ФГБОУ ВО

Уфимский университет науки и технологии

Кафедра ТК

Отчет

по лабораторной работе №1

по дисциплине «Инженерная и компьютерная графика»

Выполнил: ст.гр. ПРО-131

Семенов Л.А.

Принял:

Котельников В.А.

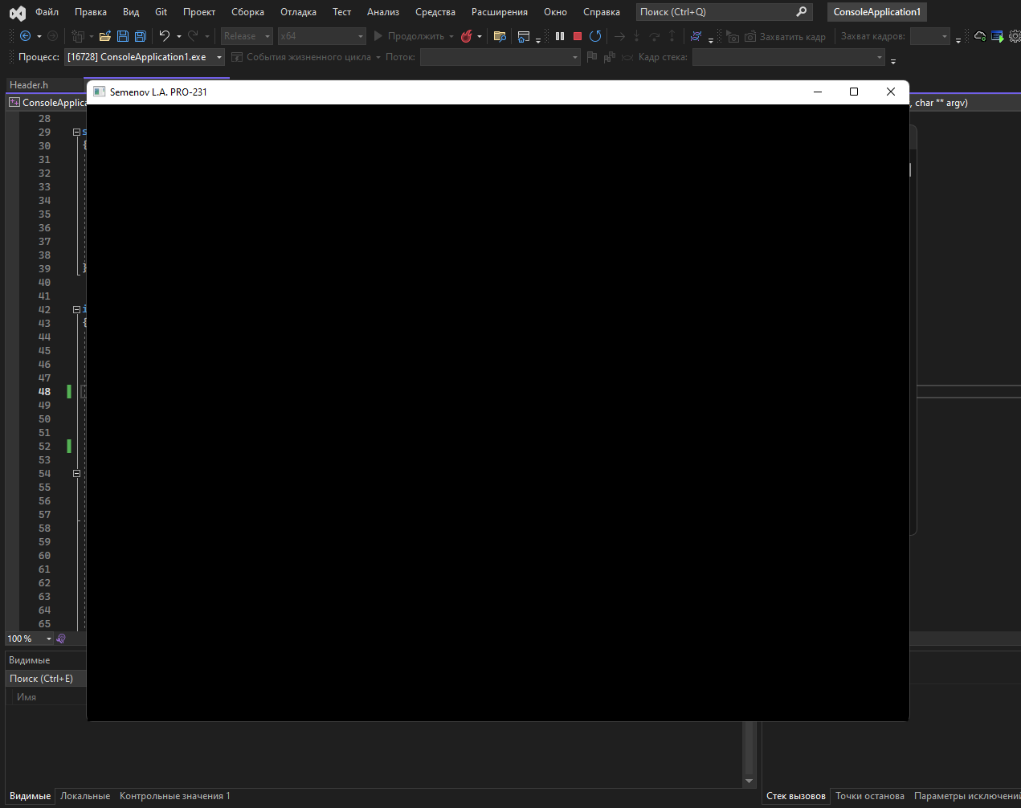
Уфа 2022 г

Цель работы: познакомиться с OpenGL

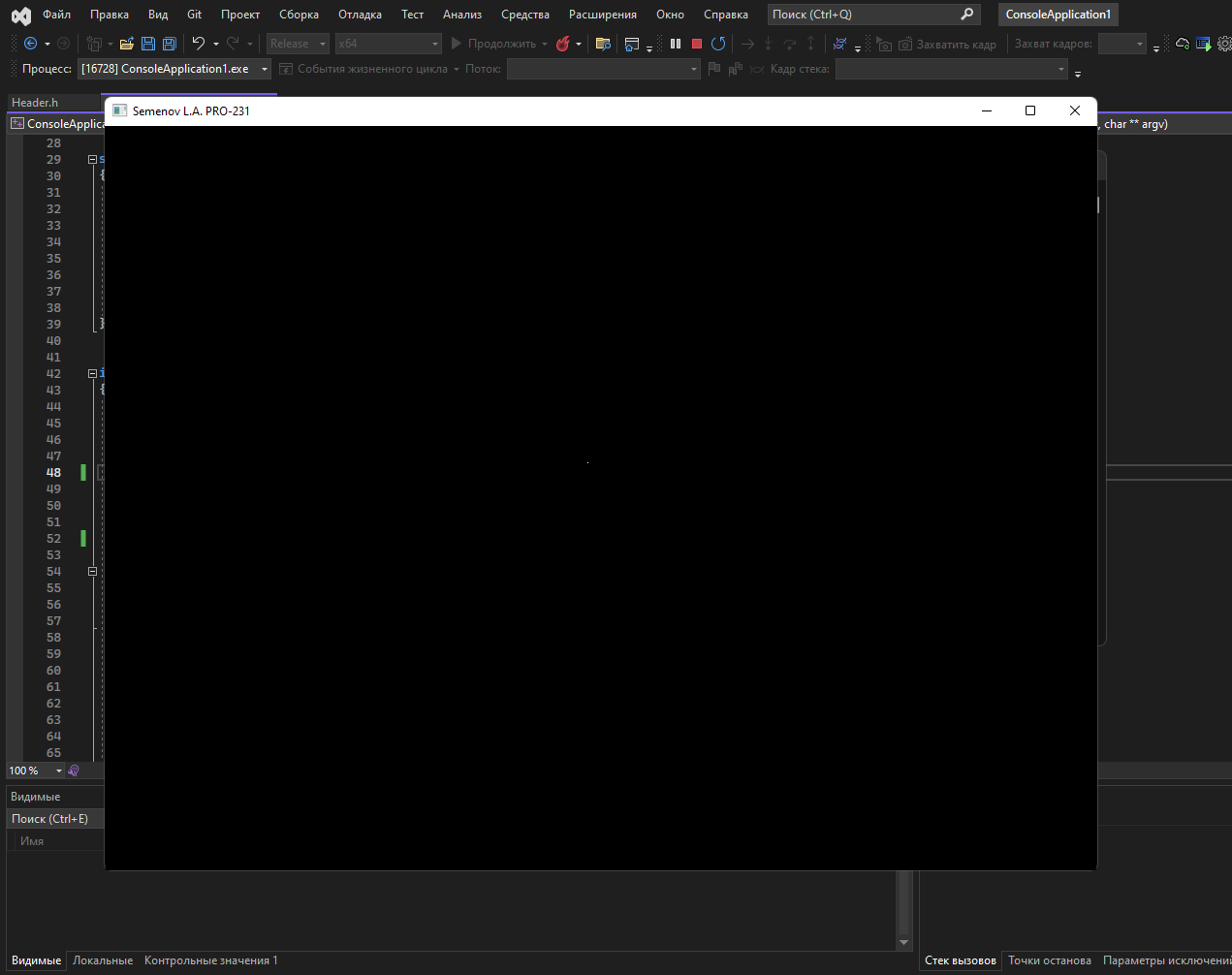
Задание: выполнить 1-3 уроки по OpenGL

Ход работы:

