

## Computer Graphics, Lab Assignment 7

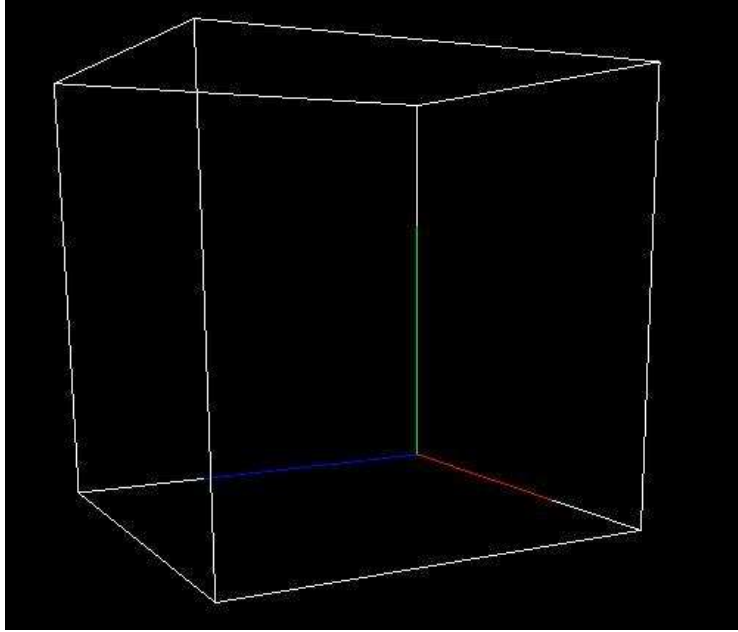
Handed out: April 18, 2021

**Due: 23:59, April 18, 2021 (NO SCORE for late submissions!)**

- Only files submitted by **git push to this course project** at <https://hconnect.hanyang.ac.kr> (<Year>\_<Course no.>\_<Class code>/<Year>\_<Course no.>\_<Student ID>.git) will be scored.
- Place your files under the directory structure **<Assignment name>/<Problem no.>/<your files>** just like the following example.

```
+ 2021_ITE0000_2019000001
+ LabAssignment2/
+ 1/
+   - 1.py
+ 2/
+   - 2.py
+ 3/
+   - 3.py
```

- The submission time is determined not when the commit is made **but when the git push is made**.
  - Your files must be committed to the **master branch**. Otherwise, it will not be scored.
1. Write down a Python program to draw following cube (정육면체) by using indexed "quad" representation and `glDrawElements()`.



- A.
  - B. Length of each line is 1.5
  - C. Start from the code in the lecture slides. Make sure camera manipulation shortcuts '1', '3', '2', 'w' work. (Don't need to care about initial view angle)
  - D. Set the window title to **your student ID** and the window size to (480,480).
  - E. Files to submit: A Python source file (Name the file whatever you want (in English). Extension should be .py)
2. Take pictures of three different objects with different diffuse and specular reflection properties, and generate surfaces having similar feeling with each picture using the following "Phong illumination demo".
    - A. <http://multivis.net/lecture/phong.html>
    - B. You can choose any object type to visualize each surface
    - C. It would be hard to simulate a real surface using Phong model. It's not your fault, it's probably from the limitation of the model. But try it anyway!
    - D. Zip the pictures of real objects & captured images of "Phong illumination demo" webpage
      - i. 1-real.xxx, 1-phong.xxx, 2-real.xxx, 2-phong.xxx
    - E. See the following example images.

1-real.jpg



2-real.jpg

...

1-phong.jpg

Phong Shading (WebGL)

Edit the shader code below and click on the button to see the result:

Phong Shading

Cube

normal mode

precision mediump

varying vec3 nor

varying vec3 ver

uniform int mode

uniform float Ka

uniform float Kd

uniform float Ks

uniform float sh

// Material colo

uniform vec3 amb

uniform vec3 dif

uniform vec3 spe

uniform vec3 lig

void main() {

vec3 N = norma

vec3 L = norma

// Lambert's c

float lambertI

float specular

Reload Shader Code

Ambient reflection coefficient (ka): 0.21

Ambient color:

Diffuse reflection coefficient (kd): 1.0

Diffuse Color:

Specular reflection coefficient (ks): 0

Specular Color:

Shininess: 29

Background Color:

Light position:

X: 1

Y: 1

Z: -1

2-phong.jpg

...