

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: Прimitives OpenGL.

Студент гр. 9304

Атаманов С.Д.

Студент гр. 9304

Силкин В.А.

Преподаватель

Герасимова Т.В.

Санкт-Петербург

2022

Цель работы.

Ознакомление с основными примитивами OpenGL.

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.

Основные теоретические положения.

GL_POINTS –каждая вершина рассматривается как отдельная точка, параметры которой не зависят от параметров остальных заданных точек. При этом вершина n определяет точку n . Рисуется N точек (n –номер текущей вершины, N –общее число вершин).

Основой графики OpenGL являются вершины. Для их определения используется команда glVertex:

```
void glVertex[2 3 4][s i f d](type coord)
```

Вызов команды определяется четырьмя координатами x , y , z и w . При этом вызов glVertex2* устанавливает координаты x и y , координата z полагается равной 0, а $w = 1$. Вызов glVertex3* устанавливает координаты x , y , z , а w равно 1.

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.

3GL_LINE_STRIP –в этом режиме рисуется последовательность из одного или нескольких связанных отрезков. Первая вершина задает начало первого отрезка, а вторая –конец первого, который является также началом второго. В общем случае, вершина n ($n > 1$) определяет начало отрезка n и конец отрезка $(n - 1)$. Всего рисуется $(N - 1)$ отрезок.

GL_LINE_LOOP –осуществляется рисование замкнутой кривой линии. Первая вершина задает начало первого отрезка, а вторая –конец первого, который является также началом второго. В общем случае, вершина n ($n > 1$) определяет начало отрезка n и конец отрезка $(n - 1)$. Первая вершина является концом последнего отрезка. Всего рисуется N отрезков.

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_STRIP –в этом режиме рисуется группа связанных треугольников, имеющих общую грань. Первые три вершины определяют первый треугольник, вторая, третья и четвертая –второй и т.д. для нечетного n вершины n , $(n+1)$ и $(n+2)$ определяют треугольник n . Для четного n треугольник определяют вершины $(n+1)$, n и $(n+2)$. Всего рисуется $(N-2)$ 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_QUAD_STRIP –рисуется группа четырехугольников, имеющих общую грань. Первая группа из четырех вершин задает первый

4четырехугольник. Третья, четвертая, пятая и шестая задают второй четырехугольник.

GL_POLYGON –задет многоугольник. При этом число вершин равно числу вершин рисуемого многоугольника.

Выполнение работы.

1. Средой выполнения работы был выбран фреймворк Qt. Данный фреймворк по умолчанию имеет средства работы с OpenGL.
2. Отображение графики выполняется с помощью класса QOpenGLWidget. Был создан класс glScene, который наследуется от класса QOpenGLWidget, а также выполнена перегрузка функций инициализации OpenGL, отрисовки графики и изменения размеров окна.

Результат работы программы представлен на рисунках 1-4.

