**ClassAssignment1**

컴퓨터소프트웨어학부

2018008877 이상원

이번 assignment의 목표는 perspective projection / orthogonal projection 에서 mouse의 동작에 따라 카메라를 조작하는 것이다. 구현해야 할 목표는 총 3가지이다.

1. 마우스에 동작에 따라 카메라를 움직이는 것
2. 이를 구현하기 위해서는 먼저 카메라에 보이는 물체가 있어야 한다. 나는 정육면체를 그렸고, 기준을 표현하기 위해서 x, y, z 축도 그렸다. 각각 drawCube(X) 함수와 drawAxis(X)에서 구현되었다. X는 물체의 크기를 설정하는 것이다.
3. render() 함수를 구현해야 했다. render() 함수 내부에서는 glMultMatrixf() 의 인자로 getMyLookat()의 반환 matrix를 넣어주었다.
4. getMyLookat() 함수 내부에서는 현재 ref (target)의 좌표와 elevation과 azimuth에 따라 eye의 좌표를 정한 후 eye의 w, u, v vector를 계산했고, 이를 토대로 Mv vector를 만들어서 return한다.
5. Azimuth와 elevation은 button\_callback을 통해 왼쪽 클릭을 했을 때, cursor\_callback 함수에 의해 정해진다. 기존 mouse의 좌표와 움직이는 mouse의 위치에 따라서 계속해서 azimuth와 elevation이 정해진다.
6. Panning을 위한 translation은 button\_callback을 통해 오른쪽 클릭을 했을 때, cursor\_callback 함수에 의해 정해진다. 이 역시 기존 mouse 좌표와 변화하는 mouse의 위치에 따라서 정해진다.
7. Zooming은 scroll\_callback 함수에 의해 정해진다.
8. V 키를 눌렀을 때 projection 모드 변경
9. key\_callback 함수를 통해 v키가 눌렸을 때, orthoMode라는 global 변수를 true or false로 toggle 되도록 만들었다.
10. 최초 프로그램 실행때는 orthoMode가 False이고, v키를 누르면 toggle 된다. orthoMode가 True일 때는 glOrtho() 함수가, False일 때는 gluPerspective() 함수가 실행된다.
11. rectangular grid 그리기
12. drawGrid(X) 함수에서 grid를 그리도록 했다. Global 변수인 GRIDFACTOR에 따라서 얼마나 넓게 그릴지 정할 수 있다.

* **perspective projection**

|  |  |  |  |
| --- | --- | --- | --- |
| Orbit |  |  | 텍스트, 그물, 어두운이(가) 표시된 사진  자동 생성된 설명 |
| Pan |  |  |  |
| zoom |  |  |  |

* **orthogonal projection**

|  |  |  |  |
| --- | --- | --- | --- |
| Orbit |  |  |  |
| Pan |  |  |  |