Computer Graphics, Lab Assignment 4

Handed out: April 22, 2024

Due: 23:59, May 2nd, 2024 **(NO SCORE for late submissions!)**

Submit a single zip **[studentID]-[assignment#].**zip file containing one source file.

1. Write down a Python program to draw a hierarchical model of boxes.
   A. Set the window title to **[studentID]-[assignment#]-[prob#]** and the window size to (480,480).
   B. Start from the following code skeleton.

```
import glfw
from OpenGL.GL import *
import numpy as np
from OpenGL.GLU import *

def render():
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT)
    glEnable(GL_DEPTH_TEST)

    glMatrixMode(GL_PROJECTION)
    glLoadIdentity()
    glOrtho(-2,2, -2,2, -1,1)

    glMatrixMode(GL_MODELVIEW)
    glLoadIdentity()

    drawFrame()
    t = glfw.get_time()

    # blue base transformation
    glPushMatrix()
    glTranslatef(np.sin(t), 0, 0)

    # blue base drawing
    glPushMatrix()
    glScalef(.2, .2, .2)
    glColor3ub(0, 0, 255)
    drawBox()
    glPopMatrix()

    # red arm transformation
    glPushMatrix()
    glRotatef(t*(180/np.pi), 0, 0, 1)
    glTranslatef(.5, 0, .01)

    # red arm drawing
    glPushMatrix()
    glScalef(.5, .1, .1)
    glColor3ub(255, 0, 0)
    drawBox()
    glPopMatrix()

    glPopMatrix()
```

```
    glPopMatrix()

def drawBox():
    glBegin(GL_QUADS)
    glVertex3fv(np.array([1,1,0.]))
    glVertex3fv(np.array([-1,1,0.]))
    glVertex3fv(np.array([-1,-1,0.]))
    glVertex3fv(np.array([1,-1,0.]))
    glEnd()

def drawFrame():
    # draw coordinate: x in red, y in green, z in blue
    glBegin(GL_LINES)
    glColor3ub(255, 0, 0)
    glVertex3fv(np.array([0.,0.,0.]))
    glVertex3fv(np.array([1.,0.,0.]))
    glColor3ub(0, 255, 0)
    glVertex3fv(np.array([0.,0.,0.]))
    glVertex3fv(np.array([0.,1.,0.]))
    glColor3ub(0, 0, 255)
    glVertex3fv(np.array([0.,0.,0]))
    glVertex3fv(np.array([0.,0.,1.]))
    glEnd()

def main():
    if not glfw.init():
        return
    window = glfw.create_window(480,480,'2017123456-lab6-1', None,None)
    if not window:
        glfw.terminate()
        return
    glfw.make_context_current(window)
    glfw.swap_interval(1)

    while not glfw.window_should_close(window):
        glfw.poll_events()
        render()
        glfw.swap_buffers(window)

    glfw.terminate()

if __name__ == "__main__":
    main()
```

C. Add a green arm at the end of the red arm, and rotate the green arm about its local z axis.

D. Also render local frames of the blue base, red arm, green arm using drawFrame().

E. The blue base, red arm, green arm should be rendered using drawBox()**.**

F. Expected result: Uploaded as LabAssignment4.mp4

G. Include a single .py file in the zip file - **[studentID]-[assignment#]-[prob#].py**