**컴퓨터그래픽스 Class assignment 1: Basic OpenGL viewer**

컴퓨터소프트웨어학부 2018007974 김서연

**A. 구현된 요구 조건**

① 마우스 움직임에 따른 카메라 조작

* Orbit: 마우스 왼쪽 버튼을 누른 채로 드래그하면 물체는 가만히 있고 드래그하는 방향에 따라 카메라만 움직이도록 했습니다. 마우스 커서의 x 좌표와 y 좌표를 받아와서 각 좌표가 이동한 만큼 azimuth와 elevation을 바꿔줌으로써 구현했습니다.
* Panning: 마우스 오른쪽 버튼을 누른 채로 드래그하면 물체와 카메라가 같이 움직이도록 했습니다. 마우스 커서의 x좌표와 y좌표가 이동한 만큼 translate했는데, 이 때 그냥 translate하면 orbit으로 화면이 돌아간 후 panning하는 경우에 돌아간 화면이 아닌 맨 처음의 화면을 기준으로 움직이기 때문에 원하는 대로 동작하지 않았습니다. 따라서 먼저 rotation 행렬을 만들어 행렬 곱을 해준 후에 translate되도록 구현했습니다.
* Zooming: 마우스 휠을 올리면 확대, 내리면 축소가 되도록 했습니다. 휠의 값이 바뀌는 만큼 distancs를 늘리거나 줄임으로써 구현했습니다. 일정 정도 이상으로는 확대/축소가 되지 않도록 했습니다.

② ‘v’ 키로 프로젝션 토글

* 전역변수 하나를 두어 v키를 누르면 perspective projection 모드와 orthogonal projection 모드를 왔다갔다 할 수 있도록 구현했습니다. Perspective projection 모드일 때는 gluPerspective()를, orthogonal projection 모드일 때는 glOrtho()를 호출한 후 gluLookAt()을 호출하는 과정을 거쳐 원하는 대로 화면이 보이도록 했습니다.

③ xz평면의 그리드

* 그리드를 그리는 drawGrid()라는 함수를 정의해서 사용했습니다. glColor3ub()를 호출해서 원하는 색으로 설정한 후 np.arange()를 이용해 그리드의 간격을 저장하는 변수를 만들어 변수 안의 값들을 이용해 반복적으로 glVertex3fv()를 호출하여 선을 그렸습니다.

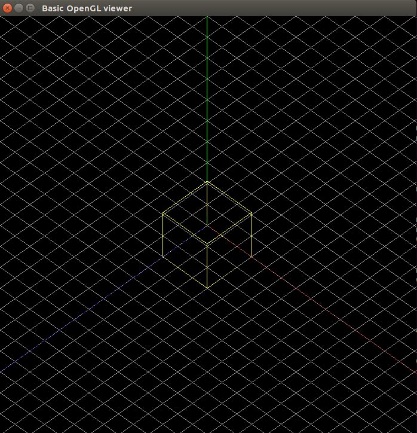
④ 그 외 요구사항

* 창의 크기는 700\*700으로, 타이틀은 ‘Basic OpenGL viewer’로 설정했습니다.
* 구현된 사항을 쉽게 확인하기 위해 양의 방향으로 길이가 30인 x, y, z축을 그리는 drawFrame()을 정의해서 사용했고, 각 변의 길이가 1.5인 정육면체를 그려두었습니다.
* .py 파일은 하나만 구현했습니다. main.py 파일만 컴파일하면 프로그램이 실행되도록 했습니다.

**B. 프로그램 스크린샷** (perspective/orthogonal)

① 처음 실행했을 때

트릭, 경사로, 하는중, 밤이(가) 표시된 사진

자동 생성된 설명 

② Orbit

검은색, 창살이(가) 표시된 사진

자동 생성된 설명 텍스트이(가) 표시된 사진

자동 생성된 설명

③ Orbit 후 Panning

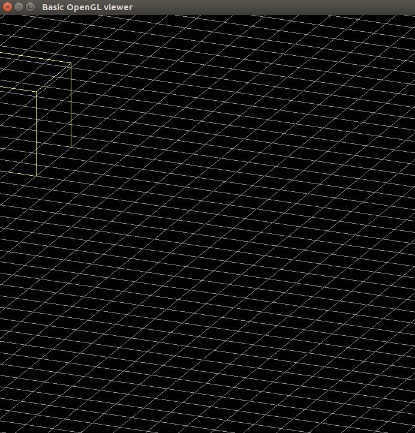
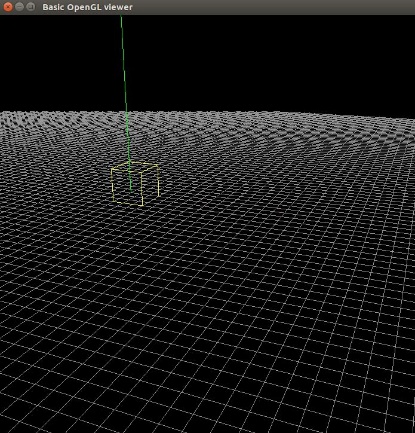
텍스트, 검은색, 창살이(가) 표시된 사진

자동 생성된 설명 실외, 바둑판식이(가) 표시된 사진

자동 생성된 설명

④ Zooming

텍스트, 검은색이(가) 표시된 사진

자동 생성된 설명  

(세 번째 사진은 perspective projection 모드에서 축소한 것입니다.)