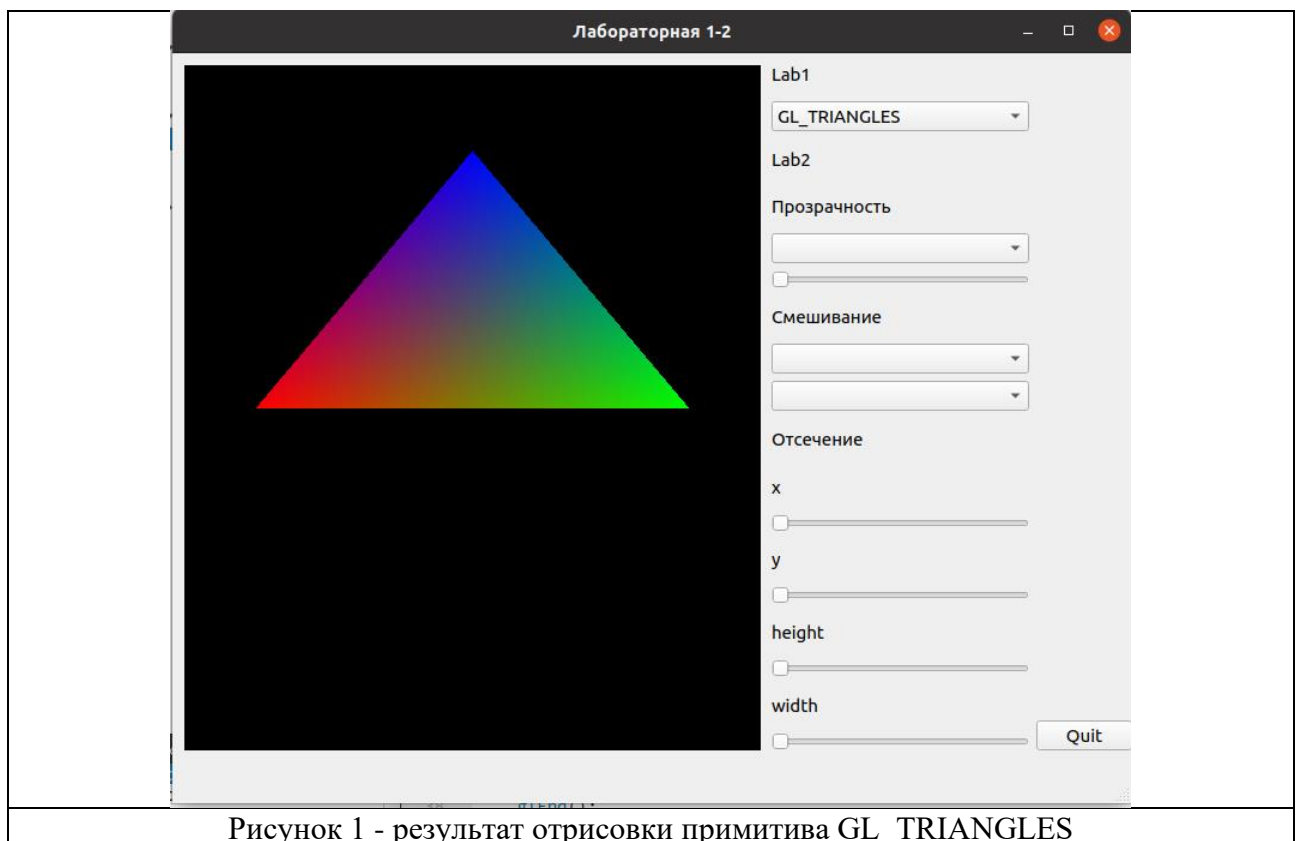


Рисунок 1 - результат отрисовки примитива GL TRIANGLES

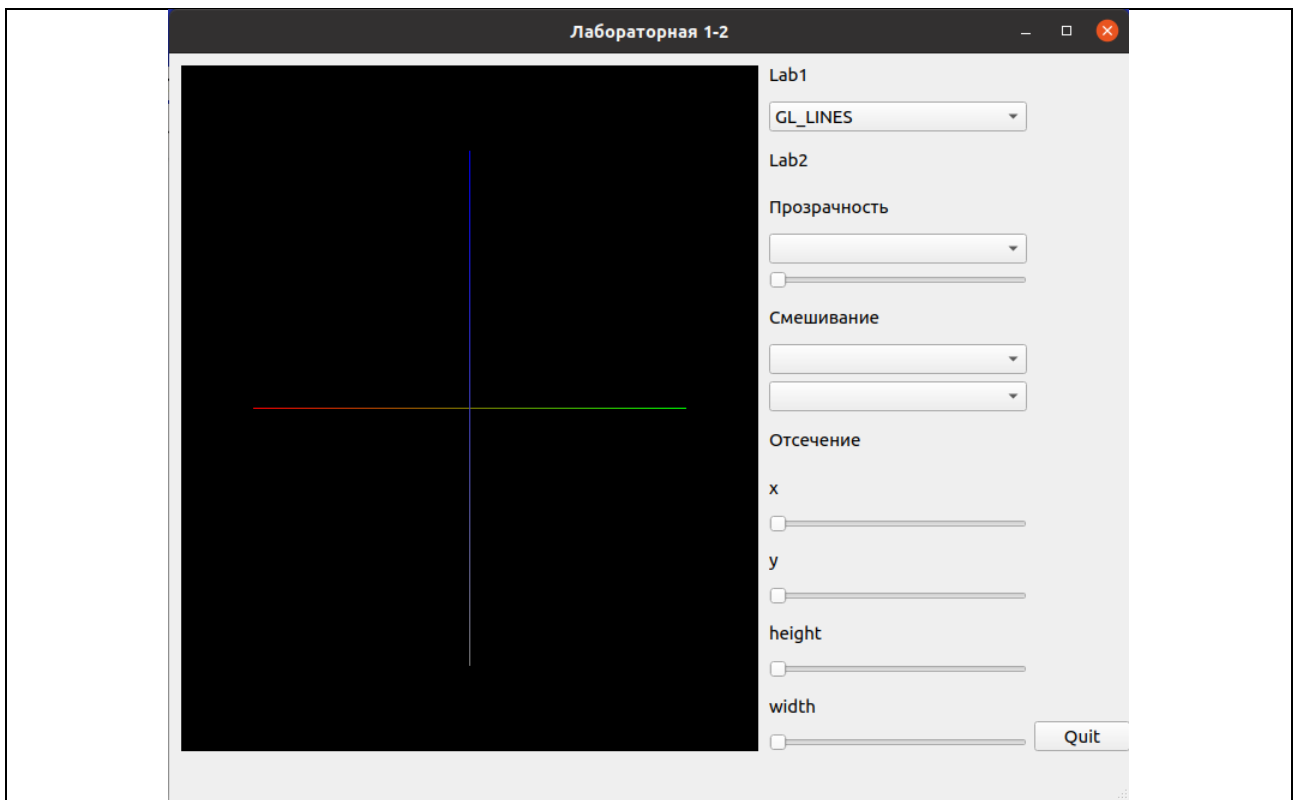


