Московский авиационный институт

(национальный исследовательский университет)

Факультет информационных технологий и прикладной математики

Кафедра вычислительной математики и программирования

Лабораторная работа №7 по курсу «Компьютерная графика»

Студент: М. А. Бронников

Преподаватель: Г. С. Филиппов

Группа: М8О-307Б

Дата:

Оценка:

Подпись:

Москва, 2019

Построение плоских полиномиальных кривыхЗадача: для поверхности, созданной в л.р. №5, обеспечить выполнение следующего шейдерного эффекта.

Задание: Написать программу, строящую полиномиальную кривую по заданным точкам. Обеспечить возможность изменения позиции точек и, при необходимости, значений касательных векторов и натяжения.

Вариант №12:

12. B-сплайн. n = 6, k = 4. Узловой вектор равномерный.

1. Описание

Для реализации я использовал библиотеки GLFW для работы с оконными приложениями, GLM для математических операций, GLEW для упрощения работы с функциями библиотек и кросслатформенности, а также язык C++.

OpenGL (Open Graphics Library) — спецификация, определяющая платформонезависимый (независимый от языка программирования) программный интерфейс для написания приложений, использующих двумерную и трёхмерную компьютерную графику. Включает более 300 функций для рисования сложных трёхмерных сцен из простых примитивов.

2. Исходный код

библиотек: GLFW, GLM, GLEW.

1 #include <iostream>

2 #define GLEW\_STATIC

3 #include <GL/glew.h>

4 #include <GLFW/glfw3.h>

5 #include <glm/glm.hpp>

6 #include <glm/gtc/matrix\_transform.hpp>

7 #include <glm/gtc/type\_ptr.hpp>

8 #include "multyplyes.h"

9 #include "Camera.h"

1 const GLuint WIDTH = 800, HEIGHT = 600;

2 Camera camera(glm::vec3(0.0f, 0.0f, 3.0f));

3 GLfloat lastX = WIDTH / 2.0;

4 GLfloat lastY = HEIGHT / 2.0;

5 bool keys[1024];

6 bool ret = true;

7 GLfloat APPROX = 0.01f;

8 GLfloat chill\_height = 0.8f;

9 glm::vec3 lightPos(1.2f, 1.0f, 2.0f);

10 GLfloat deltaTime = 0.0f; // Time between current frame and last frame

11 GLfloat lastFrame = 0.0f; // Time of last frame

12 GLfloat x\_rotation = 0.0f, y\_rotation = 0.0f, z\_rotation = 0.0f;

13 GLfloat scale = 1.0f;

Функции реализующие в себе обработку пользовательских действий, таких как об-

работка изменений экрана, нажатия клавиш и движения мыши, имеют следующие

объявления:

1 void key\_callback(GLFWwindow\* window, int key, int scancode, int action, int mode);

2 void mouse\_callback(GLFWwindow\* window, double xpos, double ypos);

3 void scroll\_callback(GLFWwindow\* window, double xoffset, double yoffset);

4 void do\_movement();

5 void new\_func\_size\_callback(GLFWwindow\* window, int width, int heigh);

