Advanced Computer Graphics

Lecture-08 Introduction to OpenGL-4

Tzung-Han Lin
National Taiwan University of Science and Technology
Graduate Institute of Color and Illumination Technology
e-mail: thl@mail.ntust.edu.tw

TAIWAN TECH National Taiwan University of Science and Technology

CIT 色彩與照明科技研究所 Graduate Institute of Color and Illumination Technology

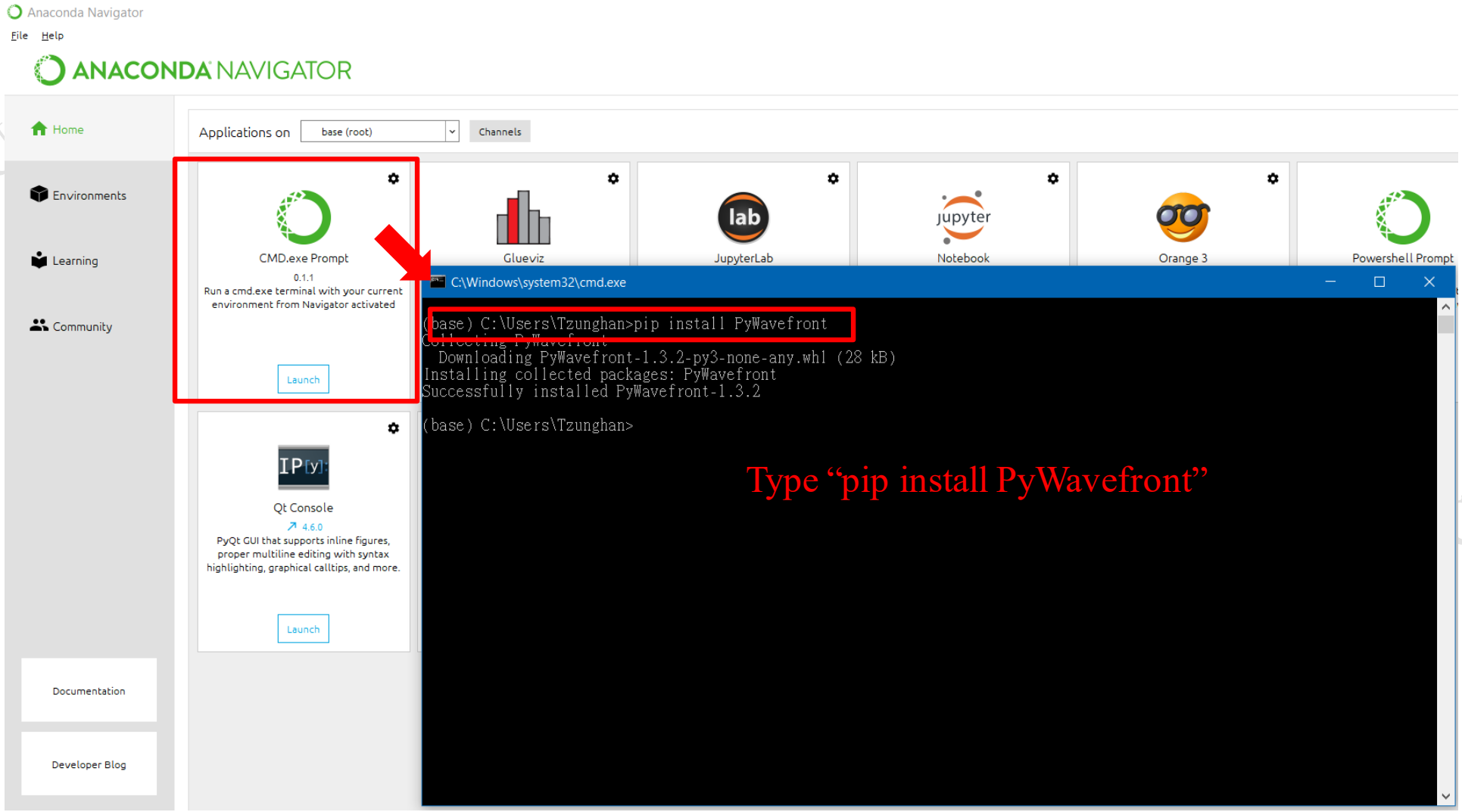
3D Wavefront File Parser

please install

PyWavefront & pyglet

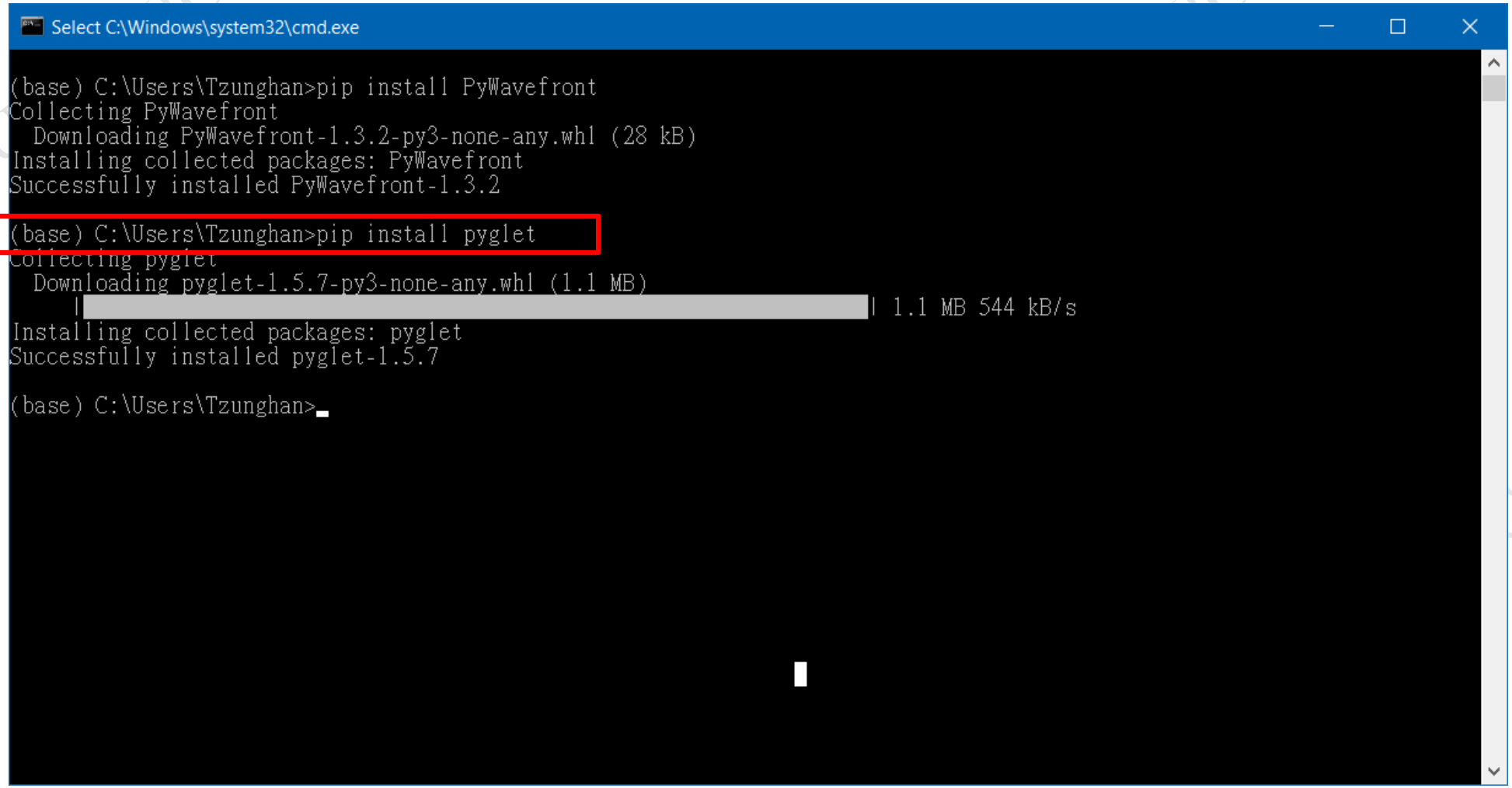
https://pypi.org/project/PyWavefront/

pip install PyWavefront



https://livebook.manning.com/book/math-for-programmers/a-loading-and-rendering-3d-models-with-opengl-and-pygame/v-8/

