

Interactive Graphic Programming

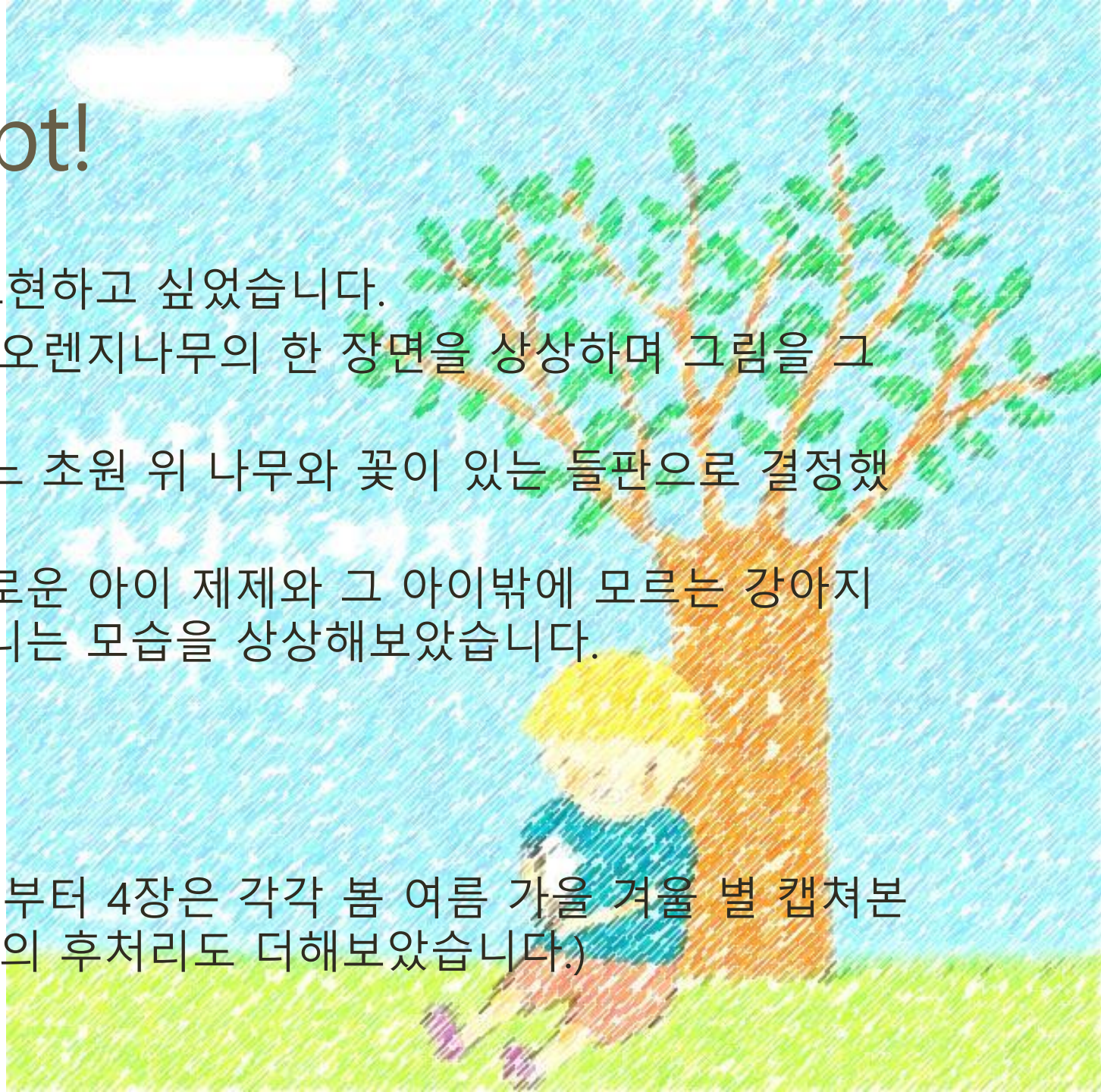
FINAL PROJECT 20132390오성기

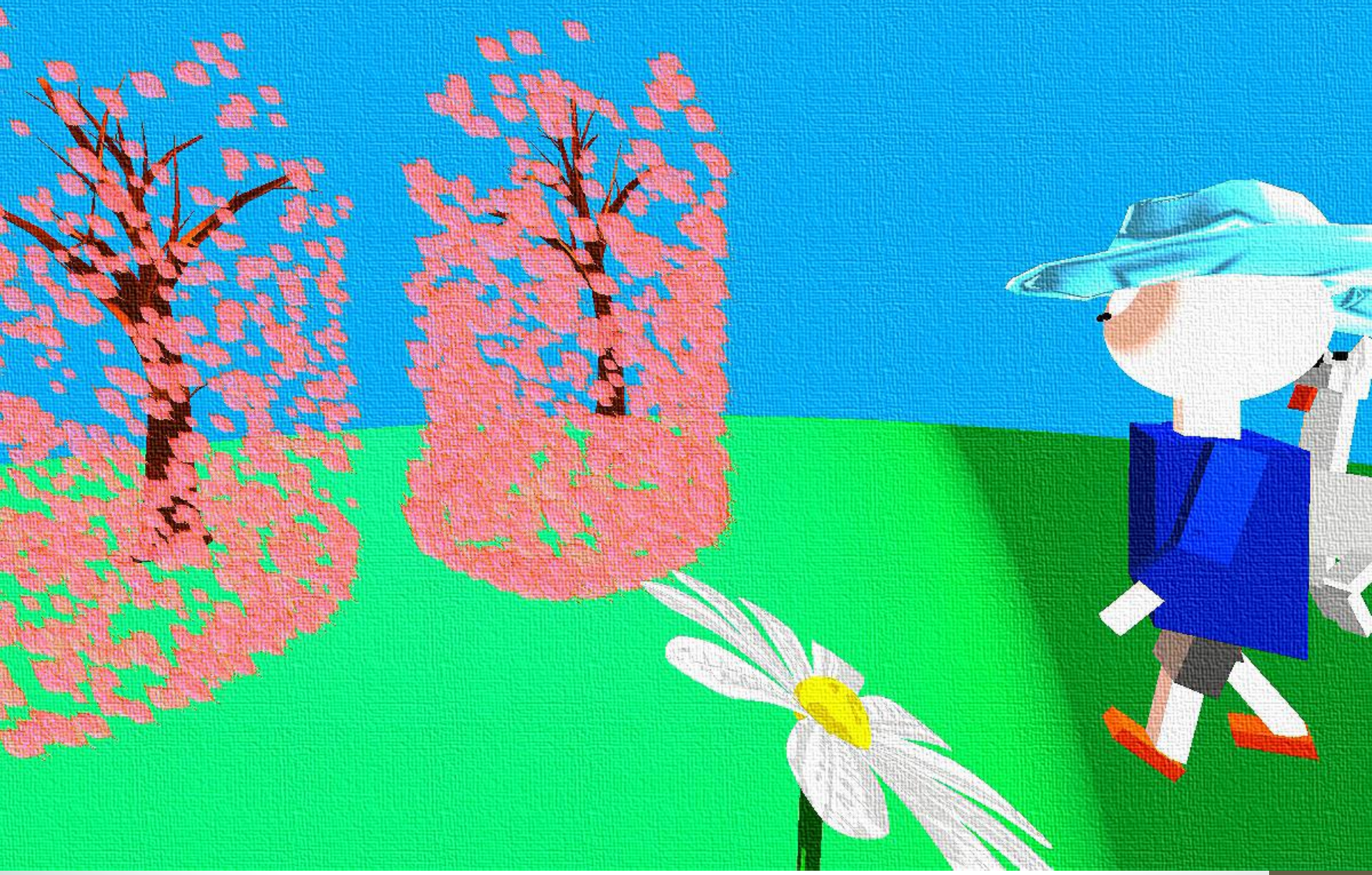


Concept!

- 4계절을 표현하고 싶었습니다.
- 나의 라임 오렌지나무의 한 장면을 상상하며 그림을 그렸습니다.
- 장소는 어느 초원 위 나무와 꽃이 있는 들판으로 결정했습니다.
- 그곳에 외로운 아이 제제와 그 아이밖에 모르는 강아지가 뛰어다니는 모습을 상상해보았습니다.

