**기초 컴퓨터 그래픽스**

**HW3 README**

학번 20191569 이름 김서연

**1. [환경 명세]**

1) 본인 프로그램의 실제 구동 환경을 명시 할 것 (OS, CPU, GPU, Compiler 등등)

window10 64bit, i5-8250U CPU, visual studio 2019 – win 32, debug

**2. [요구사항]**

   1. Modeling Transformation

1) 먼저 가상의 3차원 세상의 바닥과 좌표의 기준이 되는 세상 좌표계를 그려라 (최대 10점)

- 확인 방법:

Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 0,1,2,3,4 중 하나의 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. World view처럼 넓게 한 눈에 확인할 수 있는 카메라는 1번 카메라 즉 ‘0’ 숫자키와 연결된 카메라이므로 바닥과 좌표를 확인하려면 0을 누르는 것이 제일 한 눈에 확인 가능하다.

2) 최대 5개까지의 서로 다른 정적인 물체를 서로 다른 모델링 변환을 사용하여 가상의 세상에 배치하라 (물체 당 5점 최대 25점)

a) 사용한 물체: 소

- 부여한 **서로 다른** 모델링 변환 : S(10,10,10)을 통해 사이즈를 키우고, (0,0,1)을 축으로 90만큼 회전해서 오른쪽을 보던 소를 위를 보게 했다. 이후 (0,1,0)을 축으로 180도 회전해서 소를 좌우반전했고 T(7,0,0)을 통해 알맞은 자리로 이동시켰다.

- 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 0,1,2,3,4 중 하나의 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. World view처럼 넓게 한 눈에 확인할 수 있는 카메라는 1번 카메라 즉 ‘0’ 숫자키와 연결된 카메라이므로 바닥과 좌표를 확인하려면 0을 누르는 것이 제일 한 눈에 확인 가능하다.

b) 사용한 물체: 상자

- 부여한 **서로 다른** 모델링 변환: s(3,3,3)을 통해 사이즈를 키우고 (0,1,0)을 축으로 45도 만큼 회전했다. 이후 T(-13,5,13)을 통해 알맞은 자리로 이동시켰다.

- 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 0,1,2,3,4 중 하나의 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. World view처럼 넓게 한 눈에 확인할 수 있는 카메라는 1번 카메라 즉 ‘0’ 숫자키와 연결된 카메라이므로 바닥과 좌표를 확인하려면 0을 누르는 것이 제일 한 눈에 확인 가능하다.

c) 사용한 물체: 호랑이

- 부여한 **서로 다른** 모델링 변환: s(0.7,0.7,0.7)을 통해 사이즈를 줄이고 (1,0,0)을 축으로 -90도 만큼 회전했다. 즉 바닥을 보고 있던 호랑이를 앞을 보도록 돌렸다. 이후 T(-13,8,13)을 통해 알맞은 자리로 이동시켰다. 즉 상자 위로 호랑이를 올렸다.

- 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 0,1,2,3,4 중 하나의 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. World view처럼 넓게 한 눈에 확인할 수 있는 카메라는 1번 카메라 즉 ‘0’ 숫자키와 연결된 카메라이므로 바닥과 좌표를 확인하려면 0을 누르는 것이 제일 한 눈에 확인 가능하다.

d) 사용한 물체 : 주전자

- 부여한 **서로 다른** 모델링 변환: (2,2,2)을 통해 사이즈를 키우고, (1,0,0)을 축으로 -90만큼 회전해서 정면을 바라보던 주전자의 윗쪽 면이 위를 볼 수 있도록 했다. 이후 (0,0,1)을 축으로 -45도 회전해서 주전자를 살짝 아래로 기울였고 T(0,10,0)을 통해 공중으로 띄웠다.

- 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 0,1,2,3,4 중 하나의 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. World view처럼 넓게 한 눈에 확인할 수 있는 카메라는 1번 카메라 즉 ‘0’ 숫자키와 연결된 카메라이므로 바닥과 좌표를 확인하려면 0을 누르는 것이 제일 한 눈에 확인 가능하다.

e) 사용한 물체: 자동차

- 부여한 **서로 다른** 모델링 변환: (0.5,0.5,0.5)을 통해 사이즈를 줄이고, (0,1,0)을 축으로 45도만큼 회전해서 왼쪽을 바라보던 자동차를 비스듬히 왼쪽 정면을 바라볼 수 있도록 했다. T(15,3,15)을 통해 알맞은 위치로 이동시켰다.

- 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 0,1,2,3,4 중 하나의 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. World view처럼 넓게 한 눈에 확인할 수 있는 카메라는 1번 카메라 즉 ‘0’ 숫자키와 연결된 카메라이므로 바닥과 좌표를 확인하려면 0을 누르는 것이 제일 한 눈에 확인 가능하다.