pip install pyglet



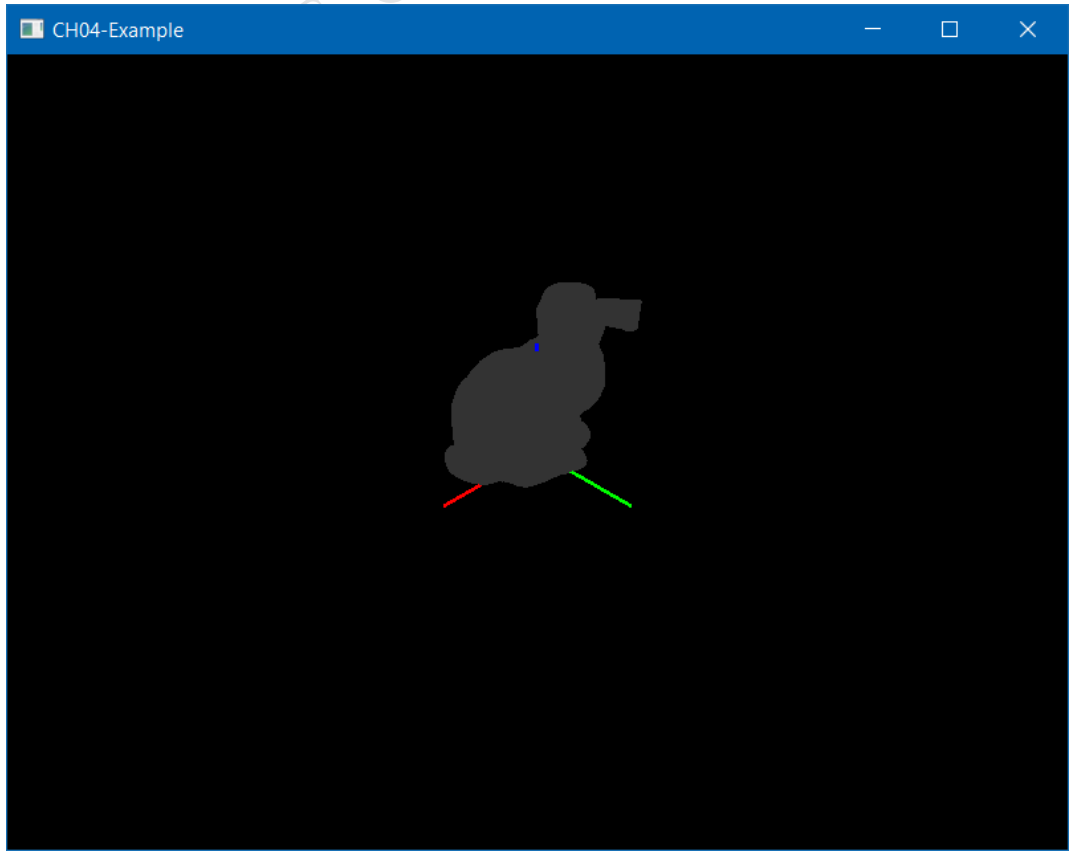
Load obj file by pywavefront (without vertex normals)

Load File

```
from pywavefront import visualization
import pywavefront

meshes = pywavefront.Wavefront('bunnyWoNormal.obj')

```

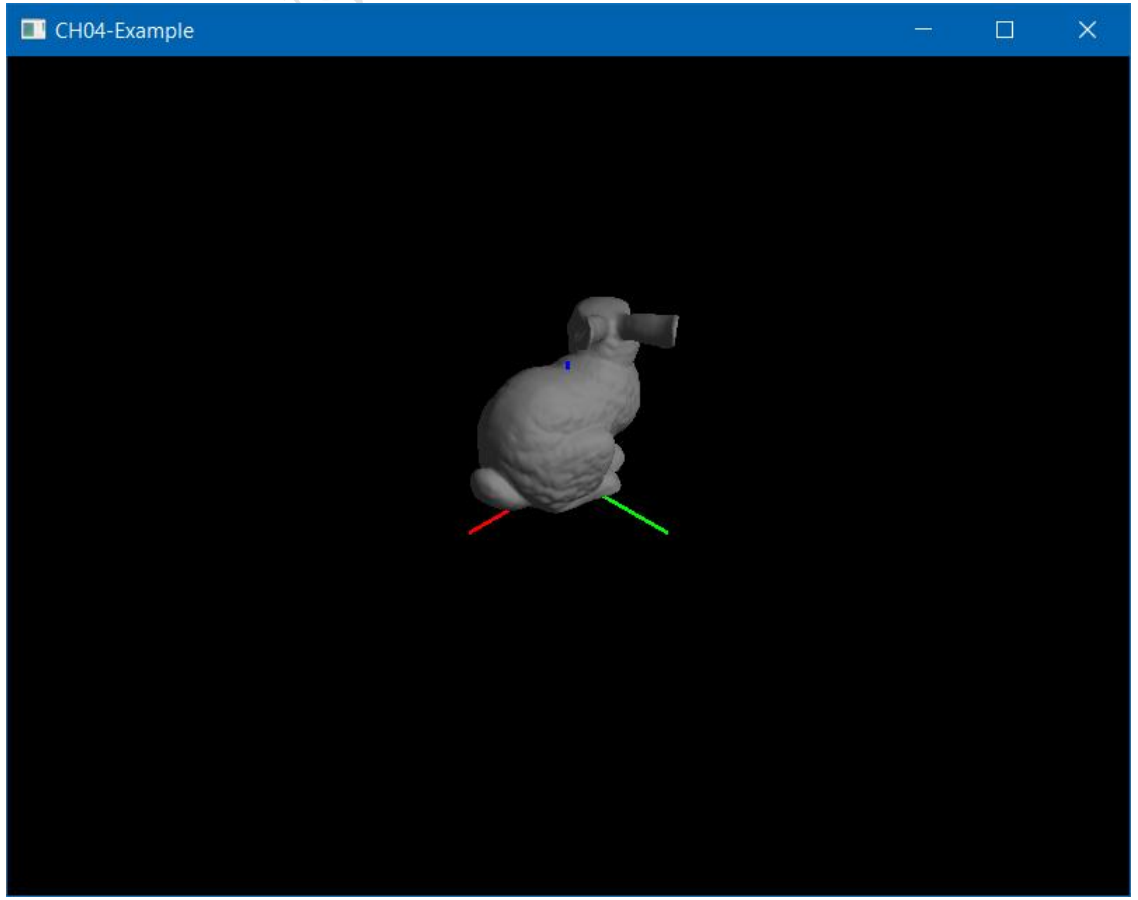
Draw File

```
def display():
    glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
    glMatrixMode(GL_PROJECTION)
    glLoadIdentity()
    glViewport(0, 0, windowWidth, windowHeight)
    glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowHeight)/
2.0,float(windowHeight)/2.0,-windowHeight*10.0,windowHeight*10.0)
    gluLookAt(1000,1000,1000,0,0,0,0,0,1)
    glEnable(GL_LIGHTING)
    visualization.draw(meshes)
    glDisable(GL_LIGHTING)
    drawCoordinate()
    glutSwapBuffers()
```



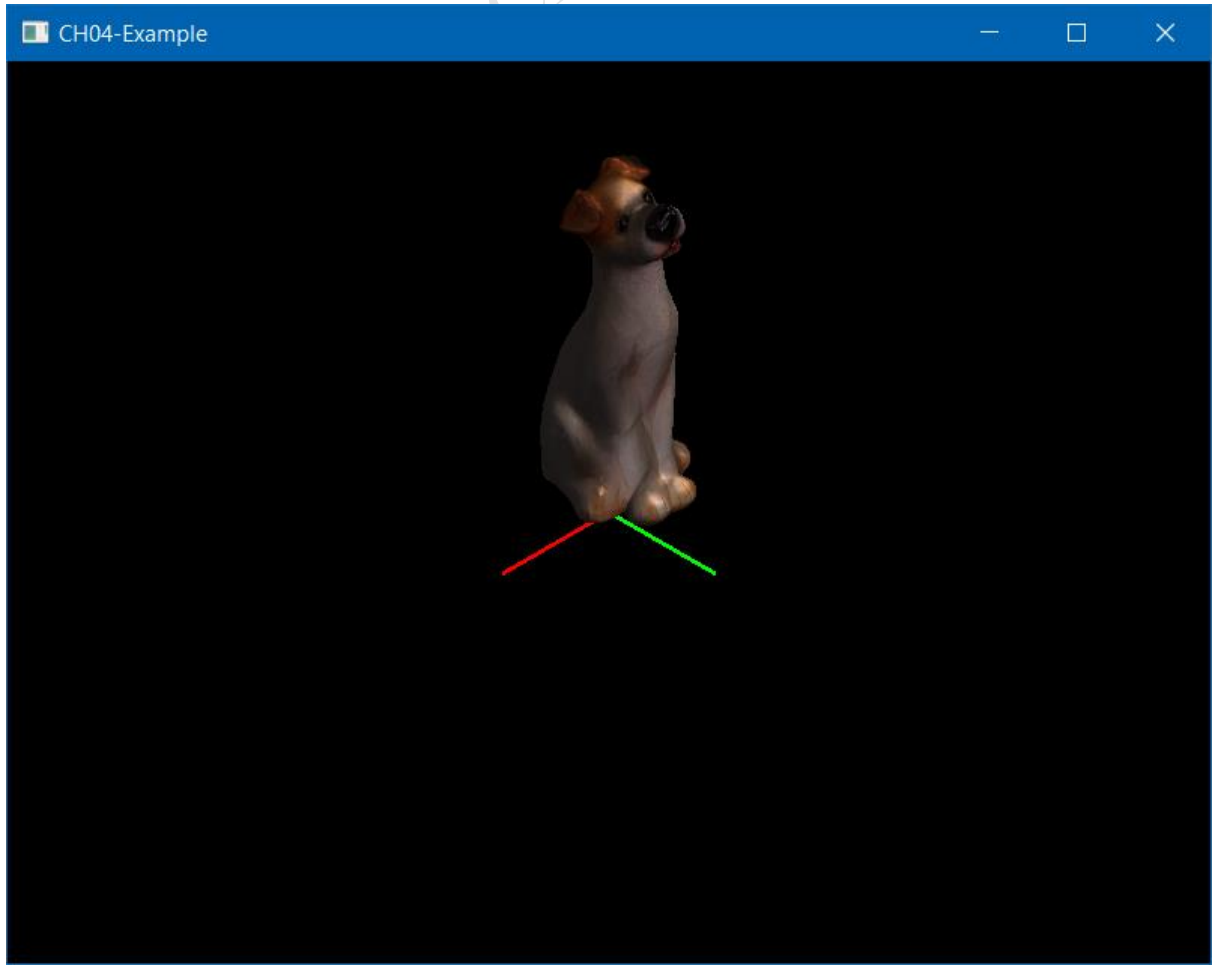
Load obj file by pywavefront (with vertex normals)

```
from pywavefront import visualization
import pywavefront

meshes = pywavefront.Wavefront('bunnyWNormal.obj')

theda = 0
angle = 0

windowWidth = 800
windowHeight = 600


def drawCoordinate():
    glLineWidth(3)
    glBegin(GL_LINES)
    glColor3f(1,0,0)
    glVertex3f(0,0,0)
    glVertex3f(100,0,0)
    glColor3f(0,1,0)
    glVertex3f(0,0,0)
    glVertex3f(0,100,0)
    glColor3f(0,0,1)
    glVertex3f(0,0,0)
    glVertex3f(0,0,100)
    glEnd()

def display():
    glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
    glMatrixMode(GL_PROJECTION)
    glLoadIdentity()
    glViewport(0, 0, windowWidth, windowHeight)
    glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowHeight)/2.0,float(windowHei
windowHeight*10.0,windowHeight*10.0)
    gluLookAt(1000,1000,1000,0,0,0,0,0,1)
    glEnable(GL_LIGHTING)
    visualization.draw(meshes)
    glDisable(GL_LIGHTING)
    drawCoordinate()
    glutSwapBuffers()
```



