**기초 컴퓨터 그래픽스**

**HW2 README**

학번 : 20191583 이름 : 김태곤

**1. [환경 명세]**

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

- Windows 11 64bit, i5-12600KF, RTX 3080, Visual Studio 2022 Release x64

**2. [요구사항]**

(a) 다음 왼쪽 그림의 9개의 물체 중 5개를 선택한 후, 각 물체에 대하여 서로 다른 창의적인 동적 효과를 생성할 것. (각 물체당 최대 15점 (최대 총 75점))

1) 사용한 물체: 칵테일

- 부여한 창의적인 동적 효과:

칵테일이 x\*sin(x)의 곡선을 그리며 이동한다. 곡선을 따라 이동하며 x값의 변화량에 따라 칵테일이 회전한다. 또한, 캌테일이 움직일 때 초기 칵테일 크기의 8배에 2cos(x)를 곱한 만큼 크기가 변한다.

프로그램 내 슬라임이 등장하는데, 슬라임이 포탈(케이크와 집으로 이루어진)에 도착하면 칵테일이 한개 증가한다.

- 구현 방법:

cocktail\_clock = (timestamp%721)-360

rotate\_cocktail = cocktail\_clock \* TO\_RADIAN

scaling\_cocktail = 2\*(cos(rotate\_cocktail))

T(cocktail\_clock, cocktail\_clock\*sin\*(rotate\_cocktail), 0) R(rotate\_cocktail)

S(8+scaling\_cocktail, 8+scaling\_cocktail, 1) 을 해주었다.

- 확인 방법:

프로그램을 실행하면 즉시 나타난다. 슬라임이 포탈에 도달할 때까지 기다리면 칵테일은 하나씩 증식한다.

2) 사용한 물체: 케이크(+집)

- 부여한 창의적인 동적 효과:

케이크와 집을 이용해 슬라임이 이용하는 포탈을 생성하였다.

케이크는 총 10개 생성되어 있다. 각각의 케이크는 한 점(집이 있는 곳)을 기준으로 36도만큼 회전되어 있다. 여러개의 케이크가 모여 하나의 원을 이루고 있다. 원을 이루고 있는 케이크는 집을 기준으로 반시계 방향으로 돌아간다.

집은 원을 이루고 있는 케이크의 중심에 있다. 집은 한 지점에 고정되어 있지만 시간의 흐름에 따라 크기가 변한다. 크기는 3+sin(3x)-0.5(sin(2x))를 기준으로 커졌다 작아졌다를 반복한다.

- 구현방법 :

cakeclock = timestamp % 360

T(-500,0,0) R(cakeclock\*TO\_RADIAN+i\*36) T(3,100,0) S(4.5,4.5,1) 을 10개 해주었다.

Houseclock(angle) = timestamp%360\*TO\_RADIAN

T(-500,0,0) S(3+sin(3\*houseclock)-0.5\*sin(2\*houseclock), 3+sin(3\*houseclock)-0.5\*sin(2\*houseclock), 1) 을 해주었다.

- 확인 방법:

프로그램을 실행하면 즉시 화면에 나타난다.

3) 사용한 물체: 자동차

- 부여한 창의적인 동적 효과:

화면의 바닥에서 cos(xsin(x))의 곡선을 그리며 움직인다. 이 자동차는 화면의 양쪽 끝, 중심을 지날 때마다 한개씩 늘어나며, 최대 8개까지 생성된다. 양쪽을 왕복운동하며 처음 왼쪽으로 갈때는 크기가 일정하지만, 오른쪽으로 갈 때는 3+cos(xsin(x))만큼 크기가 증가한다.(자동차의 높이에 따라 크기가 변화한다. 높을수록 자동차는 커진다.)

- 구현 방법:

car2\_clock = timestamp % 2001

car2\_x = car2\_clock

car2\_y = 50 + cos(car2\_x + sin(car2\_x/10000))

T(car2\_x, car2\_y-300\*i\*100, 0) S(-(3+cos(car2\_x\*sin(car2\_x/10000))), 3+cos(car2\_x\*sin(car2\_x/10000)), 1) 해주었다.

- 확인 방법:

프로그램을 실행하면 즉시 화면에 나타난다.

4) 사용한 물체: 셔츠

- 부여한 창의적인 동적 효과:

슬라임이 이용하는 길을 만들어 주었다. 100개의 빽빽한 셔츠를 만들어 이어주어 길을 생성하였다. 이 셔츠길은 왼쪽으로 계속 이동하며 일정 시간의 흐름에 따라 위아래가 계속 바뀐다.(대칭이동한다.)

- 구현 방법:

shirtclock = timestamp % 1081

T(-shirtclock+500-i\*40, -30,0) S(1,flip,1)를 100개 해주었다.

- 확인 방법:

프로그램을 실행하면 즉시 화면에 나타난다.

5) 사용한 물체: 비행기

- 부여한 창의적인 동적 효과:

하늘에서 한바퀴 도는 곡예비행을 모티브로 만들었다.

비행기는 주기적으로 x축 방향으로 이동한다. 비행기는 진행 방향을 바라보며 이동한다. 이동과 함께 회전하여 3D상으로 생각하면 x축에 평행한 직선 기준으로 360도 회전하며 회오리처럼 전진한다.

