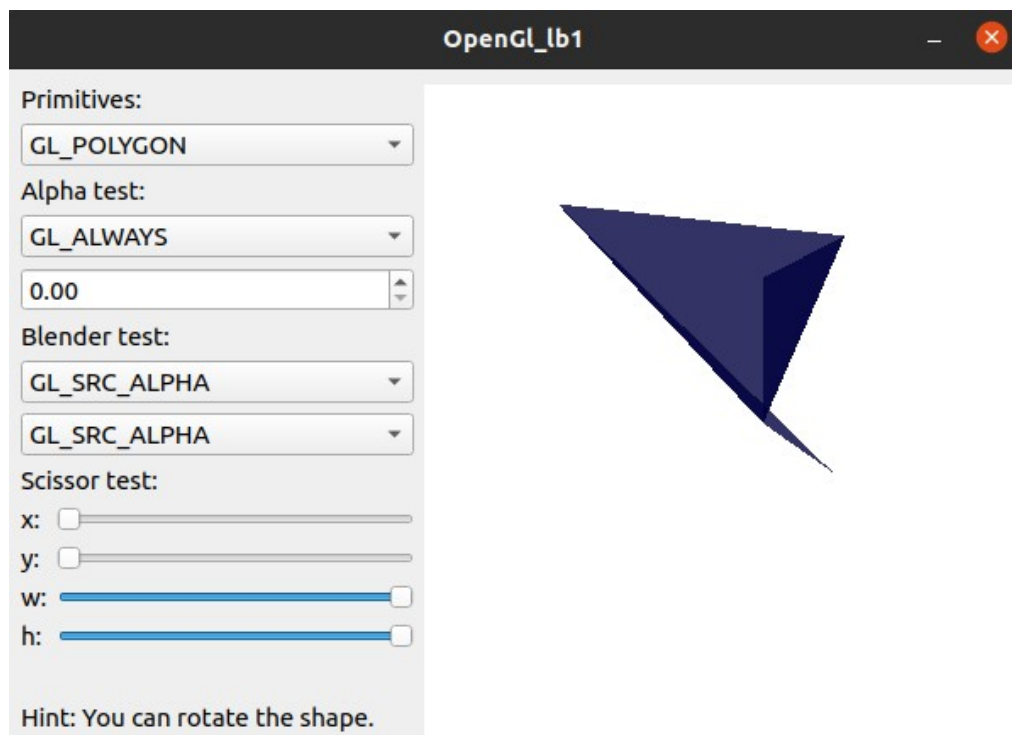


Рисунок 5 — Примитив GL\_POLYGON.

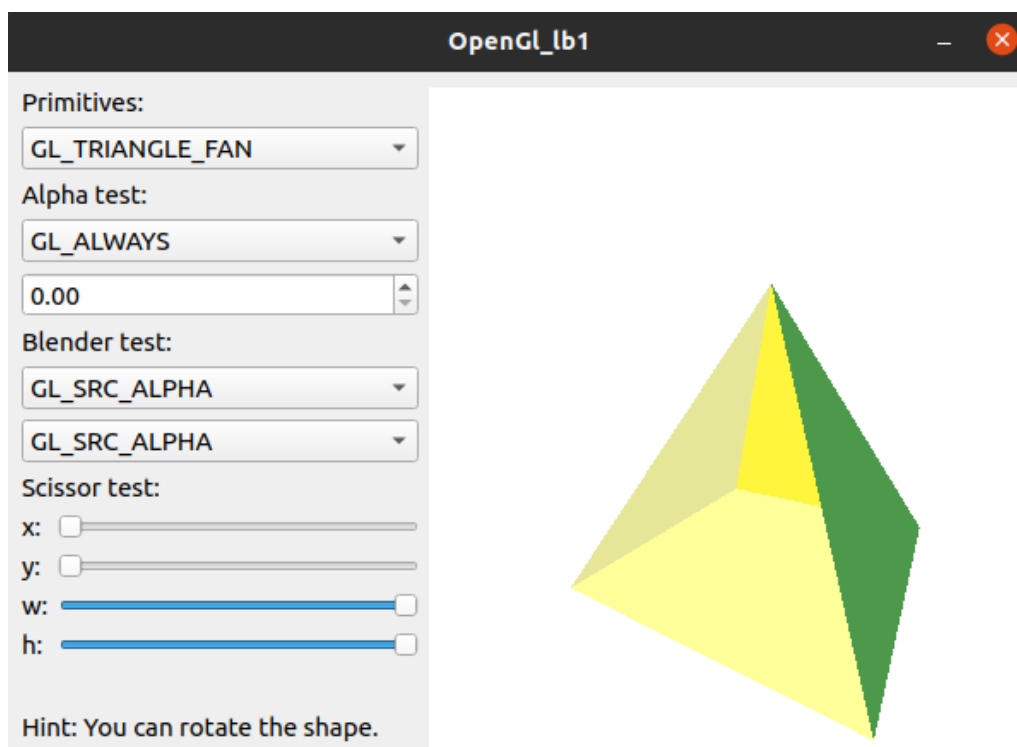


Рисунок 6 — Примитив GL\_TRIANGLE\_FAN.

### **Выводы.**

В результате выполнения лабораторной работы была разработана программа, создающая графические примитивы OpenGL. Программа работает корректно. При выполнении работы были приобретены навыки работы с графической библиотекой OpenGL.