Load obj file by pywavefront (with texture)

```
8   from pywavefront import visualization
9   import pywavefront
10
11  meshes = pywavefront.Wavefront('Dog.obj')
12
13  theda = 0
14  angle = 0
15
16  windowWidth = 800
17  windowHeight = 600
18
19
20  def drawCoordinate():
21      glLineWidth(3)
22      glBegin(GL_LINES)
23      glColor3f(1,0,0)
24      glVertex3f(0,0,0)
25      glVertex3f(100,0,0)
26      glColor3f(0,1,0)
27      glVertex3f(0,0,0)
28      glVertex3f(0,100,0)
29      glColor3f(0,0,1)
30      glVertex3f(0,0,0)
31      glVertex3f(0,0,100)
32      glEnd()
33
34  def display():
35      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
36      glMatrixMode(GL_PROJECTION)
37      glLoadIdentity()
38      glViewport(0, 0, windowWidth, windowHeight)
39      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowHeigh
    windowHeight*10.0,windowHeight*10.0)
40      gluLookAt(1000,1000,1000,0,0,0,0,0,1)
41      glEnable(GL_LIGHTING)
42      visualization.draw(meshes)
43      glDisable(GL_LIGHTING)
44      drawCoordinate()
45      glutSwapBuffers()
46
```

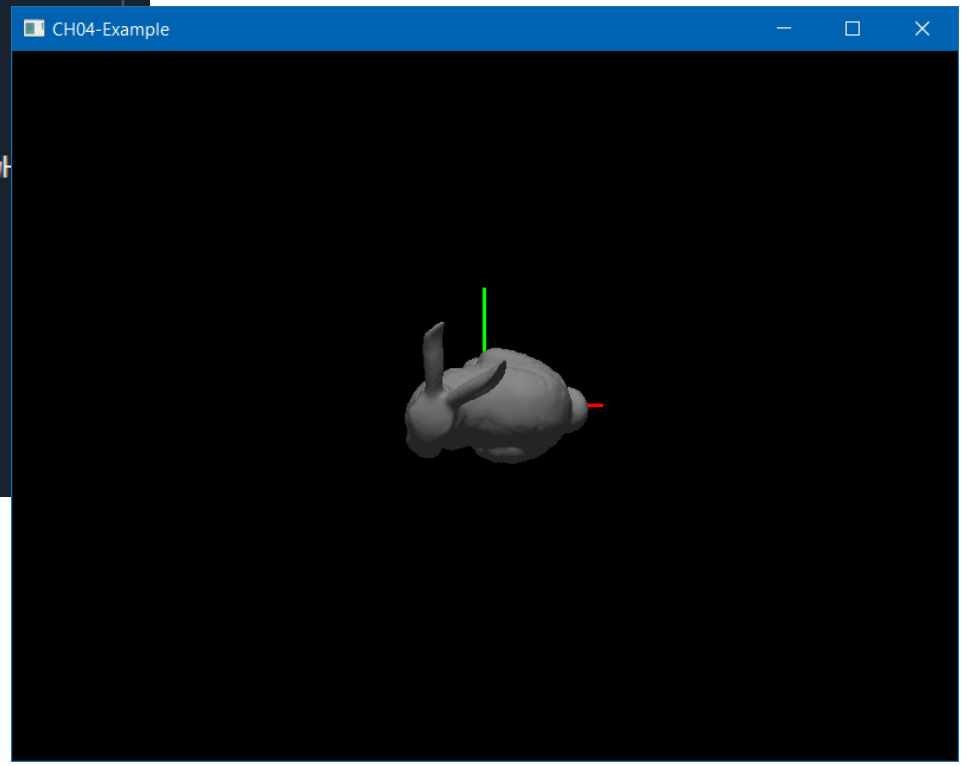


Light Position

- Deal with “GL_PROJECTION” & “GL_MODELVIEW” carefully

```
34  def display():
35      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
36      glMatrixMode(GL_PROJECTION)
37      glLoadIdentity()
38      glViewport(0, 0, windowWidth, windowHeight)
39      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowH
    windowHeight*10.0,windowHeight*10.0)
40      gluLookAt(0,0,1000,0,0,0,0,1,0)
41      glEnable(GL_LIGHTING)
42      visualization.draw(meshes)
43      glDisable(GL_LIGHTING)
44      drawCoordinate()
45      glutSwapBuffers()
46
```

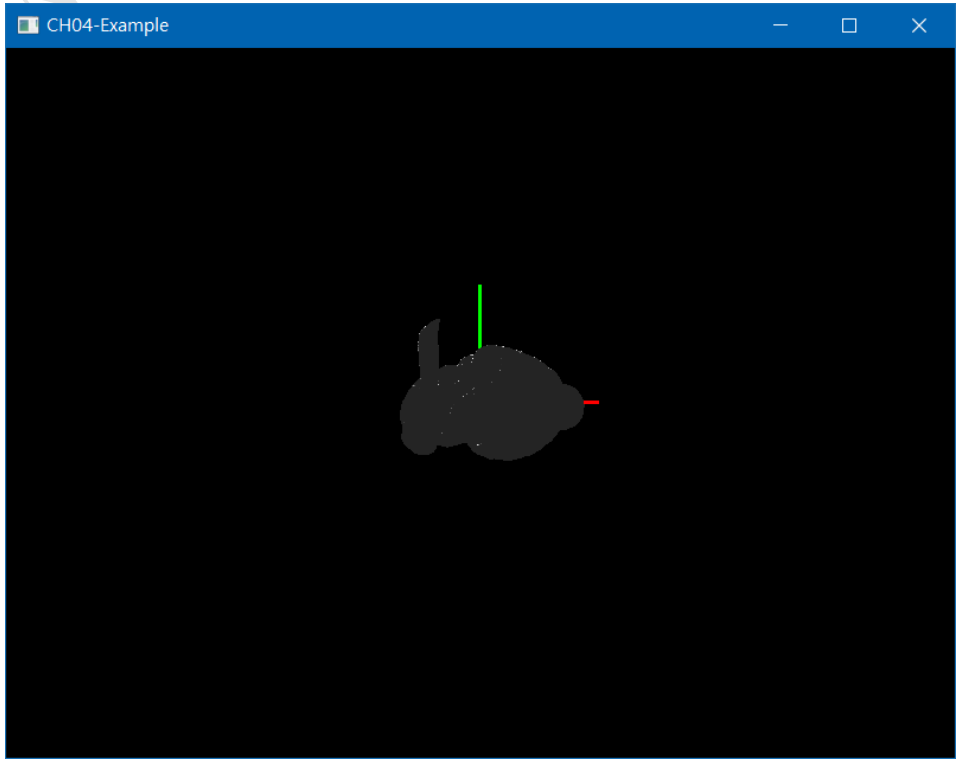
```
66  lightAmbient = [ 0.5,0.5,0.5,1.0 ]
67  lightDiffuse = [ 0.9,0.9,0.9,1.0 ]
68  lightSpecular = [ 1.0,1.0,1.0, 1.0 ]
69  lightPosition = [ 0,1000,1000,1.0 ]
70  glLightfv(GL_LIGHT0, GL_AMBIENT, lightAmbient)
71  glLightfv(GL_LIGHT0, GL_DIFFUSE, lightAmbient)
72  glLightfv(GL_LIGHT0, GL_SPECULAR, lightSpecular)
73  glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
```



Light Position

- Deal with “GL_PROJECTION” & “GL_MODELVIEW” carefully

```
39  def display():
40      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
41      glMatrixMode(GL_MODELVIEW)
42      glLoadIdentity()
43      glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
44      glViewport(0, 0, windowWidth, windowHeight)
45      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(wi
46      gluLookAt(0,0,1000,0,0,0,0,1,0)
47      glEnable(GL_LIGHTING)
48      visualization.draw(meshes)
49      glDisable(GL_LIGHTING)
50      drawCoordinate()
51      glutSwapBuffers()
52
```

