

第四次作业说明文档

16337327 郑映雪

本次作业分两部分。

第一部分是绘制三次贝塞尔曲线。贝塞尔曲线绘制的相关画法网上可以找到实例参考，公式也可以在网上查到。这次作业与之前多出的部分是贝塞尔曲线的绘制以及鼠标的响应。贝塞尔曲线的绘制。

首先是贝塞尔曲线的公式：

$$b(t) = P_0(1-t)^3 + 3P_1t(1-t)^2 + 3P_2t^2(1-t) + P_3t^3$$
$$t \in [0,1]$$

我们可以采用无线分割的形式，将 t 增加的步长设为一个极小的数，即可模拟出曲线。

对于鼠标响应，有特殊的函数 `glutMainLoop`。在对鼠标进行操作后，每次点击鼠标根据目前的状态判断应该进行哪一个操作。若点不足 4 个就画点，点画完后就画直线，直线画完后就画贝塞尔曲线，都画完就清除缓存，以便进行下一步的绘制。在这次作业里需要注意的是默认的 x 轴 y 轴与我们平常的平面直角坐标系不同，它是与显示屏的方向一样的，（之前在操作系统作业里我也碰到过这个壁哈哈），所以要注意坐标变换。

第二部分是进行文件的读取操作。这一部分我主要的参考来源是 https://blog.csdn.net/qg_28057541/article/details/51362945。这篇博客对文件的读取和其他相关的处理有详细的解答。用 `vcpkg` 安装 `openmesh` 是很快的，也不需要太多繁琐的配置。数据绘制主要是靠

面、边、顶点来遍历。在读取文件部分，首先请求顶点法线，用 `read_mesh` 读取文件，如果不存在顶点法线则要计算。在显示上面，开启深度缓冲区的功能绘制线和面。利用无数个三角形拼接成平滑的面。鼠标和键盘的交互可以使用专门的函数（在作业一中有提及）。除了上面的新功能外，这次还有了一个新函数——`glrotate`，可以实现物体的旋转。但是在双缓存、深度缓存等概念上，我还不太清楚，目前只知道要这么做，但不知道为什么。我想，我之后得花点时间好好深入一下这些概念的问题。