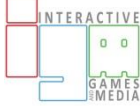
	<p style="text-align: center;">Rochester Institute of Technology Golisano College of Computing and Information Sciences School of Interactive Games and Media 2145 Golisano Hall – (585) 475-7680</p>	
---	--	---

Data Structures & Algorithms for Games & Simulation II
IGME 309, 2015 Fall
A5: 3D Transformations

Due: September 20th 2015 at 23:59Hrs.

This is not a team's homework.

You have starting code through ReEngine, you may use your own solution if you want to, but the translation of starting code is entirely your responsibility. Startup code is available at: <https://github.com/labigm/ReEngApplication> under the A05_Transformations project.

The goal of this homework assignment is to get practice performing transformations in different coordinate systems.

Using the provided code for this homework assignment you will make the Earth perform a full rotation around the sun while having the moon revolve around it.

The Sun is going to be static in the origin of the World Coordinate System (unless you are going for the extra credit); the Earth is going to perform a 360 degrees rotation around it. During that time it should have rotated around its own axis 360 times (Earth's day). The Moon is going to be revolving around its own axis but will also be revolving around the Earth (which again, in turn, should be revolving around the sun) in such a way that it only shows one face to the Earth at all time; its rotation period should be 28 earth-days.

The scales to be used and distances will be:

Planet/Star	Size	Distance from the sun
Sun	5.936(relative to the world)	0(relative to the world)
Earth	0.524(relative to the world)	11(relative to the world)
Moon	0.27(relative to the Earth)	2(relative to the Earth)

More info about moon's rotation in: https://www.youtube.com/watch?v=OZIB_leg75Q

You are provided with the following variables:

```
float m_fEarthTimer;  
float m_fMoonTimer;
```

This program is processor dependent and not time dependent so the rotation increments will be measure in cycles and not in seconds. You can safely assume that the rotation of de moon is going to be one degree per cycle (so Earth's would be 1/28 per second);

You will need to investigate the use of:

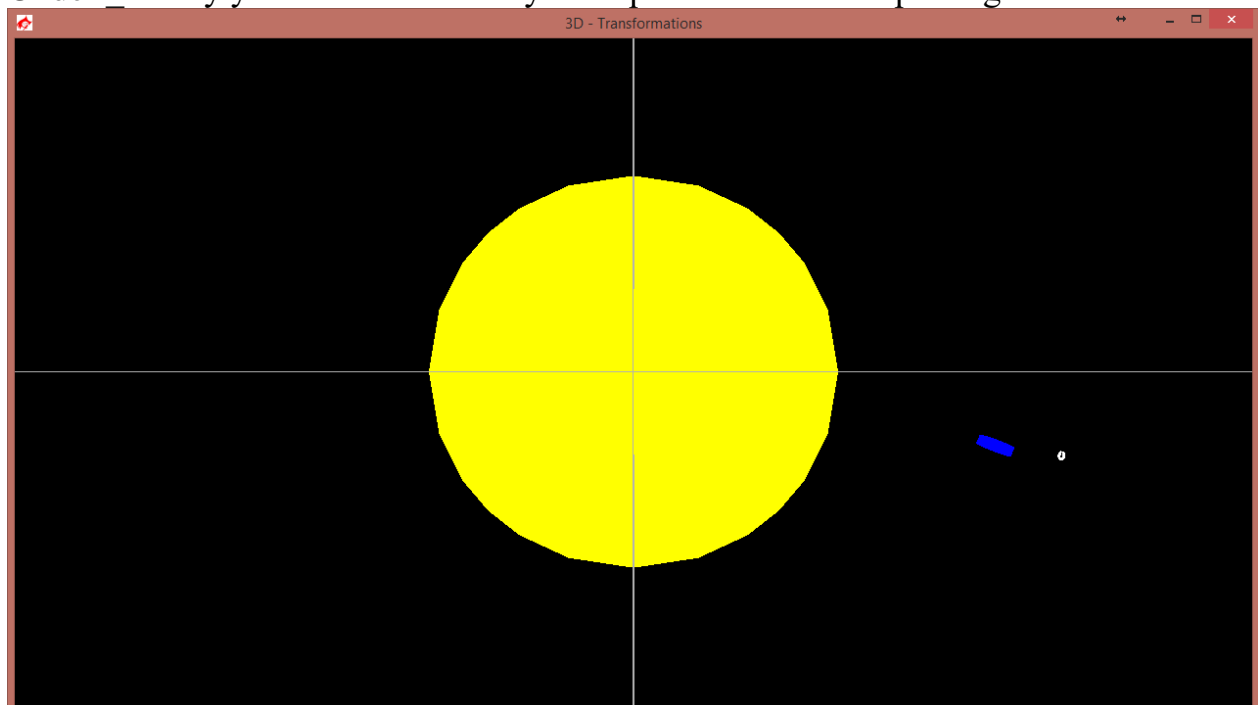
```
glm::rotate(...)  
glm::translate(...)  
and possibly glm::scale(...) You can create your shapes at the right size at the beginning.
```

You will notice under Display the following lines:

```
m_pSun->Render(m_m4Sun);  
m_pEarth->Render(m_m4Earth);  
m_pMoon->Render(m_m4Moon);
```

They are going to place the objects into the designated positions. So all of the transformations should be held into those matrices.

Under `_Binary` you will find a binary example of what I'm expecting:



Please keep your shapes as created; that will help the grader. It's really hard to appreciate rotations on a perfect sphere.

Extra (15%):

Make the sun move using the X, Y, Z keys (positive but Negative if pressing Shift) and let the Earth and the Moon move with it (revolving of course).

Submit to the dropbox labeled A5 Transformations