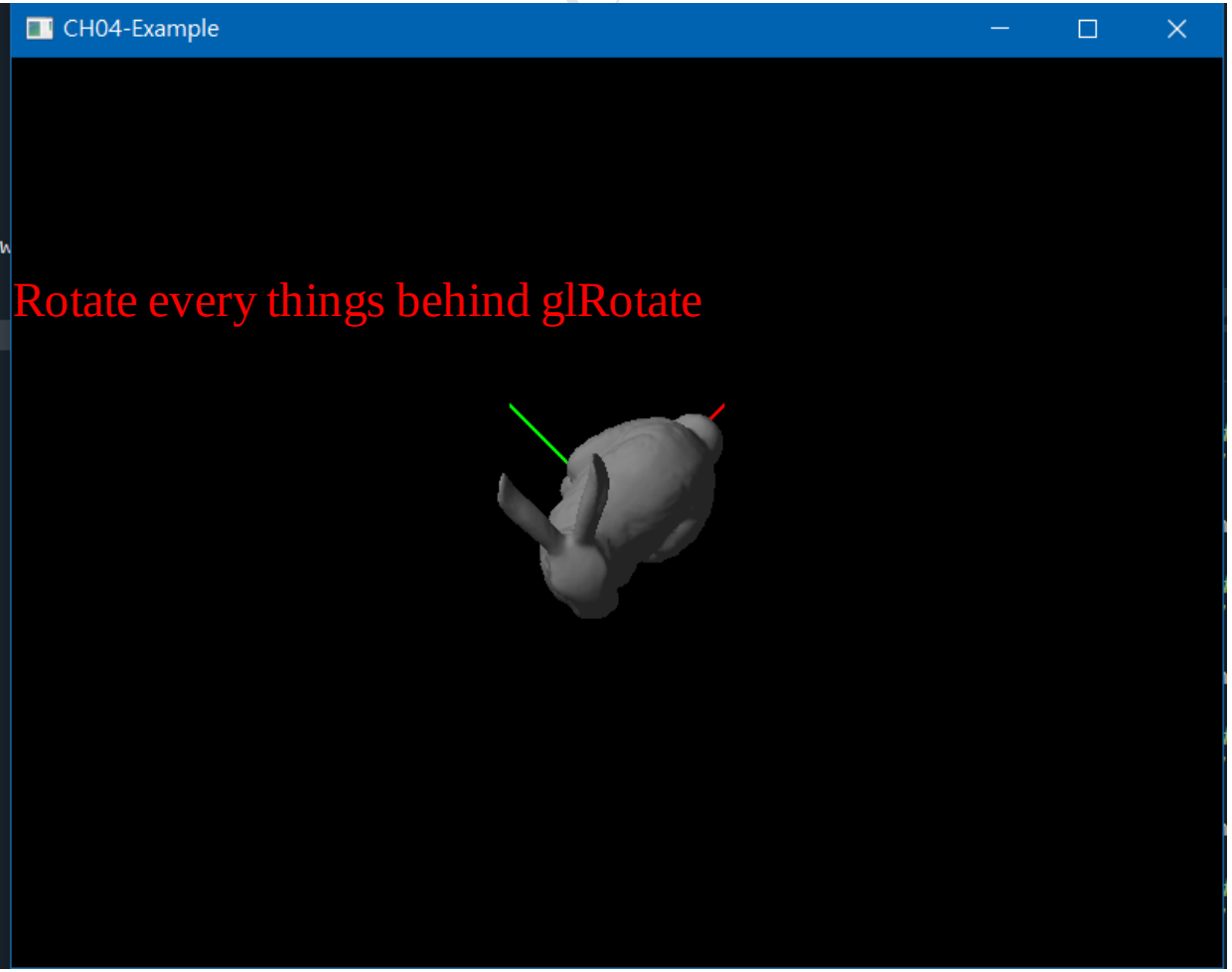


“GL_PROJECTION” & “GL_MODELVIEW”

- The difference between glMatrixMode(GL_MODELVIEW) and glMatrixMode(GL_PROJECTION) is simply that you’re operating on either the ModelView or the Projection stack, depending on which of the above function calls was made.

Rotate objects

```
38
39  def display():
40      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
41      glMatrixMode(GL_PROJECTION)
42      glLoadIdentity()
43      glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
44      glViewport(0, 0, windowWidth, windowHeight)
45      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(window
46      gluLookAt(0,0,1000,0,0,0,0,1,0)
47      glEnable(GL_LIGHTING)
48      glRotate(45,0,0,1)
49      visualization.draw(meshes)
50      glDisable(GL_LIGHTING)
51      drawCoordinate()
52      glutSwapBuffers()
53
54
55  def reshape(width,height):
56      glViewport(0, 0, width, height)
57
58  def keyboard( key, x, y ):
59      if key == esc:
60          sys.exit()
61
62
63  glutInit()
64  glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA)
65  glutCreateWindow(b'CH04-Example')
66  glutReshapeWindow(windowWidth,windowHeight)
67  glutReshapeFunc(reshape)
68  glutDisplayFunc(display)
```

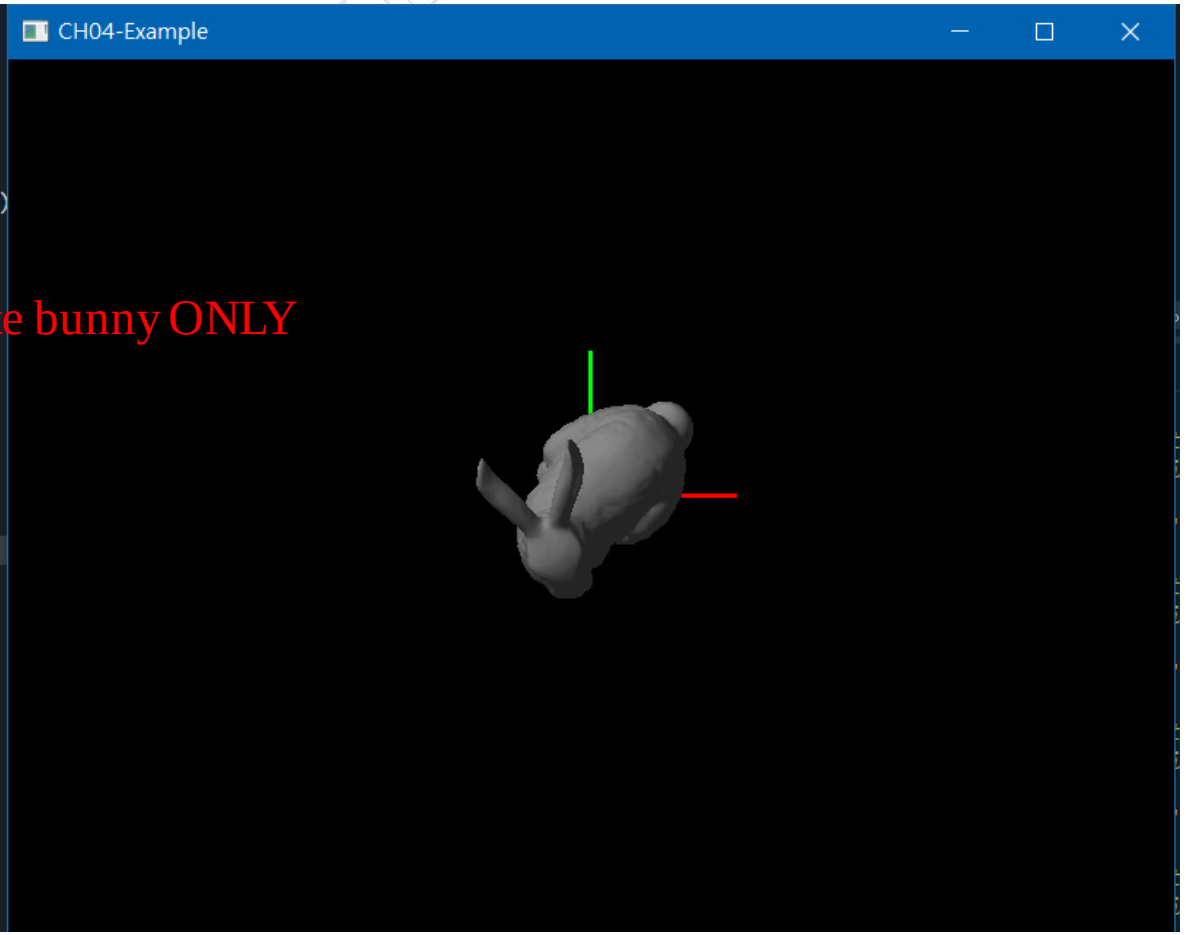


Rotate objects (with glPushMatrix & glPopMatrix)