Рисунок 2 - результат отрисовки примитива GL_LINES

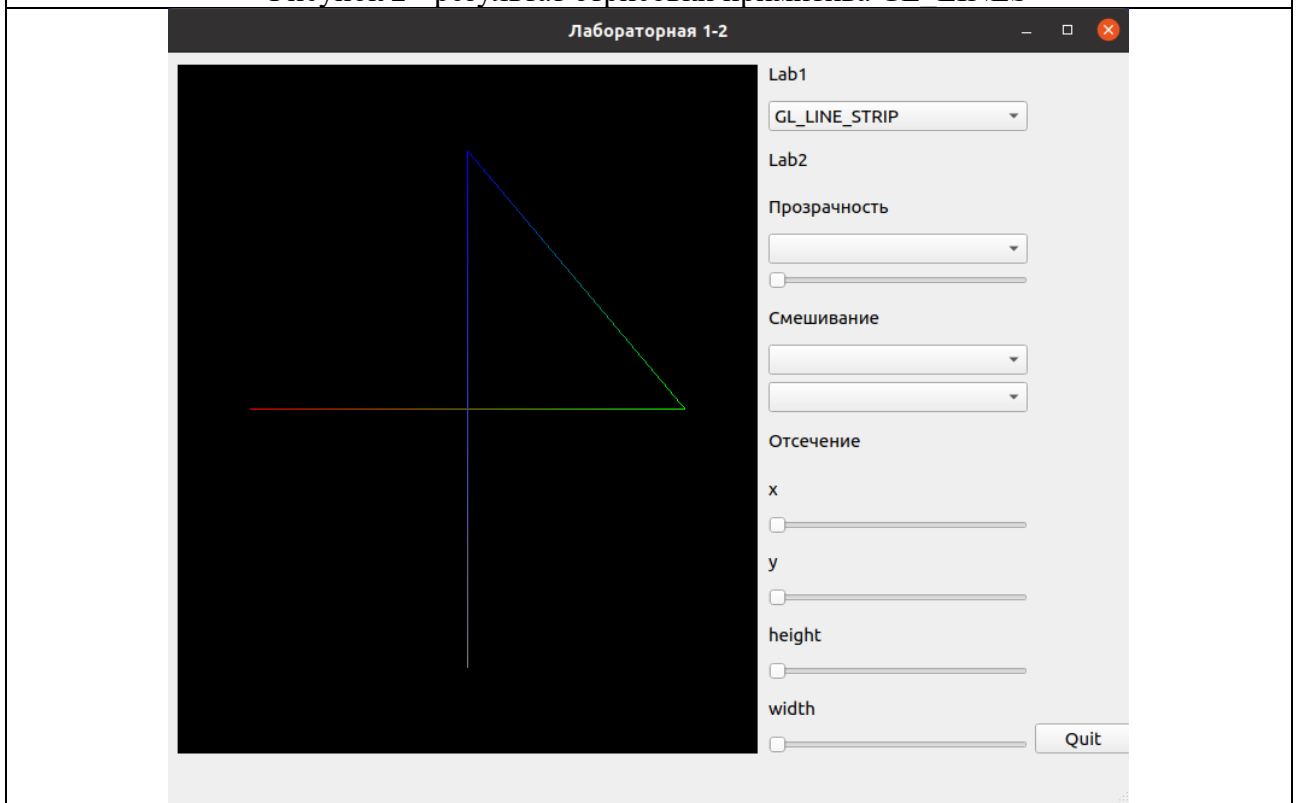


Рисунок 3 - результат отрисовки примитива GL_LINE_STRIP

