Computer Graphics Assignment 2

Handed out: April 2, 2018

Due date: 23:59, April 9, 2018 (NO SCORE for late submissions!)

Submit your assignment only through the Assignment (과제) menu of the lecture home at

portal.hanyang.ac.kr.

1. Create a hierarchical model using matrix stacks.

A. The model should consist of 3D primitives such as boxes and spheres,

You can use drawBox() and drawSphere() in the next page, or your own drawing functions

(which should use only numpy, opengl, glfw).

C. DO NOT use glut or glu functions to draw 3D primitives (e.g., glutSoildBox(), gluSphere(),...)

because they generate runtime crashes on some systems (maybe problems of some python

bindings, but don't use them anyway).

D. Because we've not covered *shading* yet, just draw your model in wireframe mode by

calling the following function at the beginning of your render function:

i. glPolygonMode(GL_FRONT_AND_BACK, GL_LINE) # call this at the beginning of your

render function

ii. You can change the color of your wireframe primitives using glColor*().

E. The model should have a hierarchy of at least three levels.

2. Animate the model to show the hierarchical structure.

A. Eg) a hand with fingers bending

B. Eg) a runner with arms and legs swing

C. The model should be animated without any mouse or keyboard inputs.

3. Your program should be able to run on systems with Python 3.5, NumPy, PyOpenGL,

PyOpenGL_accelerate, glfw. Do not use any other additional python modules.

4. What you have to submit:

A. **A single .py file** (named main.py)

```
# draw a cube of side 2, centered at the origin.
def drawCube():
   glBegin(GL QUADS)
   glVertex3f(1.0, 1.0, -1.0)
   glVertex3f(-1.0, 1.0, -1.0)
   glVertex3f(-1.0, 1.0, 1.0)
   glVertex3f( 1.0, 1.0, 1.0)
   glVertex3f(1.0,-1.0,1.0)
   glVertex3f(-1.0, -1.0, 1.0)
   glVertex3f(-1.0,-1.0,-1.0)
   glVertex3f( 1.0,-1.0,-1.0)
   glVertex3f( 1.0, 1.0, 1.0)
   glVertex3f(-1.0, 1.0, 1.0)
   glVertex3f(-1.0, -1.0, 1.0)
   glVertex3f( 1.0,-1.0, 1.0)
   glVertex3f(1.0,-1.0,-1.0)
   glVertex3f(-1.0,-1.0,-1.0)
   glVertex3f(-1.0, 1.0, -1.0)
   glVertex3f( 1.0, 1.0,-1.0)
   glVertex3f(-1.0, 1.0, 1.0)
   glVertex3f(-1.0, 1.0, -1.0)
   glVertex3f(-1.0,-1.0,-1.0)
   glVertex3f(-1.0,-1.0, 1.0)
   glVertex3f(1.0, 1.0, -1.0)
   glVertex3f( 1.0, 1.0, 1.0)
glVertex3f( 1.0, -1.0, 1.0)
   glVertex3f(1.0, -1.0, -1.0)
   qlEnd()
# draw a sphere of radius 1, centered at the origin.
# numLats: number of latitude segments (horizontal)
# numLongs: number of longitude segments (horizontal)
def drawSphere(numLats=12, numLongs=12):
   for i in range(0, numLats + 1):
      lat0 = np.pi * (-0.5 + float(float(i - 1) / float(numLats)))
      z0 = np.sin(lat0)
      zr0 = np.cos(lat0)
      lat1 = np.pi * (-0.5 + float(float(i) / float(numLats)))
      z1 = np.sin(lat1)
      zr1 = np.cos(lat1)
       # Use Quad strips to draw the sphere
      glBegin(GL_QUAD_STRIP)
      for j in range(0, numLongs + 1):
          lng = 2 * np.pi * float(float(j - 1) / float(numLongs))
```

```
x = np.cos(lng)
y = np.sin(lng)
glVertex3f(x * zr0, y * zr0, z0)
glVertex3f(x * zr1, y * zr1, z1)

glEnd()
```