- Note: glPushMatrix and glPopMatrix should exist simultaneously

```
39  def display():
40      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
41      glMatrixMode(GL_PROJECTION)
42      glLoadIdentity()
43      glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
44      glViewport(0, 0, windowWidth, windowHeight)
45      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowHeight)
46      gluLookAt(0,0,1000,0,0,0,0,1,0)
47      glEnable(GL_LIGHTING)
48      glPushMatrix()
49      glRotate(45,0,0,1)
50      visualization.draw(meshes)
51      glPopMatrix()
52      glDisable(GL_LIGHTING)
53      drawCoordinate()
54      glutSwapBuffers()
55
56
57  def reshape(width,height):
58      glViewport(0, 0, width, height)
59
60  def keyboard( key, x, y ):
61      if key == esc:
62          sys.exit()
63
64
65  glutInit()
66  glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA)
67  glutCreateWindow(b'CH04-Example')
68  glutReshapeWindow(windowWidth,windowHeight)
69  glutReshapeFunc(reshape)
70  glutDisplayFunc(display)
```

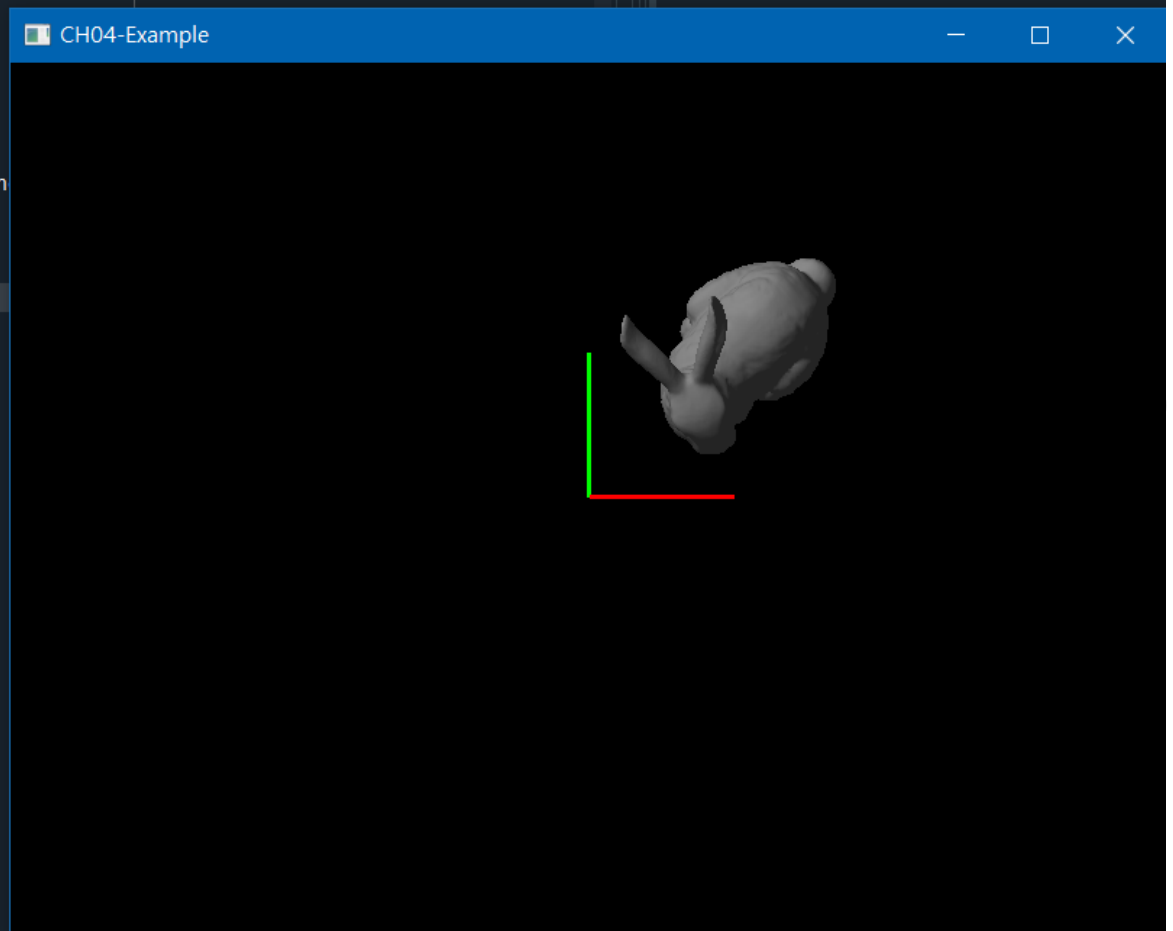
Rotate bunny ONLY



Rotate and Translate objects

- Rotate then Translate

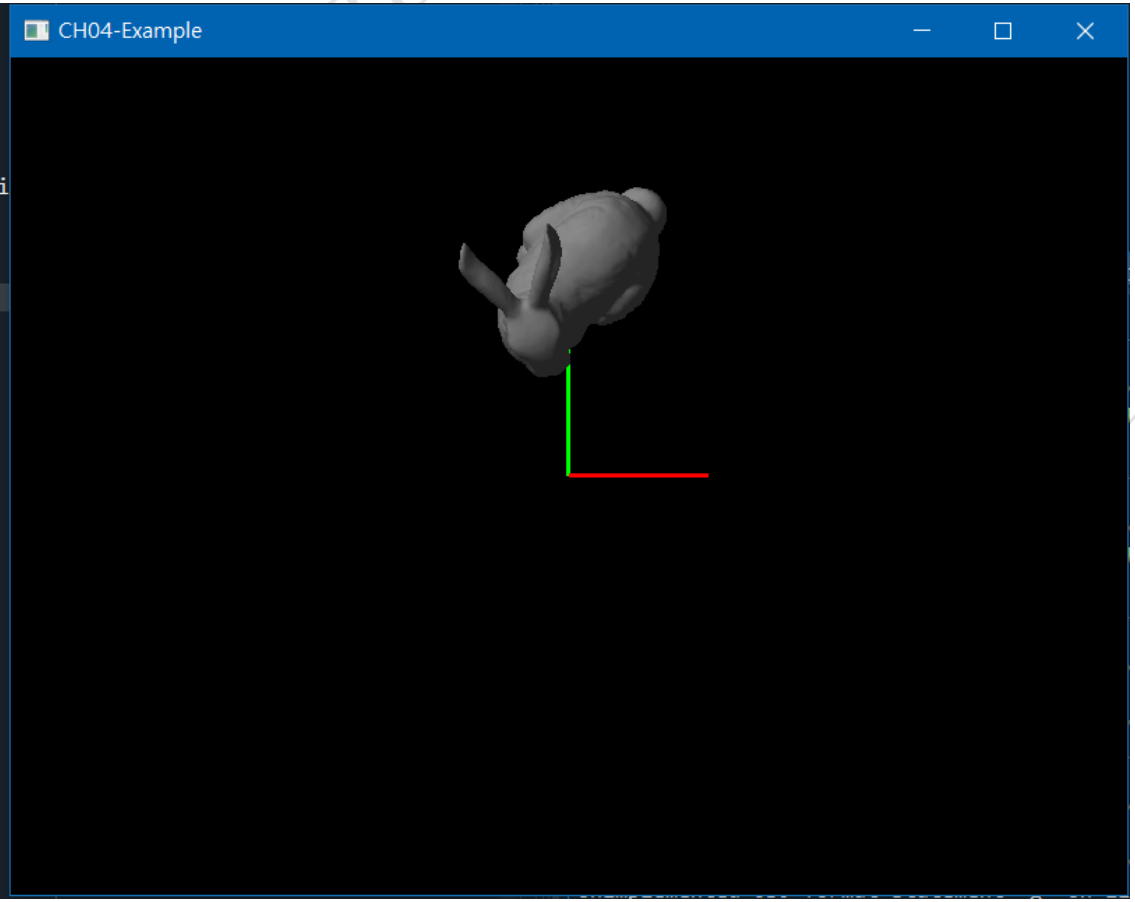
```
def display():
    glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
    glMatrixMode(GL_PROJECTION)
    glLoadIdentity()
    glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
    glViewport(0, 0, windowWidth, windowHeight)
    glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(win
    gluLookAt(0,0,1000,0,0,0,0,1,0)
    glEnable(GL_LIGHTING)
    glPushMatrix()
    glTranslate(100,100,0)
    glRotate(45,0,0,1)
    visualization.draw(meshes)
    glPopMatrix()
    glDisable(GL_LIGHTING)
    drawCoordinate()
    glutSwapBuffers()


def reshape(width,height):
    glViewport(0, 0, width, height)

def keyboard( key, x, y ):
    if key == esc:
        sys.exit()


glutInit()
glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA)
glutCreateWindow(b'CH04-Example')
glutReshapeWindow(windowWidth,windowHeight)
glutReshapeFunc(reshape)
```