3) 최대 4개까지의 서로 다른 동적인 물체를 가상의 세상에 배치하라 (물체 당 10점 최대 40점). 각 동적 물체는 이동 변환, 크기 변환, 그리고 회전 변환 등의 기본 기하 변환 중 최소한 두 개 이상을 사용하여 서로 다른 움직임을 표현해야 하며, 각 동적 물체는 키보드 또는 마우스 동작을 통하여 움직임과 멈춤을 조절할 수가 있었야 한다 (자신이 선택한 최대 네 개의 동적인 물체에 대해 이 기능이 구현이 안되어 있으면 물체 당 4점 감점). 요구 사항은 아니나 동적인 물체들 중 최소한 1개의 물체에 대해서는 뉴턴의 운동의 법칙과 같이 물리적으로 충실한 방법을 사용하여 움직임을 표현해볼 것.

a) 사용한 물체:

- 부여한 **서로 다른** 모델링 변환:

- 확인 방법: 구현하지 못함

b) 사용한 물체:

- 부여한 **서로 다른** 모델링 변환:

- 확인 방법: 구현하지 못함

c) 사용한 물체:

- 부여한 **서로 다른** 모델링 변환:

- 확인 방법: 구현하지 못함

d) 사용한 물체:

- 부여한 **서로 다른** 모델링 변환:

- 확인 방법: 구현하지 못함

2. Viewing Transformation

1) 1번부터 4번까지의 카메라는 CCTV 카메라와 같이 주어진 위치에 고정하여 세상을 바라보는 카메라이다. 적절한 사용자 인터페이스 동작을 통하여 원하는 카메라에서 세상을 바라볼 수 있도록 하라

a) 1번 카메라 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘0’ 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다.

b) 2번 카메라 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘1’ 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. (0,0,-20)에서 (0.0.0)을 바라보는 카메라이다.

c) 3번 카메라 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘2’ 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. (20,0,0)에서 (0.0.0)을 바라보는 카메라이다.

d) 4번 카메라 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘3’ 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. (0,0,20)에서 (0.0.0)을 바라보는 카메라이다.

2) 5번 카메라는 동적인 카메라로서 사용자 인터페이스 동작을 통하여 다음과 같이 움직일 수 있도록 하라

\* Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘4’ 숫자키를 누르면 해당 키에 설정된 카메라를 통해 볼 수 있다. (-20,0,0)에서 (0.0.0)을 바라보는 카메라이다.

a) 5번 카메라 translation 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘4’ 숫자키를 누르면 해당 키에 설정된 5번 카메라를 통해 볼 수 있다. ‘q’키는 카메라의 -u 벡터 방향, ‘w’키는 카메라의 u 벡터 방향으로 translation한다. ‘a’키는 카메라의 -v 벡터 방향, ‘s’키는 카메라의 v 벡터 방향으로 translation한다. ‘z’키는 카메라의 -n 벡터 방향, ‘x’키는 카메라의 n 벡터 방향으로 translation한다.

b) 5번 카메라 rotation 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘4’ 숫자키를 누르면 해당 키에 설정된 5번 카메라를 통해 볼 수 있다. ‘e’키는 카메라의 -u 벡터 축으로, ‘r’키는 카메라의 u 벡터 축으로 rotation한다. ‘d’키는 카메라의 -v 벡터 방향, ‘f’키는 카메라의 v 벡터 방향으로 rotation한다. ‘c’키는 카메라의 -n 벡터 방향, ‘v’키는 카메라의 n 벡터 방향으로 rotation한다.

3) (추가) 1번 카메라에 대하여 고정된 위치를 중심으로 시선의 방향을 바꿀 수 있도록 하라. 어떠한 방식으로 구현할 지는 본인이 결정할 것.

a) 추가 구현 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘0’ 숫자키를 누르면 해당 키에 설정된 1번 카메라를 통해 볼 수 있다. 마우스 왼쪽 키를 누른 채로 화면에서 마우스를 이동하면 마우스의 이동방향대로 시선의 방향이 바뀐다.

3. Projection Transformation

1) 적절한 사용자 인터페이스 동작을 통하여 5번 카메라에 대하여 줌 인/줌 아웃 기능을 구현하라. 이때, 최대로 줌 인/줌 아웃할 수 있도록 적절히 범위를 설정하라.

- 확인 방법: Ctrl+F5를 하면 맨 처음 까만 화면이 나온다. 이때 ‘4’ 숫자키를 누르면 해당 키에 설정된 5번 카메라를 통해 볼 수 있다. ‘p’는 줌 인, ‘m’는 줌 아웃 기능을 갖고 있다. MAX\_ZOOM\_IN과 MAX\_ZOOM\_OUT을 통해 범위 역시 설정했다.