(다음 페이지부터 4장은 각각 봄 여름 가을 겨울 별 캡쳐본입니다. 살짝의 후처리도 더해보았습니다.)





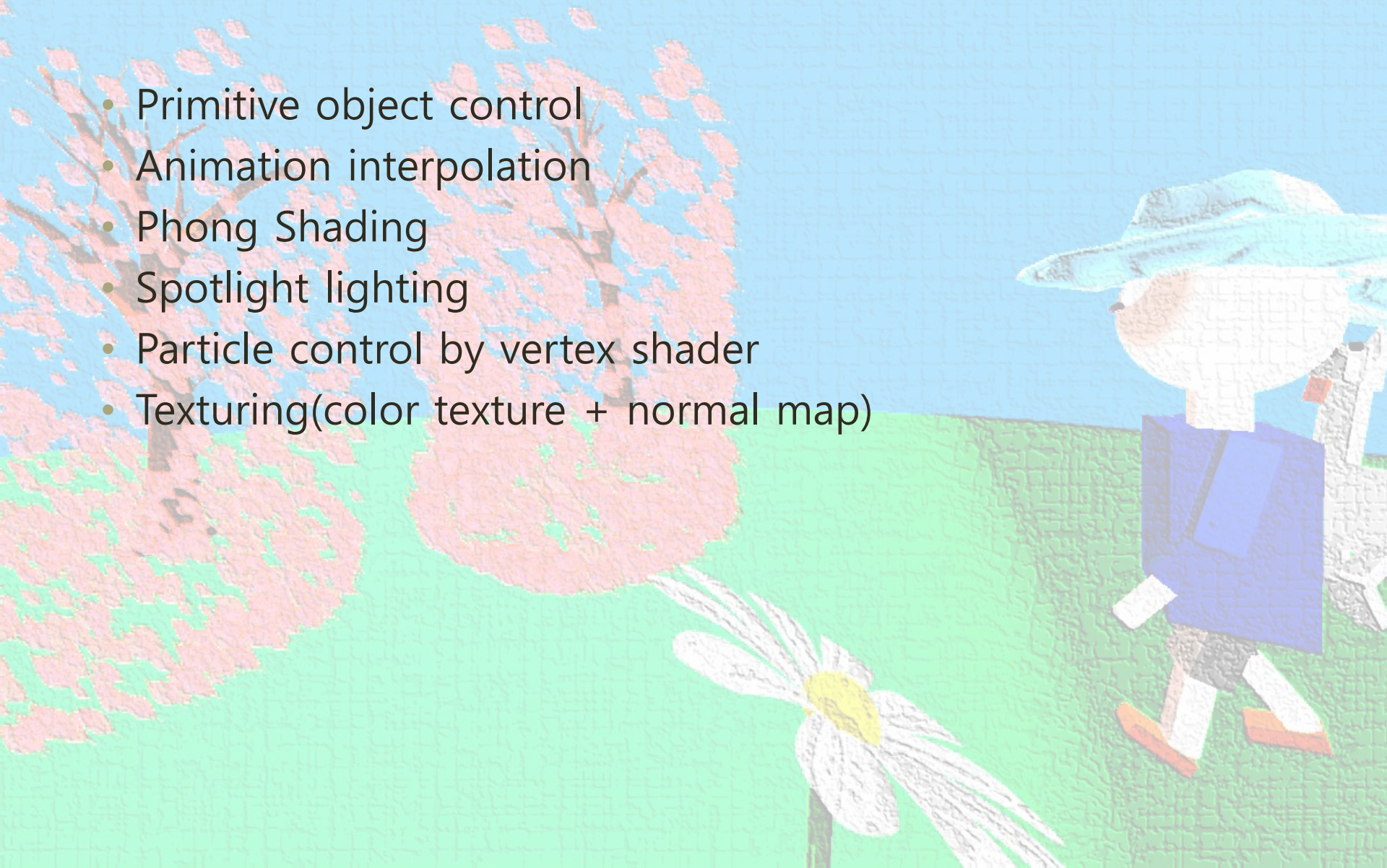






사용된 기술

- Primitive object control
- Animation interpolation
- Phong Shading
- Spotlight lighting
- Particle control by vertex shader
- Texturing(color texture + normal map)