Rotate and Translate objects

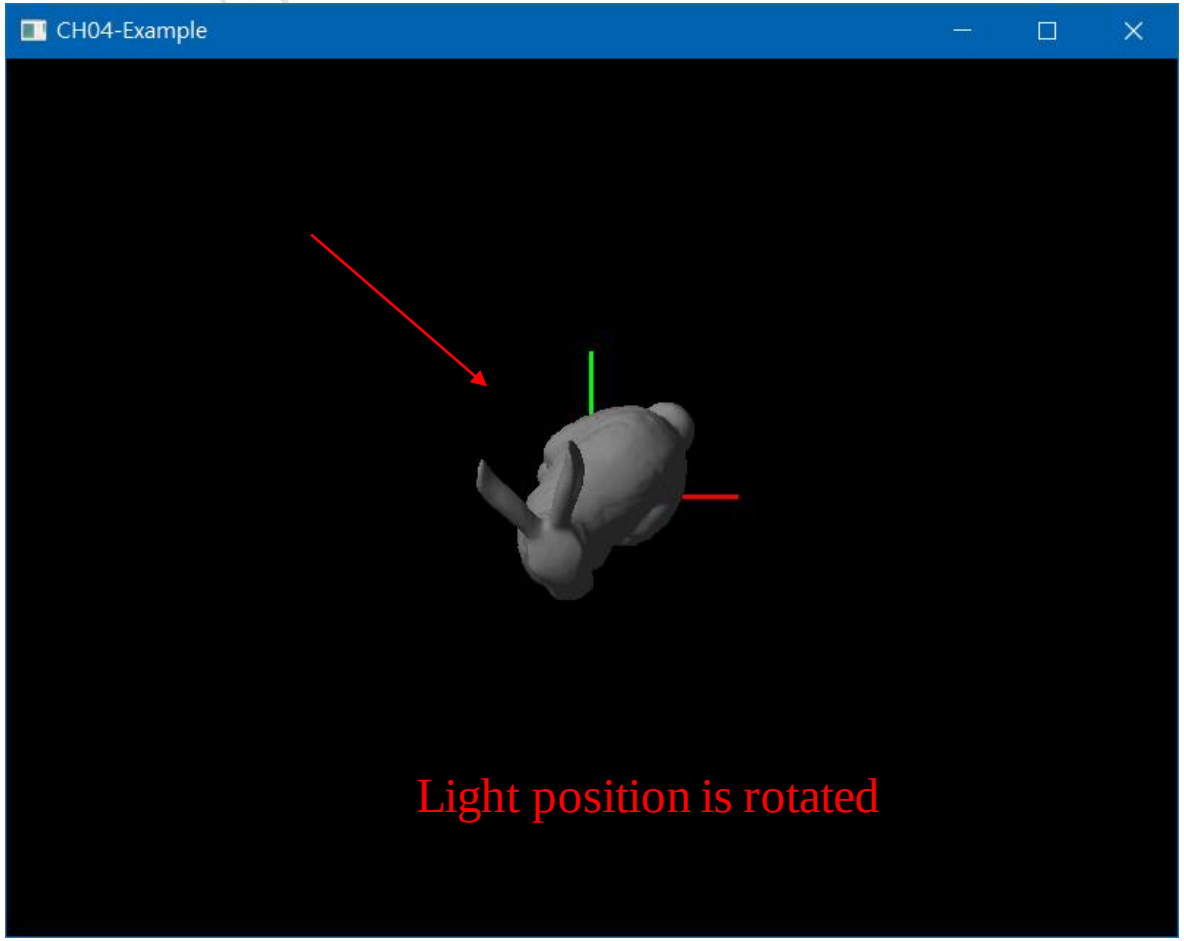
- Translate then Rotate

```
39  def display():
40      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
41      glMatrixMode(GL_PROJECTION)
42      glLoadIdentity()
43      glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
44      glViewport(0, 0, windowWidth, windowHeight)
45      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowHei
46      gluLookAt(0,0,1000,0,0,0,0,1,0)
47      glEnable(GL_LIGHTING)
48      glPushMatrix()
49      glRotate(45,0,0,1)
50      glTranslate(100,100,0)
51      visualization.draw(meshes)
52      glPopMatrix()
53      glDisable(GL_LIGHTING)
54      drawCoordinate()
55      glutSwapBuffers()
56
57
58  def reshape(width,height):
59      glViewport(0, 0, width, height)
60
61  def keyboard( key, x, y ):
62      if key == esc:
63          sys.exit()
64
65
66  glutInit()
67  glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA)
68  glutCreateWindow(b'CH04-Example')
69  glutReshapeWindow(windowWidth,windowHeight)
70  glutReshapeFunc(reshape)
```

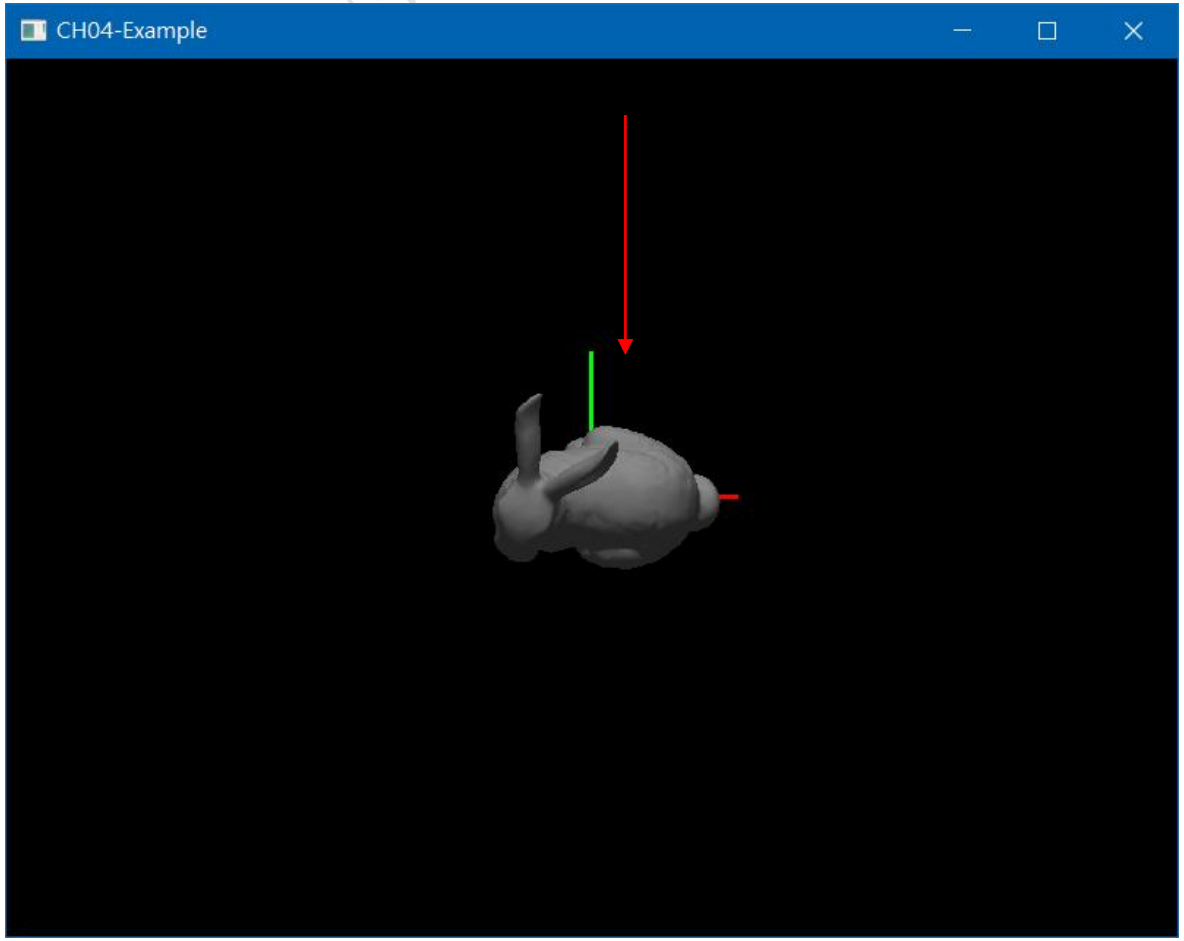


Light Position (Note: lighting direction)

P12



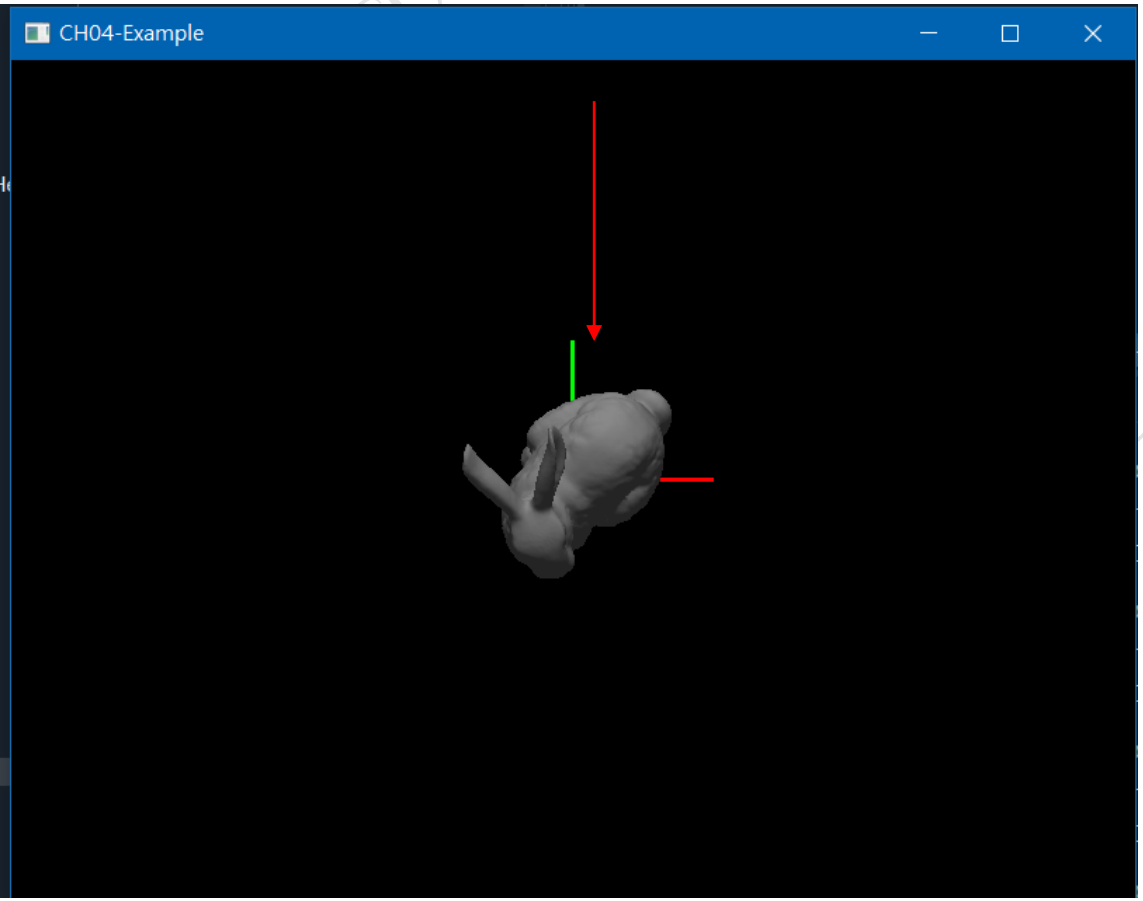
P8



Light Position (Note: lighting direction)

