

# ***Computer Graphics***

***by Ruen-Rone Lee***  
***ICL/ITRI***



# ***Assignment #1***

***Draw some 3D models with  
Viewing Transformations  
(Model/Viewing/Projection Transformation)***



# ***Purpose of the assignment***

- ◆ **Know how to manipulate the 3D models**
- ◆ **Know how to apply model transformations**
- ◆ **Know how viewing transform affect your results**
- ◆ **Know what's the difference between parallel projection and perspective projection**

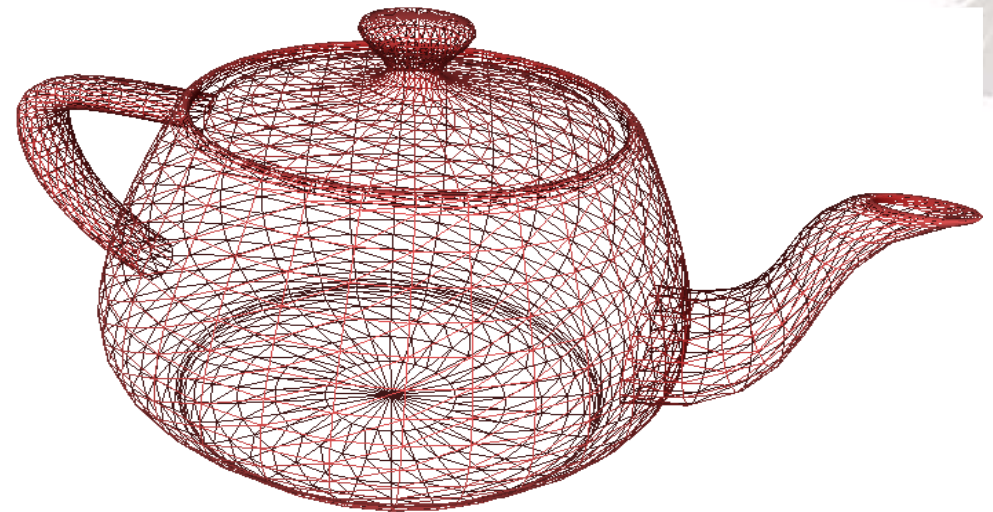


# Requirements

- ◆ You are required to use the framework that TA provided to draw some 3D models and display them in solid or in wireframe mode



Solid mode



Wireframe mode

# Requirements

- ◆ You are required to use the framework that TA provided to do some transformations, such as **geometrical, viewing, and projection**, on the input 3D models.
- ◆ **Interactive control is required.** That is, response (re-display) immediately after any mouse or keyboard action.



