**Computer Graphics Assignment 1**

**2015005205**

**최홍규**

1. **실행방법**

python3에서 실행되도록 하였다. python3 assignment1.py을 입력하여 실행한다.

파이썬 버전 : Python 3.7.1

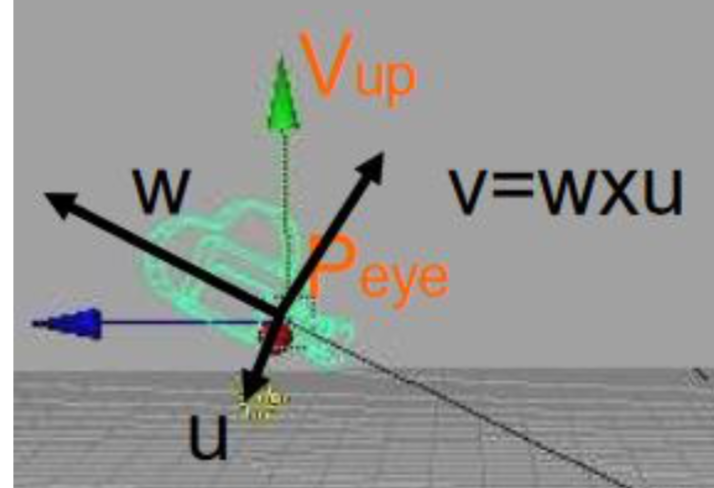
왼쪽,오른쪽 마우스 클릭 드래그와 마우스 스크롤을 통해 동작한다.

1. **구현사항**
   1. **마우스 스크롤을 통해 Zooming**

gluPerspective(fov, 1,.001,1000)

gluPerspective함수의 첫번째 인자값 fov를 통해 구현하였다. 인자의 값이 커지면 물체와 멀어지고 값이 줄어들면 물체와 가까워진다.

* 1. **마우스 오른쪽 버튼을 통해 Panning**



w=(M-at)/np.sqrt(np.dot(M-at,M-at))

u=np.cross(up,w)/np.sqrt(np.dot(np.cross(up,w),np.cross(up,w)))

v=np.cross(w,u)

a+=u\*(xpos-nowx)\*0.005+v\*(nowy-ypos)\*0.005

glTranslatef(a[0],a[1],a[2])

카메라를 기준으로 u,v 벡터를 계산하여 두 벡터의 합으로 glTranslatef함수를 사용하여 Panning을 구현하였다.

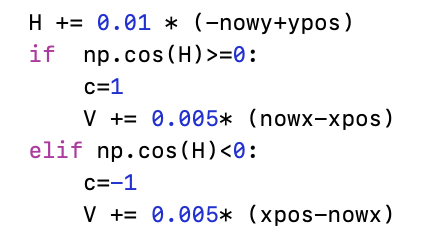
* 1. **마우스 왼쪽 버튼을 통해 Orbit**

M=np.array([ 10\*np.cos(H)\*np.sin(V),10\*np.sin(H),10\*np.cos(H)\*np.cos(V)])

up=np.array([0,c,0])

at=np.array([0,0,0])

gluLookAt(M[0],M[1],M[2],at[0],at[1],at[2],up[0],up[1],up[2])



M의 식을 통해 카메라의 위치를 정했다.

cos(H)의 값이 양수일 때 up벡터의 y의 값을 1, 음수일 때 up벡터의 y의 값을 -1로 지정하여 물체를 위 아래로 360도 돌 수 있게 하였다.

* 1. **뛰는 모습의 사람(몸통을 기준으로 4계층을 구현)**

몸통을 기준으로 첫번째 다리가 앞뒤로 움직인다.

두번째 다리가 첫번째 다리와 같이 움직이면서 독자적으로 앞뒤로 움직인다. 그 밑의 발이 첫번째와 두번째 다리의 움직임대로 움직이면서 독자적으로 앞뒤로 움직인다.

팔의 경우 첫번째 팔이 앞뒤로 움직이고 두번째 팔이 첫번째 팔 움직임대로 움직이면서 독자적으로 앞뒤로 움직인다.

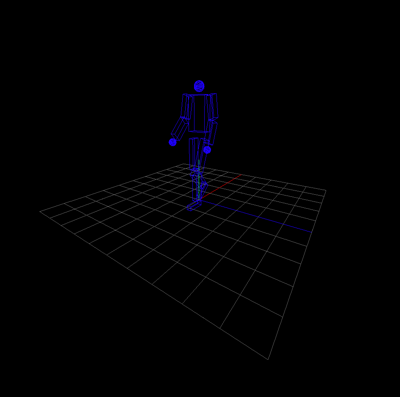
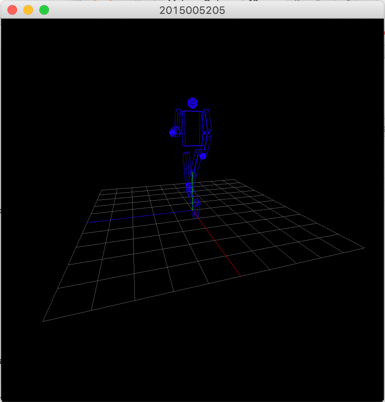
Running함수를 통해 사람을 그릴 수 있게 하였다.

* 1. **others**

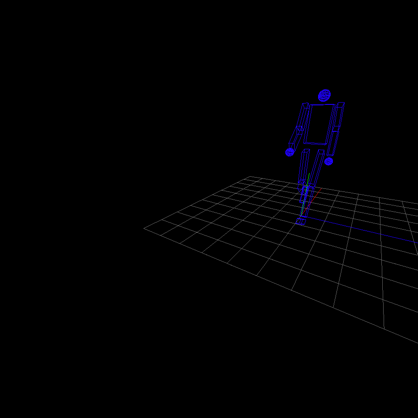
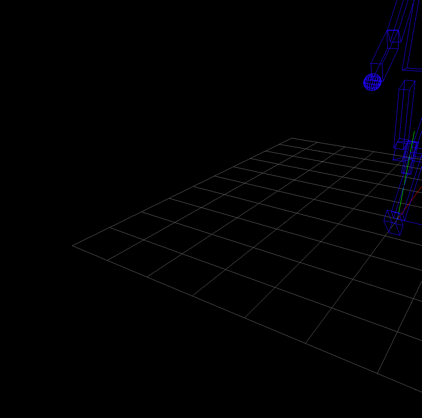
xz 평면에 격자를 만들었다. 또한 물체의 x,y,z축을 색으로 구분하여 나타내었다.

1. **실행화면**

**1)초기화면 2)Orbit 한 화면**

****

**3)Panning한 화면 4)Zooming한 화면**

****