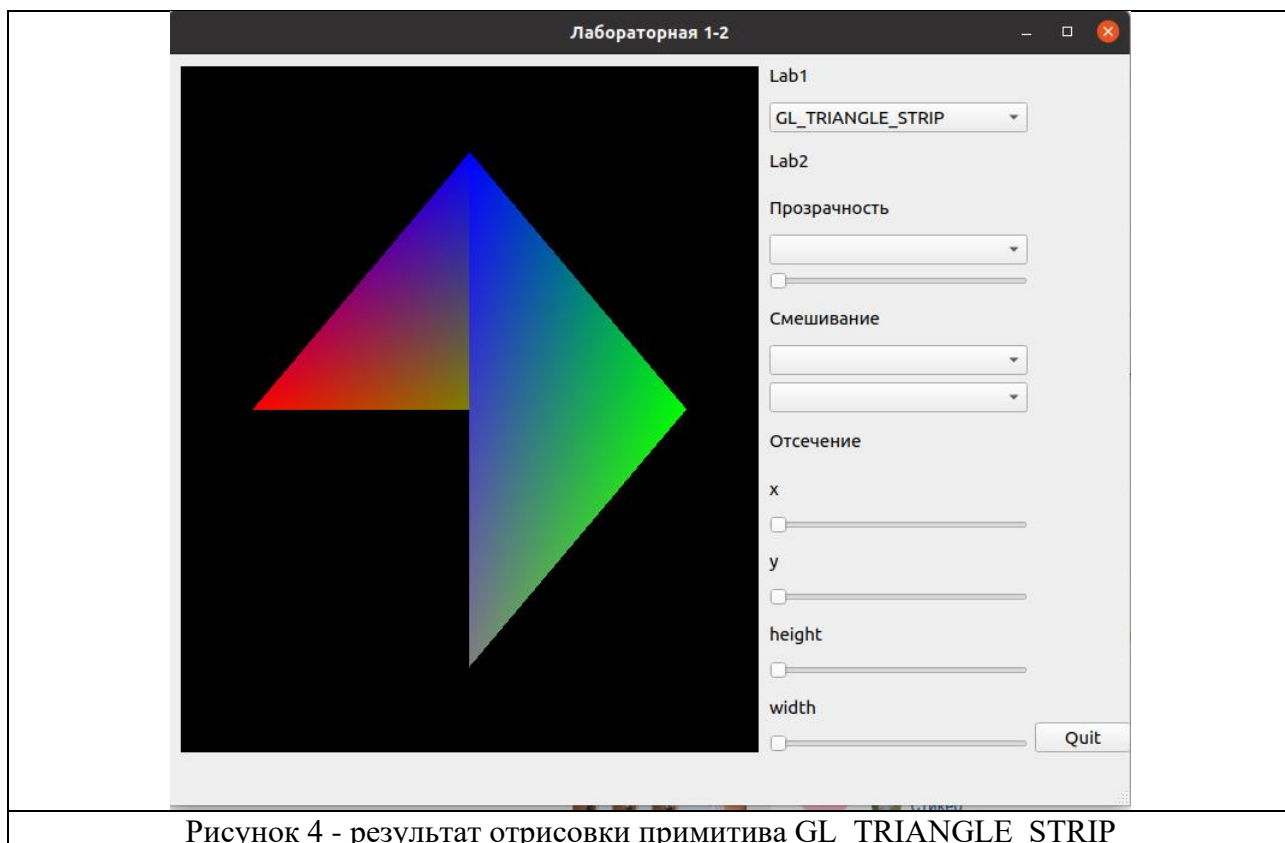


Рисунок 4 - результат отрисовки примитива GL TRIANGLE STRIP

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.