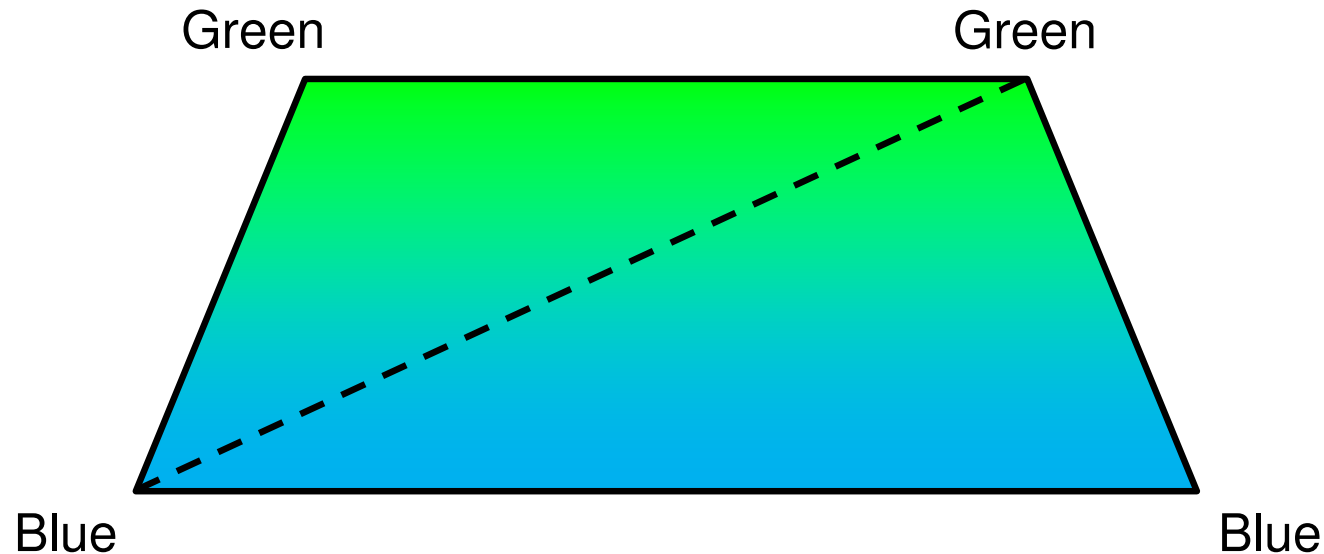
# Requirements

- ◆ **All the transformations (geometrical, viewing, projection) should be implemented**
  - Geometrical transformation – translation, scaling, rotation
  - Viewing transformation – similar to gluLookAt function
  - Projection – parallel and perspective projection, similar to glOrtho and glFrustum (or gluPerspective)
  - ***Notice: all fixed function OpenGL transformation APIs are not allowed***



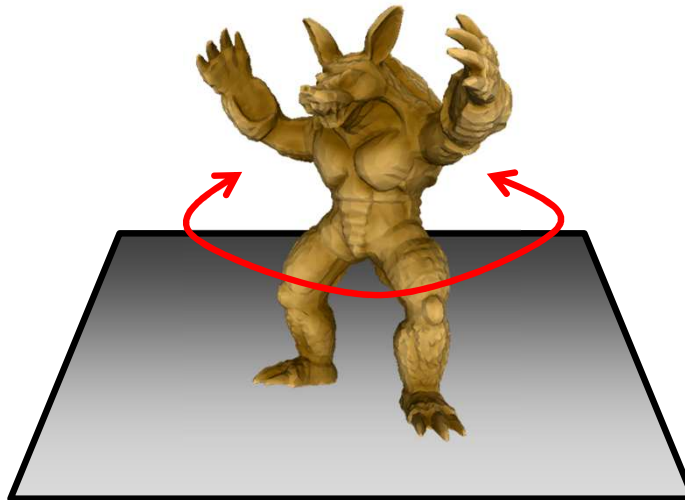
# Requirements

- ◆ Design a “**world**” that can load a specific model
- ◆ The “**world**” should contain a **default base floor** (a square or just two triangles)
  - Set the vertex colors so that the **floor** is displayed with the color you assigned. For example,



# Requirements

- ◆ Load a specific 3D model and display it with **default viewing**
  - Load a specific model as provided by TA's codes
  - Place the model properly “**above**” the base floor



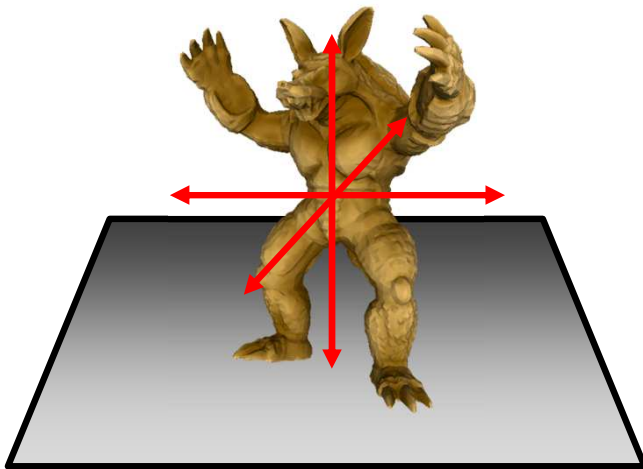
- Press a key to enable **self-rotate** in y-axis