- Add glMatrixMode(GL_MODELVIEW) before draw something

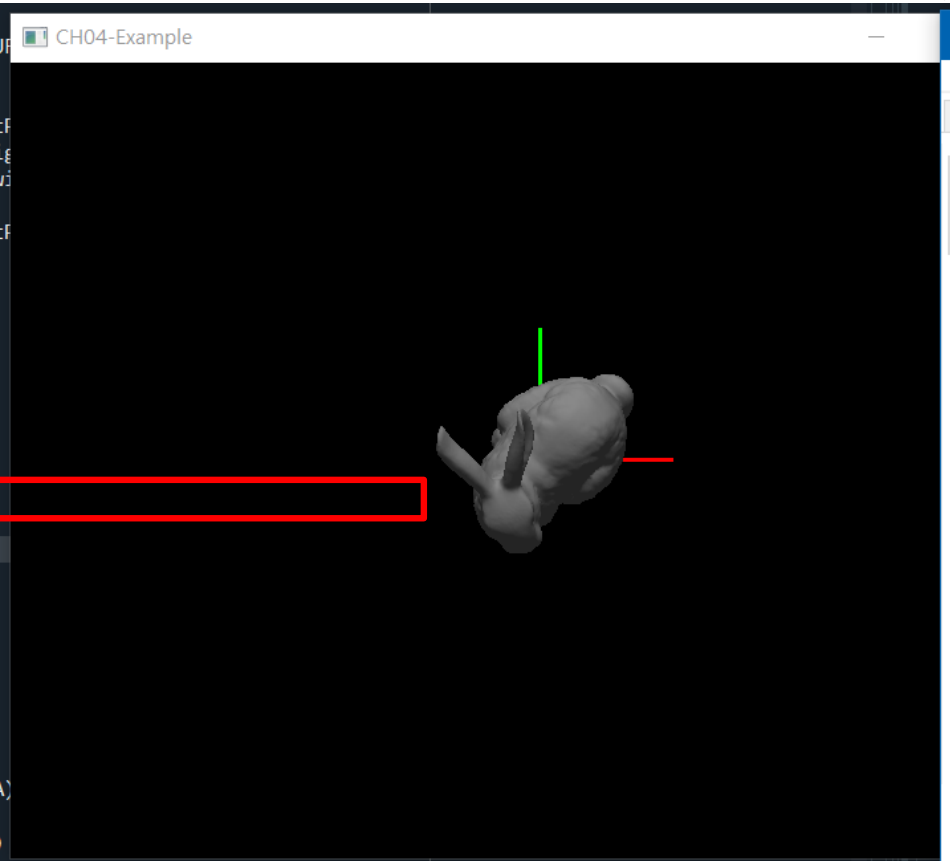
```
39  def display():
40      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
41      glMatrixMode(GL_PROJECTION)
42      glLoadIdentity()
43      glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
44      glViewport(0, 0, windowWidth, windowHeight)
45      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowH
46      gluLookAt(0,0,1000,0,0,0,0,1,0)
47      glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
48      glMatrixMode(GL_MODELVIEW)
49      glEnable(GL_LIGHTING)
50      glPushMatrix()
51      glRotate(45,0,0,1)
52      visualization.draw(meshes)
53      glPopMatrix()
54      glDisable(GL_LIGHTING)
55      drawCoordinate()
56      glutSwapBuffers()
57
58
59  def reshape(width,height):
60      glViewport(0, 0, width, height)
61
62  def keyboard( key, x, y ):
63      if key == esc:
64          sys.exit()
65
66
67  glutInit()
68  glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA)
69  glutCreateWindow(b'CH04-Example')
70  glutReshapeWindow(windowWidth,windowHeight)
```

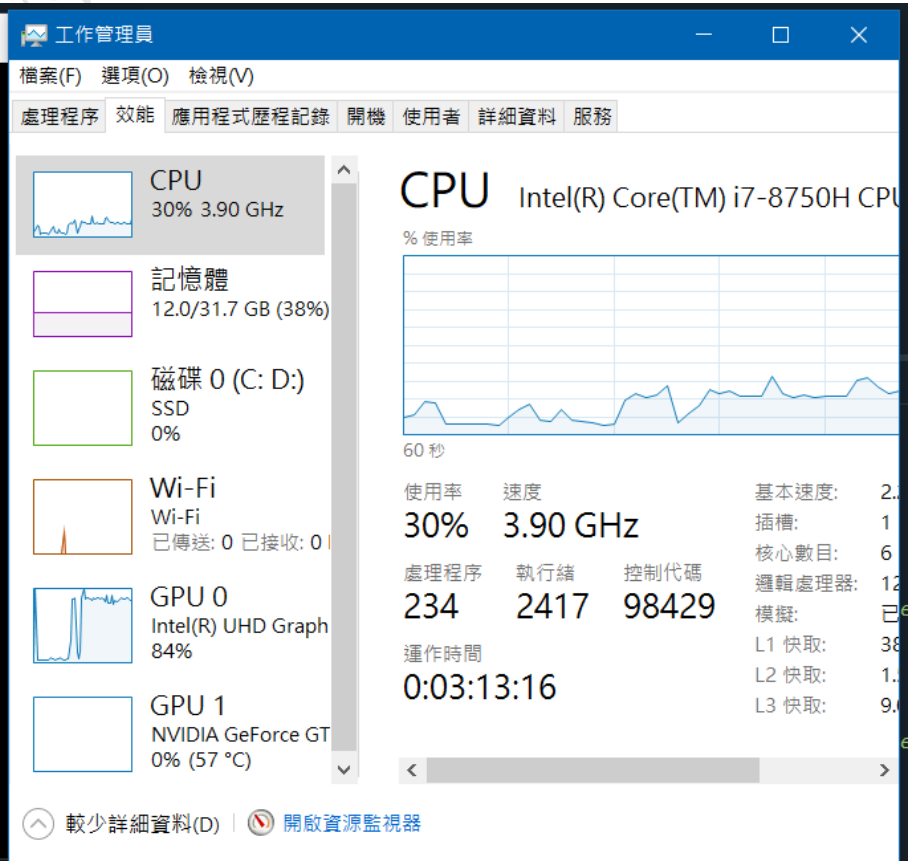


Animation with glutPostRedisplay

- glutPostRedisplay is an infinite loop (draw again and again)

```
39  def display():
40      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUF
41      glMatrixMode(GL_PROJECTION)
42      glLoadIdentity()
43      glLightfv(GL_LIGHT0, GL_POSITION, lightP
44      glViewport(0, 0, windowWidth, windowHeig
45      glOrtho(-float(windowWidth)/2.0,float(wi
46      gluLookAt(0,0,1000,0,0,0,0,1,0)
47      glLightfv(GL_LIGHT0, GL_POSITION, lightP
48      glMatrixMode(GL_MODELVIEW)
49      glEnable(GL_LIGHTING)
50      glPushMatrix()
51      glRotate(45,0,0,1)
52      visualization.draw(meshes)
53      glPopMatrix()
54      glDisable(GL_LIGHTING)
55      drawCoordinate()
56      glutSwapBuffers()
57      glutPostRedisplay()
58
59  def reshape(width,height):
60      glViewport(0, 0, width, height)
61
62  def keyboard( key, x, y ):
63      if key == esc:
64          sys.exit()
65
66
67  glutInit()
68  glutInitDisplayMode(GLUT_DOUBLE | GLUT_RGBA)
69  glutCreateWindow(b'CH04-Example')
70  glutReshapeWindow(windowWidth,windowHeight)
```

