Московский авиационный институт

(Национальный исследовательский университет)

**Институт «Информационных технологий и прикладной математики»**

**Лабораторная работа №4-5**

Ознакомление с технологией OpenGL

Работу выполнил:

Рябыкин Алексей Сергеевич

Группа: М8О-309Б-18

Преподаватель: Филиппов Г.С.

Оценка:

Дата:

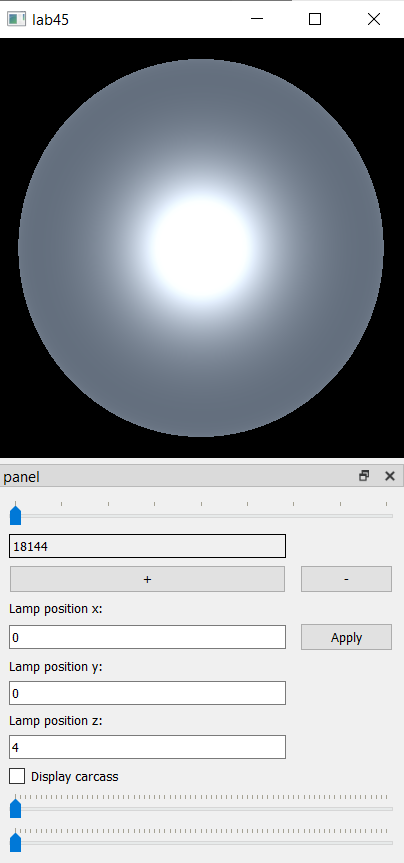
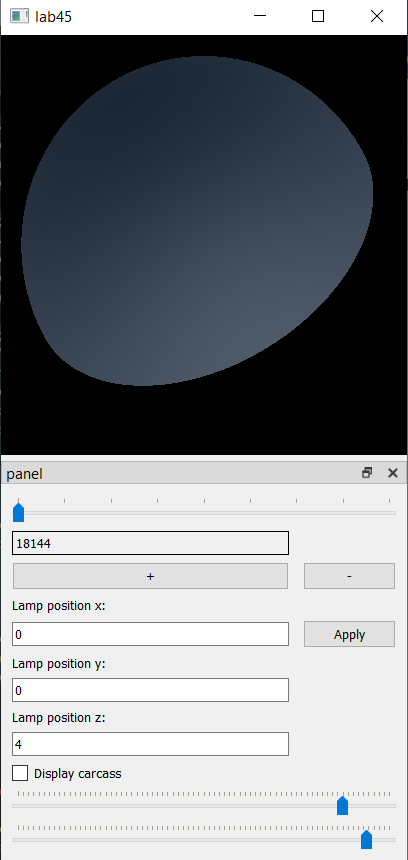
**Москва, 2020**

**Постановка задачи**

Создать графическое приложение с использованием OpenGL. Используя результаты Л.Р.№3, изобразить заданное тело (то же, что и в л.р. №3) с использованием средств OpenGL 2.1. Использовать буфер вершин. Точность аппроксимации тела задается пользователем. Обеспечить возможность вращения и масштабирования многогранника и удаление невидимых линий и поверхностей. Реализовать простую модель освещения на GLSL. Параметры освещения и отражающие свойства материала задаются пользователем в диалоговом режиме.

**Вариант многогранника: 4. Полушарие**

**С**криншоты работы программы:

 ****

Фрагменты кода:

Функция PaintGL:

void View::*paintGL*() {

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

glEnable(GL\_DEPTH\_TEST);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

float r;

r = fig.get\_r()

*if* (r < 1.f) {

r = 1.;

}

glOrtho(-2. \* *static\_cast*<double>(r), 2. \* *static\_cast*<double>(r), -2. \* *static\_cast*<double>(r),

2. \* *static\_cast*<double>(r), -2. \* *static\_cast*<double>(r), 2. \* *static\_cast*<double>(r));

glMatrixMode(GL\_MODELVIEW);

glLoadIdentity();

glRotatef(rotateX, 1.f, 0.f, 0.f);

glRotatef(rotateY, 0.f, 1.f, 0.f);

glRotatef(rotateZ, 0.f, 0.f, 1.f);

glScalef(scale, scale, scale);

*if* (displayCarcass) {

glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_LINE);

glDisable(GL\_LIGHTING);

} *else* {

glPolygonMode(GL\_FRONT\_AND\_BACK, GL\_FILL);

glEnable(GL\_LIGHTING);

}

glPushMatrix();

glLoadIdentity();

glEnable(GL\_NORMALIZE);

glMaterialfv(GL\_FRONT\_AND\_BACK, GL\_AMBIENT, fig.get\_ambient\_color());

glMaterialfv(GL\_FRONT\_AND\_BACK, GL\_DIFFUSE, fig.get\_diffuse\_color());

glMaterialfv(GL\_FRONT\_AND\_BACK, GL\_SPECULAR, fig.get\_specular\_color());

glMaterialf(GL\_FRONT\_AND\_BACK, GL\_SHININESS, fig.get\_shininess());

float light\_ambient[] = {0.f, 0.22f, 0.51f, 1.f};

float light\_diffuse[] = {1.f, 1.f, 1.f, 1.f};

float light\_specular[] = {1.f, 1.f, 1.f, 1.f};

float light\_position[] = {lightPositionX,

lightPositionY,

lightPositionZ, 1.f};

glEnable(GL\_LIGHT0);

glLightfv(GL\_LIGHT0, GL\_DIFFUSE, light\_diffuse);

glLightfv(GL\_LIGHT0, GL\_SPECULAR, light\_specular);

glLightfv(GL\_LIGHT0, GL\_AMBIENT, light\_ambient);

glLightfv(GL\_LIGHT0, GL\_POSITION, light\_position);

glLightf(GL\_LIGHT0, GL\_SPOT\_EXPONENT, 128);

glLightf(GL\_LIGHT0, GL\_CONSTANT\_ATTENUATION, 1.f);

glPopMatrix();

glEnable(GL\_CULL\_FACE);

glCullFace(GL\_BACK);

glColor3f(1.f, 0.f, 0.f);

*for* (*auto* polygon: fig.get\_polygons()) {

glBegin(GL\_POLYGON);

*for* (*auto* vertex: polygon.vertices) {

glVertex3f(vertex.x(), vertex.y(), vertex.z());

}

glEnd();

}

glDisable(GL\_CULL\_FACE);

glDisable(GL\_LIGHT0);

glDisable(GL\_LIGHTING);

glDisable(GL\_DEPTH\_TEST);

}

**Среда разработки:** Qt Creator 4.10.1

**Вывод:** В процессе выполнения лабораторной работы научился отрисовывать, масштабировать, центрировать при изменении окна, вращать и удалять невидимые линии для отрисовки выпуклых тел для одного источника света с использованием OpenGL.