Primitive object control

- Primitive 객체를 cpp + header 파일로 지정하여 class 객체를 불러오는 것으로 rendering code를 객체화 함
- > 코드 설계의 복잡도를 낮춤
- 지정된 객체 : flower, humo(뛰어다니는 아이), runningDog, snow(눈과 비), tree

flower

- 다중 obj 파일의 경우 Bmp 파일을 이용한 텍스처맵핑이 제대로 되지 않는 점을 발견
- Maya프로그램을 이용하여 3개로 obj를 따로 저장하여 불러온 뒤 color를 입힘
- 3개 obj를 렌더링하는 함수를 class member 함수로 지정 함으로써,

vbocube->render();와 같은 문법을 사용할 수 있도록 함
(flower->render();)

humo

- Project 3에서 만들었던 강아지를 이용하여 humanoid 만든 뒤 클래스 객체화 함
- Flower와 같은 문법으로 렌더링 가능
- VBOCube + VBOSphere + VBOmesh(obj)
- 얼굴 : sphere, 모자 : hat.obj 그 외 몸통 : cube
- Hierarchy&instance 변환을 사용하여 인간과 같은 움직임을 나타낼 수 있도록 설계
- 클래스 내 멤버함수인 Autorun(); 를 이용하여 뛰는 동작에 필요한 변수 변위를 make set

runningDog

- Project3에서 제출하였던 전체 소스코드를 cpp + header로 객체화 하여 runningDog->render();의 문법으로 메인 소스에서 불러올 수 있도록 수정

Humo&Dog interpolation

- Humo&Dog은 클래스 내에 현재 위치 변수vec4 position을 가짐
- Interp class 내의 rand_destination함수를 이용하여 Humo의 Destination을 난수함수로 지정
- z평면 위 Destination Vector와 position Vector의 dot product와 length를 이용하여 $\cos(\theta)$, $\sin(\theta)$ 값을 구하고, 객체의 RotateY를 \cos, \sin 으로 부터 구함
- 이동방향을 \cos, \sin 으로 지정하여 Translate하게 만들어 난수적인 움직임을 나타낼 수 있도록 함
- 한 이동의 frame은 100으로 설정
- Dog의 Destination은 Humo의 좌표로서, 주인을 따라다니도록 설계함

-> 문제점 : 대부분의 경우 맞게 움직이지만 원점을 지나는 운동일 경우, 급격히 운동변위가 증가하는 오류가 발생, 이는 \cos 을 구하는 과정에서, length로 나누어질 때 divided 0 오류가 발생하는 것으로 보임

snow

- 눈이 내리는 모습을 표현하기 위해 Particle에서 사용되었던 point sprite를 사용
 - Texture를 직접 제작하여, Texture number를 바꾸는 것으로 snow코드로부터 비까지 표현가능하였다
 - Point sprite는 2D이기 때문에 Primitive Projection이 불가능함 -> eye좌표로부터 z좌표 값 차이를 계산하여 gl_pointsize 함수의 인수를 각각 다르게 처리함
- > 개별적 pointsize 지정을 위해 gl_Begin()&gl_End()를 사용함
- Buffer를 이용하여 Gluint 형으로도 개별적 제어가 가능할 수 있을까 했으나 vertex shader내에서 gl_pointsize()함수가 작동되지 않아 cpp에 처리부분을 포함시킨 것이 아쉽다.

tree

- Tree의 stem 부분은 obj파일을 불러와 brown컬러로 fragment_shader를 이용하여 색을 칠함\
- 나뭇잎의 텍스처를 알파채널을 포함하는 bmp로 직접 제작하여 지정, 색을 계절에 따라 분홍, 초록, 노랑, 빨강으로 바뀔 수 있게 하였다.
- Uniform변수로 vec4 orgcolor를 이용하여 main code에서 shader로 컬러값을 전달할 수 있도록 설계하였다.
- Particle과 조합을 시켜 나뭇잎이 떨어지는 것을 표현해보았다.