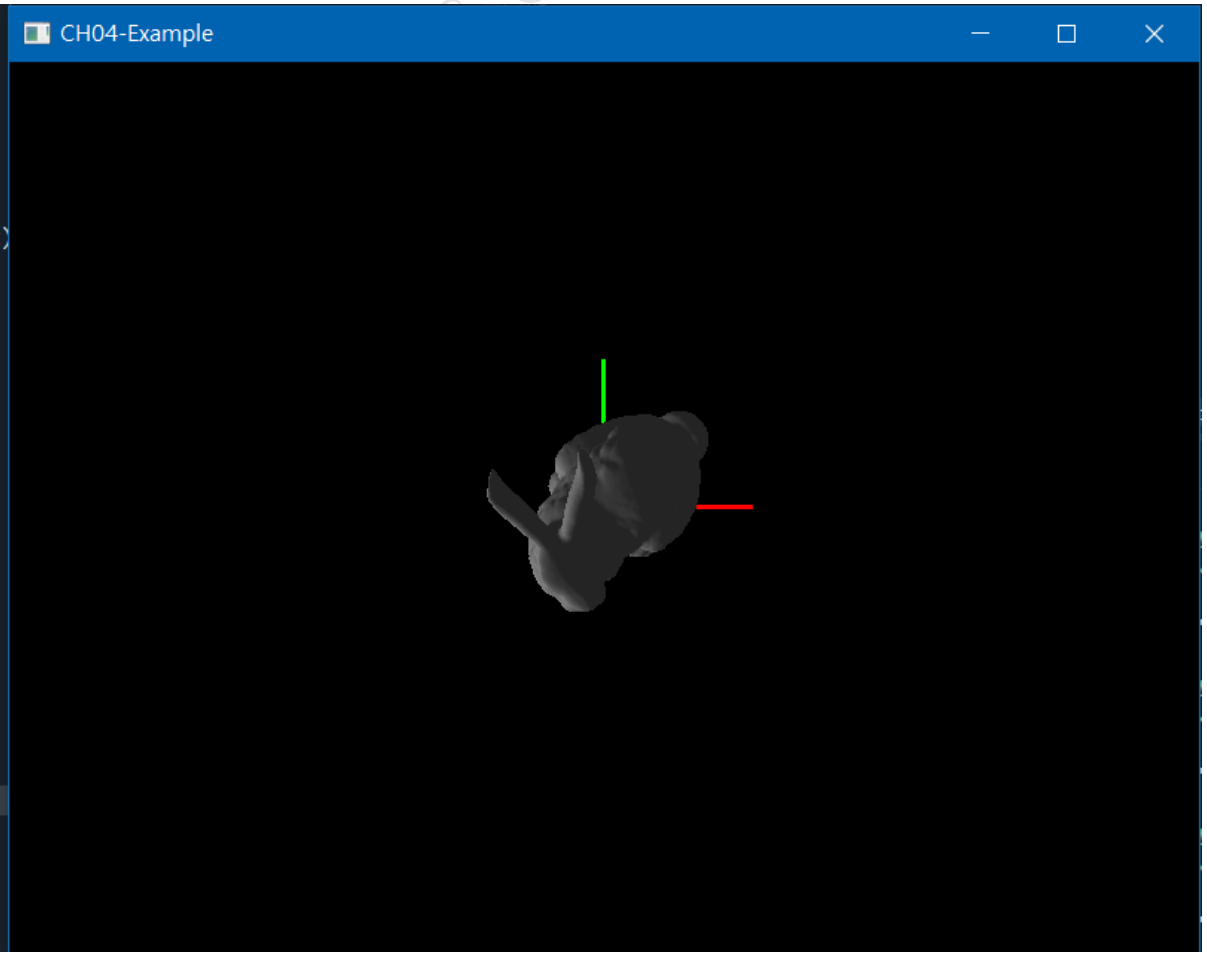




Your CPU will be busy, though you are draw a static scene

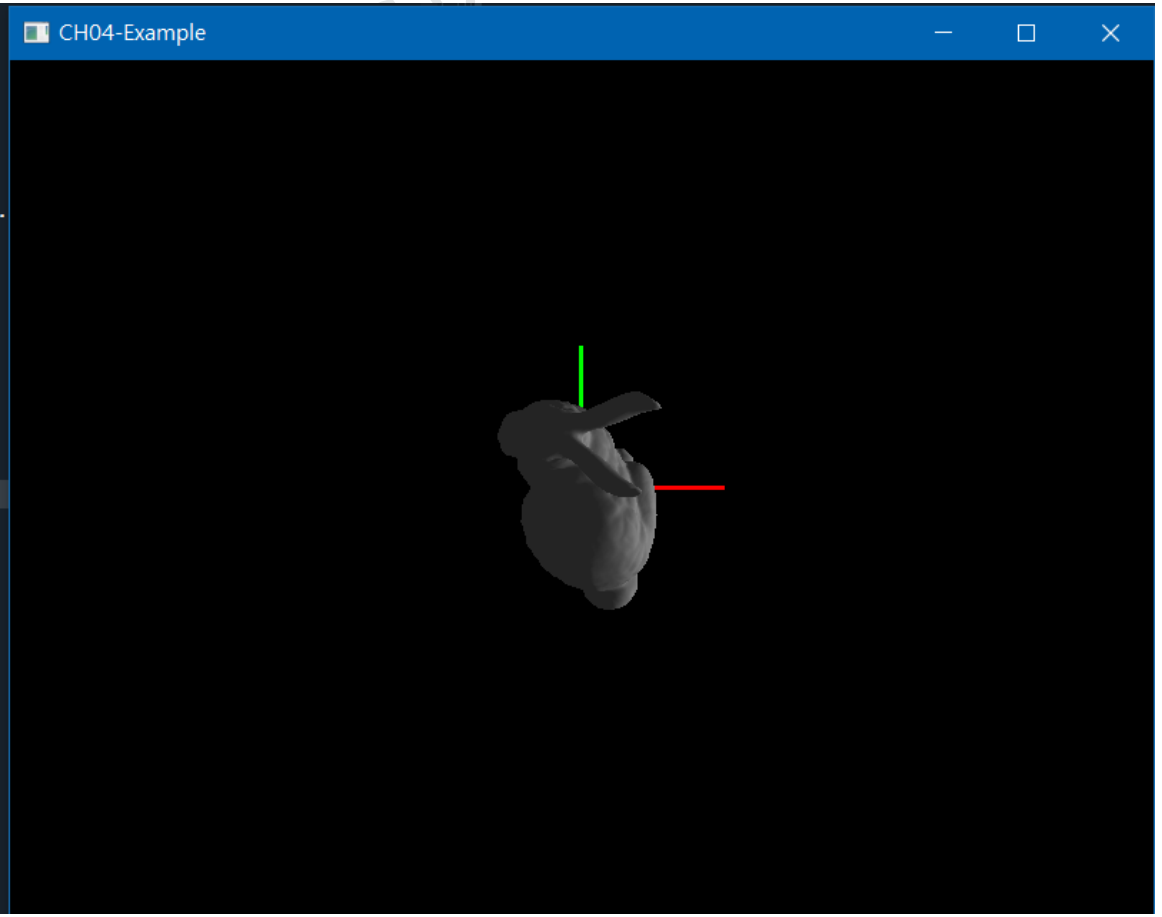
Rotating Light

```
38
39  def display():
40      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
41      glMatrixMode(GL_PROJECTION)
42      glLoadIdentity()
43      glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
44      glViewport(0, 0, windowWidth, windowHeight)
45      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowHeight)
46      gluLookAt(0,0,1000,0,0,0,0,1,0)
47      global theda
48      theda = theda+0.1
49      lightPositionLocal =[1000*cos(theda/57.0),1000*sin(theda/57.0),0,1.0]
50      glLightfv(GL_LIGHT0, GL_POSITION, lightPositionLocal)
51      glMatrixMode(GL_MODELVIEW)
52      glEnable(GL_LIGHTING)
53      glPushMatrix()
54      glRotate(45,0,0,1)
55      visualization.draw(meshes)
56      glPopMatrix()
57      glDisable(GL_LIGHTING)
58      drawCoordinate()
59      glutSwapBuffers()
60      glutPostRedisplay()
61
62  def reshape(width,height):
63      glViewport(0, 0, width, height)
64
65  def keyboard( key, x, y ):
66      if key == esc:
67          sys.exit()
68
```



Rotating Light and Object

```
38
39  def display():
40      glClear(GL_COLOR_BUFFER_BIT|GL_DEPTH_BUFFER_BIT)
41      glMatrixMode(GL_PROJECTION)
42      glLoadIdentity()
43      glLightfv(GL_LIGHT0, GL_POSITION, lightPosition)
44      glViewport(0, 0, windowWidth, windowHeight)
45      glOrtho(-float(windowWidth)/2.0,float(windowWidth)/2.0,-float(windowHeight)/2.
46      gluLookAt(0,0,1000,0,0,0,0,1,0)
47      global theda
48      theda = theda+0.1
49      lightPositionLocal =[1000*cos(theda/57.0),1000*sin(theda/57.0),0,1.0]
50      glLightfv(GL_LIGHT0, GL_POSITION, lightPositionLocal)
51      glMatrixMode(GL_MODELVIEW)
52      glEnable(GL_LIGHTING)
53      glPushMatrix()
54      global angle
55      angle = angle-0.02
56      glRotate(angle,0,0,1)
57      visualization.draw(meshes)
58      glPopMatrix()
59      glDisable(GL_LIGHTING)
60      drawCoordinate()
61      glutSwapBuffers()
62      glutPostRedisplay()
63
64  def reshape(width,height):
65      glViewport(0, 0, width, height)
66
67  def keyboard( key, x, y ):
68      if key == esc:
69          sys.exit()
```



色彩與照明科技研究所
Graduate Institute of
Color and Illumination Technology

Find us on: facebook