

Title: 운석을 피해 우주를 탐사하자

학번: 2018147558

이름: 김 정 주

1. 요약: 이 게임은 앞에서 다가오는 운석을 피해 우주를 탐사하는 게임입니다. 우주선의 생명은 3개이며, 3번 운석에 충돌하면 게임 종료가 됩니다. 좌우 버튼을 누르면 해당 방향으로 기우는 애니메이션이 있으며 운석에 충돌하면 투명화가 되어 2초동안 무적상태가 됩니다. 게임을 중간에 일시정지 하는 기능도 있습니다.
2. 사용법: up, down, left, right arrow 버튼으로 상하좌우로 움직이며, 처음에는 space bar를 통해 시작할 수 있으며, 게임 중간에 space bar를 통해 일시정지가 가능합니다. 게임의 어느때든 esc 버튼을 누르면 게임이 종료됩니다.

3. 기능 구현 표:

	기능	구현여부	구현위치	비고
1	Vertex shader	O	Main.cpp(280, 281, 486, 537), model.vs, planet.vs, star.vs, test.vs	
2	Fragment shader	O	Main.cpp(280, 281, 486, 537), model.fs, planet.fs, star.fs, test.fs	
3	Transformation	O	Main.cpp(360, 447)	
4	Viewing (Camera)	O	Main.cpp(638, 578)	
5	Texture	O	Main.cpp(178)	
6	Lighting	O	Main.cpp(640, 641), model.fs	
7	Model Loading	O	Main.cpp(333, 434)	
8	Text	O	Main.cpp(483), text.vs, test.fs	
9	Animation	O	Main.cpp(700, 715, 730)	

4. 기타 사용 기능

A. Geometry shader

- i. 구현 위치 : main.cpp(573), model.gs
- ii. 간단한 기능 : geometry shader는 vertex shader와 fragment shader의 중간단계이며, 모든 vertex에 대한 vertex 데이터에 접근할 수 있다. 그렇기에 데이터의 양을 증가하거나 감소시킬 수 있다. 그렇기에 삼각형으로 그리는 것을 점이나 선으로 그리도록 바꿀 수 있고, vertex shader에서 지정해준 vertex에 더 많은 표현을 추가하여 그릴 수가 있다. 그래서 이 기능은 털 같은 하나의 vertex에서 많은 선이나 점 같은 표현을 하기에 적합하다. Geometry shader에는 특수하게 제공되는 primitive를 사용하여, 주어진 vertex에 대한 새로운 데이터 생성이 가능하다. EmitVertex()를 통해 vertex에 새로운 데이터를 생성하며, EndPrimitive()를 통해 새로운 데이터에 대한 계산을 수행하는 방식으로 구동된다.
- iii. 구현 기능 : 이 프로그램에서는 geometry shader가 vertex의 정보를 받아 각 vertex마다의 normal 방향으로 삼각형을 움직이는 역할을 수행한다. 일시정지를 하였을 때, 배경이 멈추는 것 빼고는 구별하기가 어려울 것이라고 생각해서, 이 기능을 넣어서 모델이 시간의 변화에 따라 vertex의 normal 방향으로 움직이다가 다시 제 모습이 되는 것을 구현했다.

B. Blending

- i. 구현 위치 : main(464), model.fs(43)
- ii. 간단한 기능 : alpha blending은 강의시간에 잠깐 언급된 적이 있다. 이 기능은 z축이 카메라와 가까운

물체와 먼 물체 사이의 색을 blending해주는 역할을 한다. 흔히 투명도 조절이라고 할 수 있다. 그렇기에 유리창과 같은 투명도를 가지는 물체를 그리기에 적합한 기능이다. 이 기능의 사용을 위해서는 `GL_Enable(GL_BLEND)`를 사용하여 블렌딩 기능의 사용을 선언하고, `glBlendFunc`함수를 사용하여 어떤 방식으로 alpha값을 사용하여 블렌딩을 할 것인지를 조절할 수 있다.

- iii. 구현 기능 : 우주선이 운석에 맞았을 때 잠시동안의 무적을 구현하려고 했으나, 눈으로는 무적상태라는 것을 판단할 수 없었기에 무적상태일 때는 이 기능을 통해 우주선이 투명하게 보이는 효과를 구현 했다.

C. Anti-Aliasing

- i. 구현 위치 : `main.cpp(596)`
- ii. 간단한 기능 : 이 기능은 수업시간에도 잠깐 언급된 적이 있다. Aliasing은 픽셀에서 데이터를 결정하기에 주어진 데이터가 한 데이터로 나타나기에 애매한 것일 뿐 발생한다. opengl에서는 점, 선, 다각형의 표현에서 anti-aliasing을 해주는 기능이 존재한다. 이는 `glEnable(GL_POINT_SMOOTH)`, `glEnable(GL_LINE_SMOOTH)`, `glEnable(GL_POLYGON_SMOOTH)`를 통해 각각 선언하여 사용함을 알리고, `glHint` 함수를 통해 조정에 대상을 어떻게 조정할 것인가를 설정할 수 있다. 이를 통해 우리가 그린 다각형의 주변이 계단현상을 나타내는 것을 방지할 수가 있다.
- iii. 구현 기능 : 전체적인 부분에 적용되며, aliasing을 방지하도록 구현했다.

5. 기타