



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ \_\_\_\_\_ «Информатика, искусственный интеллект и системы управления»

КАФЕДРА \_\_\_\_\_ «Прикладная математика и информатика»

## Лабораторная работа № 3

### по курсу «Алгоритмы компьютерной графики»

Студент группы ИУ9-41Б Горбунов А. Д.

Преподаватель Цапкович П. А.

Москва 2024

# 1 Цель

Целью работы является знакомство с библиотекой OpenGL, принципами разработки алгоритмов компьютерной графики и их реализацией на языке C++.

## 2 правильная призма

Для выполнения лабораторной нужно будет взять код, который вы получили после лабораторной 2, отключить проективные преобразования и заменить куб на фигуру указанную в варианте.

\*наклонный эллиптический цилиндр

### 3 Практическая реализация

```
#include <GLFW/glfw3.h>
```

```
#include <cmath>
```

```
using std::cos, std::sin;
```

```
int mode = 0;
```

```
float degree_y = 0.0;
```

```
float degree_x = 0.0;
```

```
float move_y = 0.0;
```

```
float move_x = 0.0;
```

```
float osnov_x = 0.1;
```

```
float osnov_y = 0.0;
```

```
void key_callback(GLFWwindow *window, int key, int scancode, int action, int mods)
{
```

```
    if (action == GLFW_PRESS || action == GLFW_REPEAT)
```

```
    {
```

```
        if (key == GLFW_KEY_ESCAPE) { glfwSetWindowShouldClose(window,
```

```
        else if (key == GLFW_KEY_UP) { degree_y += 0.2; }
```

```
        else if (key == GLFW_KEY_DOWN) { degree_y -= 0.2; }
```

```
        else if (key == GLFW_KEY_LEFT) { degree_x += 0.2; }
```

```
        else if (key == GLFW_KEY_RIGHT){ degree_x -= 0.2; }
```

```
        else if (key == GLFW_KEY_D) { move_x += 0.2; }
```

```
        else if (key == GLFW_KEY_A) { move_x -= 0.2; }
```

```
        else if (key == GLFW_KEY_W) { move_y += 0.2; }
```

```
        else if (key == GLFW_KEY_S) { move_y -= 0.2; }
```

```
        else if (key == GLFW_KEY_L) { osnov_x += 0.1; }
```

```
        else if (key == GLFW_KEY_K) { osnov_x -= 0.1; }
```

```
        else if (key == GLFW_KEY_I) { osnov_y += 0.1; }
```

```
        else if (key == GLFW_KEY_O) { osnov_y -= 0.1; }
```

```
        else if (key == GLFW_KEY_SPACE){ mode = (mode + 1) % 2; }
```

```
}  
}
```

```
void display(GLFWwindow* window)  
{  
    glEnable(GL_DEPTH_TEST);  
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);  
  
    if (mode == 0) {  
        glPolygonMode(GL_FRONT_AND_BACK, GL_LINE);  
    } else {  
        glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);  
    }  
    glPushMatrix();  
  
    glTranslatef(0.0f + move_x, 0.0f + move_y, 0.0f);  
    glRotatef(degree_y * 50.f, 1.f, 0.f, 0.f);  
    glRotatef(degree_x * 50.f, 0.f, 1.f, 0.f);  
  
    glBegin(GL_QUAD_STRIP);  
    glColor3f(0.4f, 0.4f, 1.0f);  
    for (int i = 0; i <= 360; i += 10)  
    {  
        float angle = i * M_PI / 180 ;  
        glVertex3f(0.5 * cos(angle) + osnov_x, 0.25 * sin(angle) + osnov_y, 0.0);  
        glVertex3f(0.5 * cos(angle), 0.25 * sin(angle), 0.5);  
    }  
    glEnd();  
  
    glBegin(GL_POLYGON);
```

```

glColor3f(1.0f, 0.3f, 0.3f);
for (int i = 0; i <= 360; i++)
{
    float angle = i * M_PI / 180;
    glVertex3f(0.5 * cos(angle) + osnov_x, 0.25 * sin(angle) + osnov_y, 0.0);
}
glEnd();

glBegin(GL_POLYGON);
glColor3f(0.5f, 0.7f, 0.7f);
for (int i = 0; i <= 360; i++)
{
    float angle = i * M_PI / 180;
    glVertex3f(0.5 * cos(angle), 0.25 * sin(angle), 0.5);
}
glEnd();

glPopMatrix();
}

int main()
{
    if (!glfwInit()) {
        return -1;
    }
    GLFWwindow* window = glfwCreateWindow(1000, 1000, "Lab 3", NULL, NULL);
    if (!window) {
        glfwTerminate();
        return -1;
    }

    glfwMakeContextCurrent(window);

```

```
glfwSetKeyCallback(window, key_callback);

while (!glfwWindowShouldClose(window))
{
    display(window);

    glfwSwapBuffers(window);
    glfwPollEvents();
}

glfwTerminate();

return 0;
}
```

## 4 Вывод

В данной работе я изучил возможности языка с++ и библиотеки OpenGL, приобрёл навыки разработки на языке с++ алгоритмов компьютерной графики, углубил свои знания в программировании и изучил подробнее устройство компьютерной графики.

