

# **A4: Full-Solar-System**

**2017313135 소프트웨어학과 권동민**

## Data Structure

For making Dwarf easily, I fixed some data structure "circle\_t". In Assignment3, I added constant theta for each planet in main function, but now I set the constant theta(delta theta) in circle\_t structure.

So I added more features to circle\_t.

// For rotation

*Theta* : Current theta for rotation. *Dt\_theta* : Amount of theta change in each planet per frame.

// For orbit

*Origin* : the orbital id of each planet *Global\_theta* : Current theta for orbit from the origin. *Dt\_global\_theta* : Amount of global\_theta change in each planet per frame.

For making ring, I made ring\_t structure and set info of each ring.

*Id* : each ring's id *Tex* : texture index (whitch texture for fragment shader)

*Center* : position of the origin of the ring *Radius* : radius of ring(based on outline)

## Algorithm

### Dwarf planets

It is easy to update Dwarf planet's center position and rotation angle because I already set the information in circle\_t structure when each planet were initialized. The center position can be calculated as follows.

$$\begin{aligned} \text{center pos} &= \text{origin's center pos} \\ &+ \text{distance from origin}(\text{rotating by using global}_{\theta}) \end{aligned}$$

When I set the information of each planet, I set also origin planet's id to "int origin" so I can access the origin's center position by using id as index of circle vector.

### **Set Texture and Shader**

There are many textures for shading(SUN, VENUS, MOON, ...) so I set the array "tex" to access each texture by index. And I updated each planets' texture by using planet's id as index of tex array. If the planet's id is 0(it means that planet is SUN) then, I set the mode to 1, because in fragment shader I shouldn't apply light and material.

### **A textured ring system of the Saturn.**

I used distinct vertex array for set ring's vertex. It is simple to make vertex array(just two circle like Assignment 1 to create) and set the index buffer using vertex's position.(consider face curling) and set normal vector(I think that it is better to set the normal vector equal to position vector for shading)

In render function, I set the variable "mode" to 2 when I draw the ring. And I used two textures("TEX", "TEX\_A") for alpha blending. So in fragment shader, first I set the color with TEX\_A, and calculated alpha value of final frag color.