비행기는 캐릭터의 팔과 다리로 사용되었다. 키보드의 방향키를 이용하면 비행기로 이루어진 다리가 다리 끝을 기준으로 좌우 일정각도 회전을 반복한다. 이 회전은 쯔꾸르 게임의 분위기를 내기 위해 부드럽게 회전하지 않고, 좌우 최대 각도로만 즉시 움직이게 하였다.  
- 구현 방법:

하늘을 나는 비행기

airclock = timestamp % 1080

T(-500,250,0) R(airclock\*TO\_RADIAN) T(3,100,0) R(90\*TO\_RADIAN) S(1.5,1.5,1) 해주었다.

다리에서 회전운동 하는 비행기

T(centerx-393,centery+40) R(leg\_angle) T(0,-50,0) S(0.5,2,1)을 해주었다.

- 확인 방법:

프로그램을 실행하면 즉시 화면에 나타난다. 키보드의 방향키를 누르면 비행기 다리의 회전을 확인할 수 있다.

(b) 충분히 복잡도가 있는 2차원 기하 물체를 자신이 한 개 모델링 한 후, 위의 물체들처럼 창의적인 동적 효과를 생성하라. (최대 총 25점)

- 부여한 창의적인 동적 효과:

메이플스토리의 슬라임을 구현하였다. 슬라임의 몸통을 보다 더 부드럽게 표현하기 위해 지오지브라를 이용해 슬라임 이미지에서 점을 하나하나 따왔다.

슬라임은 주어진 셔츠길 위를 통통 점프하면서 이동한다. 슬라임이 점프할 때 살짝 shearing을 주어 슬라임이 점프하는 것을 좀 더 실감나게 묘사하였다. 슬라임은 집이 있는 포탈에 도달하면 처음 위치로 respone된다. 또한 캐릭터가 휘두른 칼에 4번 맞으면 슬라임은 빨간색으로 변해 아파하며, 뒤로 넉백된다. 빨간색이 된 슬라임은 일정 시간이 지나면 다시 되돌아온다.

- 확인 방법:

프로그램을 실행하면 즉시 화면에 나타난다. 캐릭터가 슬라임 주변에서 a를 4번 누르면 슬라임의 색이 빨갛게 변하며 뒤로 넉백된다.(캐릭터가 휘두르는 칼이 기준이다.)

(c) 제출한 숙제 중 가장 재미있고 복잡도가 있는 기하 변환 및 애니메이션 효과를 생성한 학생을 적절히 선정하여 최대 25점까지 추가 점수를 부여할 수 있음 (정확한 숫자는 상황에 따라 유동적임). 공정을 기하기 위하여 컴퓨터 그래픽스 연구실 대학원생들이 공동으로 심사함.

- 추가 점수를 부여 받아야 할 이유   
(복잡한 기하 변환과 재미있는 애니메이션 효과 사용과 확인 방법 등을 서술)

게임을 모티브로 따와 검을 휘두르는 전사 캐릭터가 있었으면 좋겠다고 생각하였다. 이에 과제에서 주어졌던 다양한 물체를 활용하여 캐릭터를 만들었다. 모자를 쓰고, car1과 car2를 이용해 얼굴을 구현하였다. 이후 셔츠로 몸통을 만들고 비행기를 이용해 팔과 다리를 제작하였다. 그리고 한쪽 손에는 칼이 쥐어져 있다.

이 캐릭터는 방향키를 이용해 움직일 수 있다. 캐릭터가 움직이면 다리도 함께 움직여 걷는 모션을 추가하였다.

캐릭터가 슬라임에게 다가가 ‘a’키를 눌러 칼을 휘두르면 슬라임을 공격할 수 있다. 슬라임이 4번 공격 받으면 빨간색으로 변해 아파하며, 뒤로 넉백된다.

슬라임이 포탈에 도착하면 처음 위치로 respone되며 칵테일이 증식되는데 이를 막기 위해서는 슬라임을 공격해 포탈로 가지 못하게 막아야한다!

이번 과제에서 게임의 기본적인 형태를 구현하기 위해 노력하였다. 플레이어의 이동, 공격, 몬스터 구현과 각각의 애니메이션 모션을 적용했다. 또한 플레이어와 몬스터의 상호 작용이 가능하게 구현하였다.(슬라임의 중심 기준 박스모양 콜리더) 이는 게임의 가장 기본적인 기능들은 구현하였다고 생각한다.

느낀점 : 기존에 유니티를 활용하여 3D 메타버스도 만들어 보았고, 지금은 베타버전까지 완성한 출시 직전의 게임도 있다. 다양한 게임을 만들면서 지금까지 게임 제작 툴의 원리를 모른채 배우고 사용해 왔는데, 이번 과제를 통해 그 툴의 동작 원리를 일부 이해할 수 있었고, 좀더 low level에서 작성해 보면서 사용자들의 편의에 맞춰진 게임 엔진에 감사함을 느꼈다. 또한 기회가 된다면 자체 엔진을 사용하는 회사나 연구실에 들어가 제대로 3D게임 엔진을 개발해 보고 싶다는 생각을 하였다.