Tree Particle

- 나뭇잎을 MAX_LEAFS로 개수를 Define하여 난수함수로 x,y,z 좌표를 지정
- 나뭇잎이 구의 형태로 분포되어있다고 가정하여 x,y,z좌표를 각각 $\sin T \cdot \cos P$, $\sin T \cdot \sin O$, $\cos P$ 로 지정함(bounding에 구면 좌표계를 적용)
- Vertex shader를 이용하여 main에서 각 점의 좌표와 initvel 정보를 uniform 변수로 넘겨주고 shader에서 Translating 처리 진행
- 땅에 쌓이는 모습을 표현하기 위하여 일정 값 이하로 Translating되면 정지되도록 하여 쌓이도록 하였다

Lighting

- Keyboard Function을 이용하여 Light 옵션을 켜고 끌 수 있도록 설정함

7 : spotlight

8 : phong shading

- 낮과 밤을 표현하기 위하여 Light의 K_d, K_s, K_a 값을 조정하여 vec3 변수에 저장함
- Uniforming 함수의 인수 포인터 값을 바꿔주는 것으로 상황에 맞는 set들을 fragment shader로 넘겨줄 수 있도록 설정함

5 : 낮

6 : 밤

Texturing

- 영상에서는 좌표 밖으로 나가서 안보이지만 오른쪽 윗쪽에 태양이 존재!
 - Project4 에서 사용하였던 태양을 그대로 가져 와서 사용함
- >문제점 : 실제 태양처럼 보이기 위해 위치시켜야 할 좌표가 애매했다

Fragment shader

- Uniform형인 coloring 변수를 받아옴으로써, option을 선택하여 lighting할 경우를 나눠서 처리하도록 하였다.

Default option : Phong shading

Option 1 : Primitive object coloring

Option 2 : Particle Shading

->추가적으로 Phong이나 Spotlight를 추가 할 수 있도록 uniform 변수로 판단하는 if문을 추가하였다.

Vertex Shader

- Option 변수를 통하여 Particle shading이나 normal shading 중 하나의 작업을 수행한다
- Tree의 나뭇잎을 float 형인 poslimit변수를 이용하여 일정 값 이하로 내려갈 시에 멈추게 하여 쌓이는 효과를 표현한다.
- Lifetime이 다 되었을 경우 서서히 사라지도록 한다.

Etc

- 4계절을 표현하기 위하여, 내리는 텍스처의 종류, 나뭇잎의 색 등을 base.h 내의 season class에 포함시켜 class 객체를 바꾸는 것으로 전체적 제어가 가능하도록 하였다.
- 멤버변수 초기화는 main의 startup()에서 진행
- 카메라 좌표가 원형 웨이브 좌표로 움직이게 하였다. (시점은 원점 고정)
- Main의 render()함수가 매우 짧다? :

각 객체를 하나로 묶어 계층구조로 만들었기 때문!
따라서 object->render();문 직전에 model행렬을 수정해주면 쉽게 원하는 객체 배치를 할 수 있다.(Vertex Buffer Object의 rendering pipeline에서 아이디어를 얻음)

Etc

- Mouse :

왼쪽부터 Rotate, Translating ,Scaling

- Key 정보 :

B : eye좌표 자동 이동 켜기/끄기

7 : spotlight 켜기/끄기

8 : phong 켜기/끄기

` : 나뭇잎 초기화/떨어뜨리기

1 : 봄 (나뭇잎 분홍, 느린 비)

2 : 여름(나뭇잎 초록, 빠른 비)

3 : 가을(나뭇잎 노랑, 느린 비)

4 : 겨울(나뭇잎 적갈색, snow!)

5 : 낮

6 : 밤

N : 계절 별 눈/비 내리는 것 켜기/끄기

감사합니다!

날씨가 많이 춥고 시리지만
마음만은 따뜻한 겨울이 되길 바랍니다!