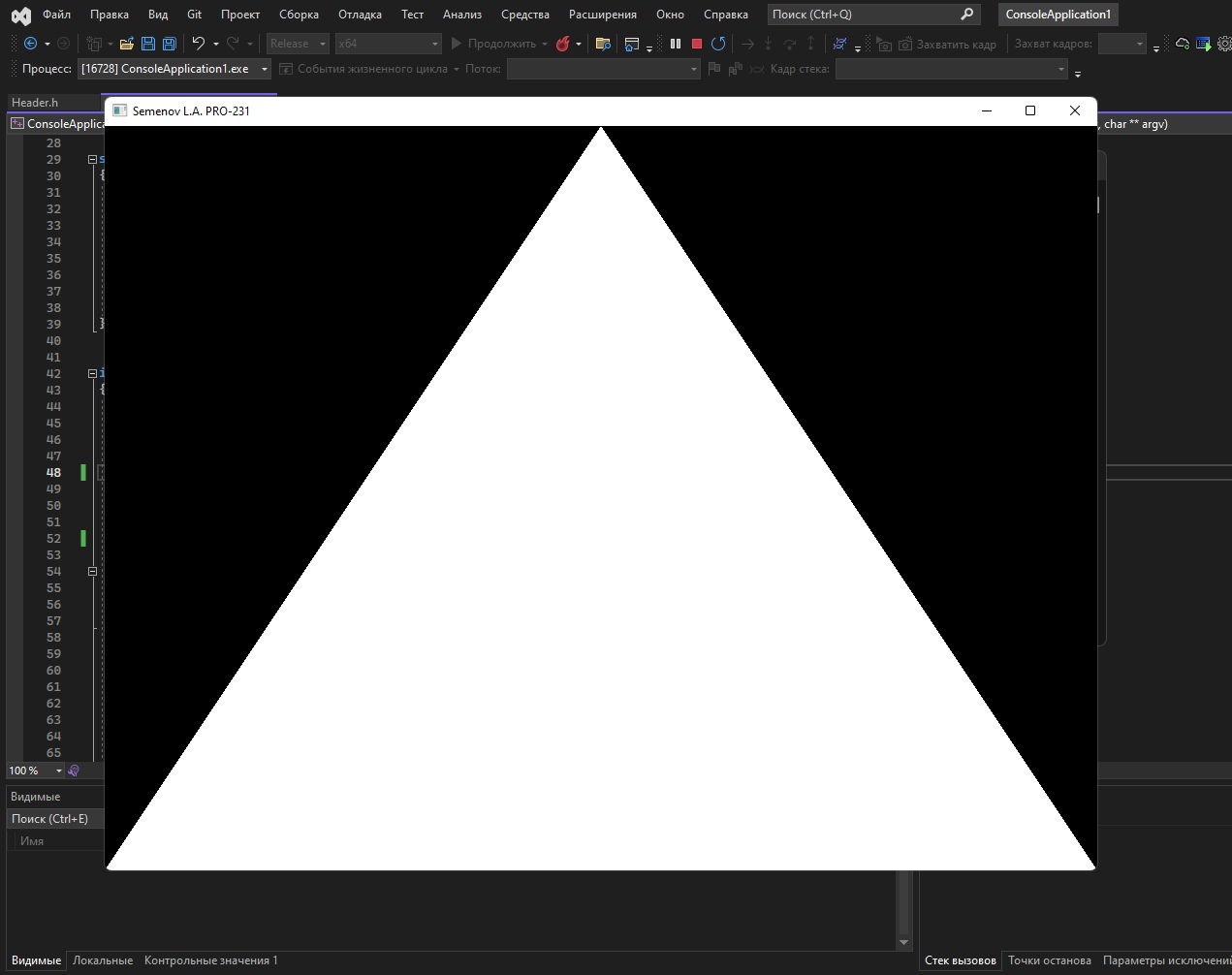
1. [Создание окна](http://ogldev.atspace.co.uk/www/tutorial01/tutorial01.html)



1. Создание точки



1. Создание треугольника.



Ссылка на GitHub:

<https://github.com/Semenovla/ingenerian_graphics/tree/main/1lab%20IiCG%20Semenov.L.A>

Приложение 1. Код программы Main.cpp

#include <stdio.h>

#include <GL/glew.h>

#include <GL/freeglut.h>

#include "Header.h"

GLuint VBO;

static void RenderSceneCB()

{

glClear(GL\_COLOR\_BUFFER\_BIT);

glEnableVertexAttribArray(0);

glBindBuffer(GL\_ARRAY\_BUFFER, VBO);

glVertexAttribPointer(0, 3, GL\_FLOAT, GL\_FALSE, 0, 0);

glDrawArrays(GL\_TRIANGLES, 0, 3);

glDisableVertexAttribArray(0);

glutSwapBuffers();

}

static void InitializeGlutCallbacks()

{

glutDisplayFunc(RenderSceneCB);

}

static void CreateVertexBuffer()

{

Vector3f Vertices[3];

Vertices[0] = Vector3f(-1.0f, -1.0f, 0.0f);

Vertices[1] = Vector3f(1.0f, -1.0f, 0.0f);

Vertices[2] = Vector3f(0.0f, 1.0f, 0.0f);

glGenBuffers(1, &VBO);

glBindBuffer(GL\_ARRAY\_BUFFER, VBO);

glBufferData(GL\_ARRAY\_BUFFER, sizeof(Vertices), Vertices, GL\_STATIC\_DRAW);

}

int main(int argc, char\*\* argv)

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_DOUBLE | GLUT\_RGBA);

glutInitWindowSize(1024, 768);

glutInitWindowPosition(100, 100);

glutCreateWindow("Semenov L.A. PRO-231");

InitializeGlutCallbacks();

GLenum res = glewInit();

if (res != GLEW\_OK) {

fprintf(stderr, "Error: '%s'\n", glewGetErrorString(res));

return 1;

}

glClearColor(0.0f, 0.0f, 0.0f, 0.0f);

CreateVertexBuffer();

glutMainLoop();

return 0;

}

Приложение 2. Header.h

#ifndef MATH\_3D\_H

#define MATH\_3D\_H

struct Vector3f

{

float x;

float y;

float z;

Vector3f()

{

}

Vector3f(float \_x, float \_y, float \_z)

{

x = \_x;

y = \_y;

z = \_z;

}

};

#endif