3. Консоль

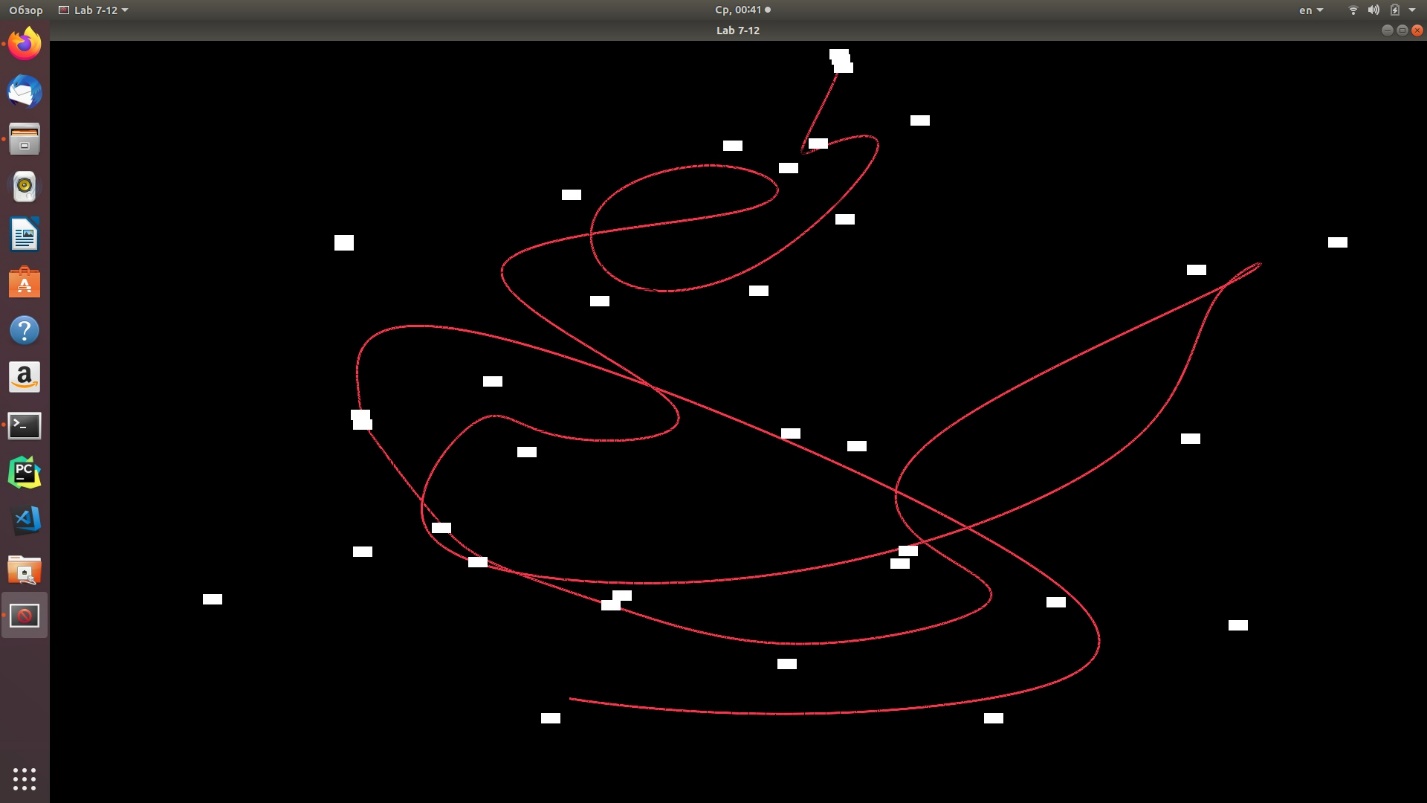
(base) romka@romka-Lenovo-ideapad-510-15IKB:~/KG/lab7$ g++ main.cpp -o hello -lGL -lGLEW -lglfw  
(base) romka@romka-Lenovo-ideapad-510-15IKB:~/KG/lab7$ ./hello  
Added point: 0  
Added point: 1  
Added point: 2  
Added point: 3  
Added point: 4  
Added point: 5  
Added point: 6  
Added point: 7  
Added point: 8  
Added point: 9  
Added point: 10  
Added point: 11  
Added point: 12  
Selected point: 12  
Added point: 13  
Added point: 14  
Added point: 15  
Added point: 16  
Added point: 17  
Added point: 18  
Added point: 19  
Added point: 20  
Added point: 21  
Added point: 22  
Added point: 23  
Added point: 24  
Added point: 25  
Added point: 26  
Added point: 27  
Added point: 28  
Added point: 29  
Selected point: 29  
Selected point: 11  
Selected point: 11  
Added point: 30  
Selected point: 7  
Selected point: 30  
Selected point: 21  
Selected point: 20  
Selected point: 21  
Added point: 31  
Added point: 32  
Added point: 33  
Added point: 34

После откроется изображение фигуры в окне.

Это окно можно изменять по размерам и перемещать по экрану без всяких побочных

эффектов, график подстраивается под изменение размеров экрана и масштабируется

соответствующим образом.



4. Выводы

Выполнив 7 лабораторную работу по курсу «Компьютерная графика», я получил представление о том как строить полиномиальные кривые, потренировался в работе с OpenGL, и

попрактиковался в этом сам на языке C++. Также я научился создавать объемные

фигуры.

В процессе решения задачи мне пришлось изучить много материала на эту тему.

Наиболее трудным для меня было вникнуть в математическую сторону этого вопроса.

Данная лабораторная работа показала мне важность математики в программировании, а также дала новые практические навыки, которые я непременно буду использовать в будущем.