Homework 8 - Bezier Curve

陈曦 16340036

Basic:

1. 用户能通过左键点击添加Bezier曲线的控制点，右键点击则对当前添加的最后一个控制点进行消除

2. 工具根据鼠标绘制的控制点实时更新Bezier曲线。

Hint: 大家可查询捕捉mouse移动和点击的函数方法

通过鼠标添加消除控制点：

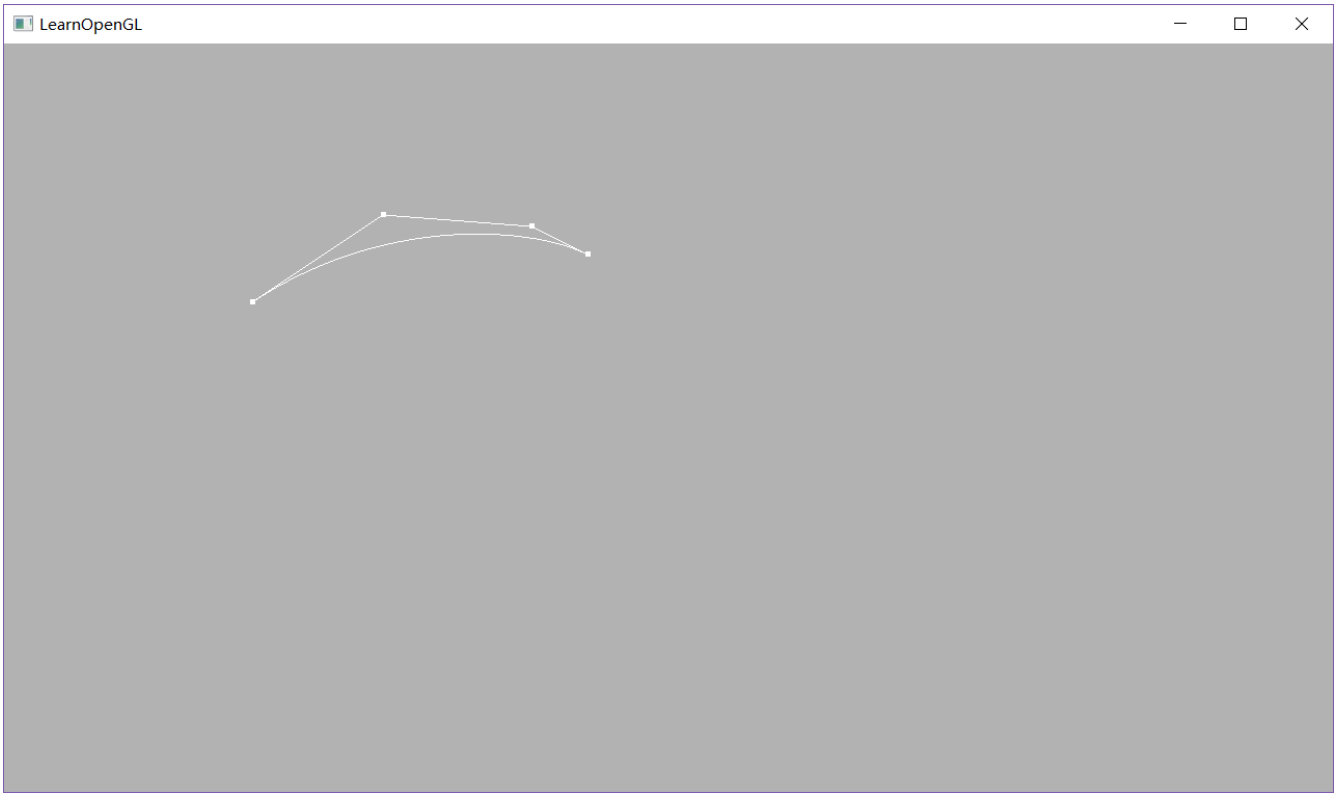
```
glfwSetMouseButtonCallback(window, mouse_button_callback);

void mouse_button_callback(GLFWwindow* window, int button, int action, int mods) {
    if (action == GLFW_PRESS) {
        switch (button) {
        case GLFW_MOUSE_BUTTON_LEFT:
            double xpos, ypos;
            glfwGetCursorPos(window, &xpos, &ypos); //width和height鼠标的实时坐标。
            vertices[point_num * 2] = (float)xpos / (float)(WIDTH) * 2.0f - 1.0f; //横坐标
            vertices[point_num * 2 + 1] = -((float)ypos / (float)(HEIGHT) * 2.0f - 1.0f); //
纵坐标
            point_num++;
            break;
        case GLFW_MOUSE_BUTTON_RIGHT:
            if (point_num > 0) {
                point_num--;
            }
            break;
        default:
            break;
        }
    }
}
```

绘制Bezier曲线：

```
        //画点
        glPointSize(5.0f);
        glDrawArrays(GL_POINTS, 0, point_num);
        //画线
        glDrawArrays(GL_LINE_STRIP, 0, point_num);
        //画BezierCurve
        q_vertices[0] = vertices[0];
        q_vertices[1] = vertices[1];

        for (float t = 0.0f; t < 1.0f; t += 0.02f) {
            double p1 = pow((1 - t), point_num - 1), p2 = 1;
            for (int i = 0; i < point_num; i++) {
```

```
            q_vertices[2] += vertices[i * 2] * C(point_num - 1, i) * p1 * p2;
            q_vertices[3] += vertices[i * 2 + 1] * C(point_num - 1, i) * p1 * p2;
            p1 /= 1 - t;
            p2 *= t;
        }
        unsigned int qVAO, qVBO;
        glGenVertexArrays(1, &qVAO);
        glGenBuffers(1, &qVBO);//使用glGenBuffers函数和一个缓冲ID生成一个VBO对象:

        glBindVertexArray(qVAO);

        glBindBuffer(GL_ARRAY_BUFFER, qVBO);//把新创建的缓冲绑定到GL_ARRAY_BUFFER目标上
        glBufferData(GL_ARRAY_BUFFER, sizeof(q_vertices), q_vertices, GL_STATIC_DRAW);//
把之前定义的顶点数据复制到缓冲的内存中

                                                                                //解析定
点数据:位置属性
        glVertexAttribPointer(0, 2, GL_FLOAT, GL_FALSE, 2 * sizeof(float), (void*)0);
        glEnableVertexAttribArray(0);


        glBindBuffer(GL_ARRAY_BUFFER, 0);
        glBindVertexArray(qVAO);

        if (point_num > 0) {
            glBindVertexArray(qVAO);
            glDrawArrays(GL_LINE_STRIP, 0, 2);
        }

        glDeleteVertexArrays(1, &qVAO);
        glDeleteBuffers(1, &qVBO);

        q_vertices[0] = q_vertices[2];
        q_vertices[1] = q_vertices[3];
        q_vertices[2] = 0.0f;
        q_vertices[3] = 0.0f;
    }
```

截图如下，具体实现效果见gif图:


LearnOpenGL