## 5 Результат запуска

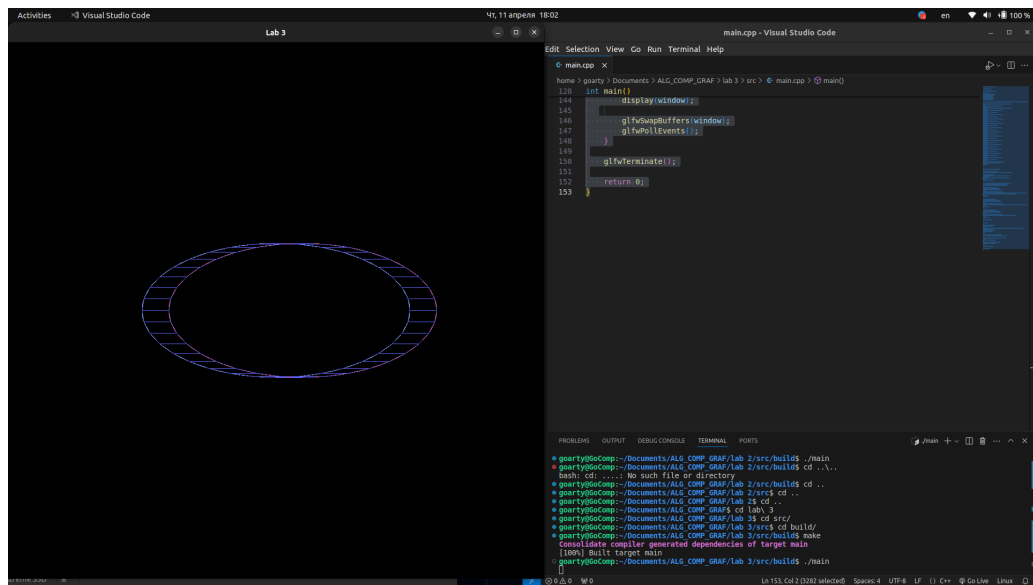


Рис. 1 — Без воздействия

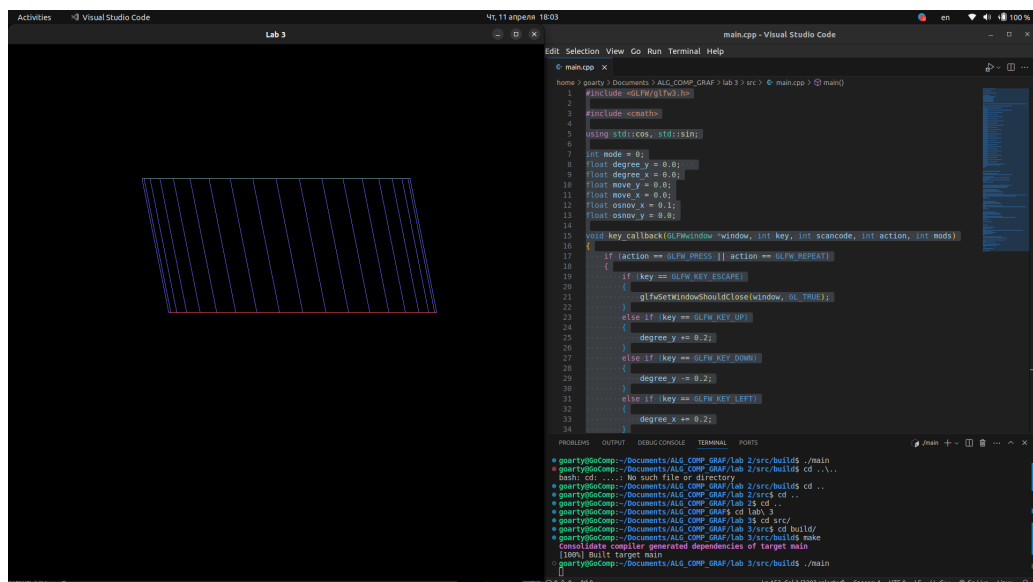


Рис. 2 — Повёрнутый на 90 градусов