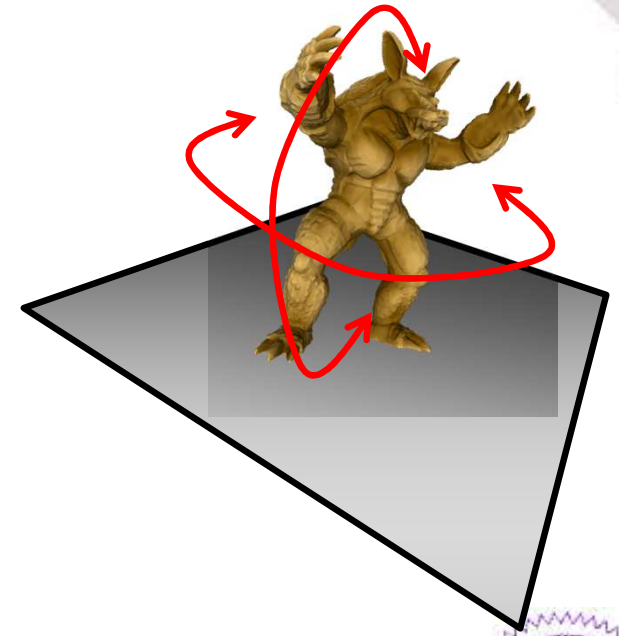
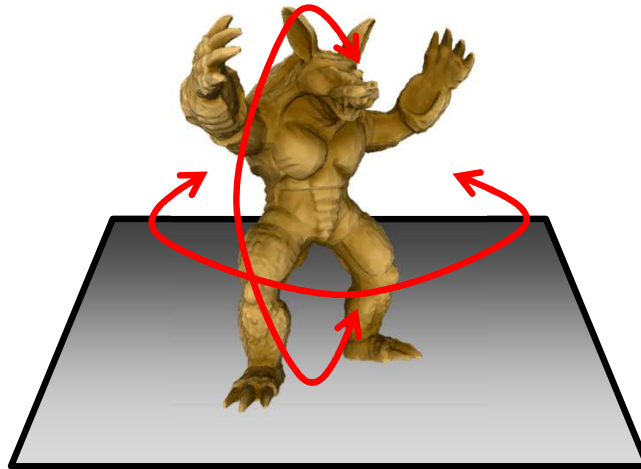


# Requirements

- ◆ Set the default viewing direction to view **from positive Z to origin** (toward negative Z direction)
- ◆ Set the default projection to **perspective** projection
- ◆ Use **keyboard** and **mouse** to do all the required transformations



Geometrical Transformation



Viewing Transformation

# *Hints for the Interactive Control*

- ◆ **Follow the guidelines that TA provided to write the required transformations codes such as geometrical, viewing, and projection, based on the input controls (keyboard, mouse).**
- ◆ **Apply those generated matrices in the **vertex shader codes** to perform the transformation with respect to the input control**

# *Input Model Format*

- ◆ Wavefront 3D Graphics color models provided by TA
  - There are some **validation 3D models** released to validate your design during your code development
  - There are also some **testing 3D models** that **will not** be released but is used for TA to test and grade your work
  - ***Check the TA's template for how to provide a list of 3D models in running the program***
    - TA might change the testing sets randomly



# Hints

- ◆ Use TA's sample codes as the basic foundation to revise and add the functions required in this assignment
  - Framework with **glfw+glad**
  - Model loader with **tinyOBJ**
- ◆ Set the viewing direction always from the eye position to the origin (assume the model has been normalized and placed at the origin)



# Hints

- ◆ The base floor will be seen only when the viewing direction is not aligned with the z-axis
- ◆ For geometrical transformation, it should apply to the model only
- ◆ For viewing transformation, it should apply to both the model and the base floor

# Hints

- ◆ You have to illustrate your control clearly so that TA can justified the correctness of your implementation
- ◆ You are required to demonstrate the implementation to TAs if there is a need
  - Operations incorrect
  - Insufficient documentation for the operations
  - **Book the time with TAs before you go or check with them after class**



# Hints

- ◆ Check with TAs if you would like to demonstrate your work personally
  - Any fancy operations or functions regarding the assignment
  - Any doubt to the score that TAs have graded
  - **Again, check the time with TAs first or check with them after class**



# ***Due Date***

- ◆ **Three weeks** after announcement (should be **4/28**)
- ◆ Late submission is allowed with less score
- ◆ **No score if you did not submit your assignment**
- ◆ Plagiarism is strictly forbidden
  - If you copy from others, your score will become zero
  - The score to the one who provides the original copy will also be downgraded



# Submission Guide

- ◆ Please submit to **course webpage at NTHU iLMS system**
  - *Notice: E-mail submission will not be accepted*
- ◆ Submission should include
  - Source codes (including solution and project files)
  - Executable binary (can be run on PC/windows)
  - Documentation (explain how you did it and how to operate it)
  - *Notice: please do not submit any 3D models to save the disk space*
- ◆ Contact with TAs if you have problem in submission

# Q&A

