Assignment 2

COMP 4002

Date: October 8 2014

Due: on October 24, 2014 before 22:00 (10:00 PM)
Submission: Electronic submission on CULearn.

Assignment Objectives:

a. Familiarization with transformation
b. Familiarization with object modelling
c. Familiarization with hierarchical object (scene)
d. Familiarization with viewing pipeline (view coordinates point and projections)
e. Familiarization with interaction (keyboard)
f. Familiarization with camera

Grades:

1. Assignment total marks: 100%.
   .

1. Task 1 – Render a sphere (40)

Purpose:

a. Create projection matrix
b. Create view coordinate transformation
c. View a simple 3D object.

To do:

1. Create a project and modify code.
   1.1. Create a sphere with a few sides and place it on the world space on the xz plane at 100, 10, 100.
   1.2. Set up a view position (e.g., 200,200,200) and set up the transformation matrix so that you can see the sphere.
   1.3. Set up perspective projection (e.g., symmetric).
   1.4. Display the sphere.

2. Task 2 – Create a simple hierarchical object (40)

Purpose:

a. Create hierarchical object
b. Use keyboard commands
c. Use transformations

To do:

2. Create a simple hierarchical object and place it beside the sphere from Task 1 (Figure 1).
   2.1. The object consists of a box and two spheres, where the spheres are placed on top of the box (Figure 1). Randomly set the three primary colours to the vertices (so that you can see the spheres rotating).
   2.2. One sphere should rotate counter clock wise around the axis (CCW) at a rate of 0.1^o per frame (about 3^o per second).
   2.3. The second sphere should rotate around the y-axis clock wise (CW) at a rate of 0.2^o per frame (about 6^o per second).
   2.4. The user can move the cylinder in the xz plane by using the i,j,k,l keyboard strokes –
      2.4.1. Pressing on i means translate the object by a small delta (e.g., 1) x along the x–axis = x –delta;
      2.4.2. Pressing on l means translate the object by a small delta (e.g., 1) along the x-axis x = x + delta;
      2.4.3. Pressing on j means translate the object by a small delta (e.g., 1) along the z-axis z = z –delta;
      2.4.4. Pressing on k means translate the object by a small delta (e.g., 1) along the z-axis z = z + delta;

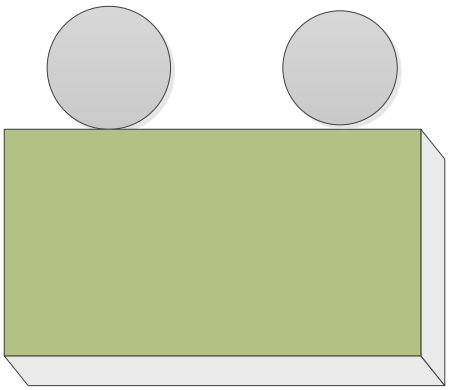

Figure 1: Simple hierarchical object

3. Task 3 – Create a camera class (40)

Purpose:

a. Manipulating a camera
b. Use keyboard commands
c. Use transformations

To do:

3. Create a camera class that will enable the user to manipulate the camera.
   3.1. Keep with the camera three variables: position, look at vector or (reference point), and up vector.
   3.2. The user can manipulate the camera as follows:
      3.2.1. Pitch operation – the up and down arrow keys will rotate the camera around the x-axis by +1 degree and -1 degree respectively.
      3.2.2. Yaw operation – the left and right arrow keys will rotate the camera around the y-axis by +1 degree and -1 degree respectively.
      3.2.3. Roll operation – the "a" and "d" keys will rotate the camera around the z-axis by +1 degree and -1 degree respectively.
      3.2.4. Forward motion – the "w" key will move the camera forward by 1 unit along the "look at" vector.
      3.2.5. Backward motion - the "s" key will move the camera backward by 1 unit along the "look at" vector.

   Hints:

   a. Roll, pitch and yaw
      a. Determine the rotation vector around which the camera should be rotated around
      b. Create a single function to rotate the up and look-at vectors around the rotation vector by a given angle
      c. Update the update the look-at and up vectors to ensure that they are orthogonal
      d. Normalize the look-at and up
   b. Motion – move the camera along the look-at vector
      3.2.6. Pressing on j means translate the object by a small delta (e.g., 1) along the z-axis z = z –delta;
      3.2.7. Pressing on l means translate the object by a small delta (e.g., 1) along the z-axis z = z + delta;

4. Bonus – Render a "robot arm" complex hierarchical object (25)

1. Create a simple "robot arm" (hierarchical object) as follows:
   1.1. The object consists of 3 cylinders and 2 spheres (Figure 2).
2. The user can identify each of the parts using the numeric keys 1,2,3,4,5. Figure 2 shows how the parts and their ids.
3. The user can manipulate the objects as follows (note all rotations are with respect to the model object):
   3.1. The user selects an object part using a numeric key 1-5
   3.2. Action for cylindrical parts (id – 1, 3, 5)
      3.2.1. Once a part is selected the user can rotate the object as follows: rotate the part using keys z and x around the y-axis (yaw rotation) by +1 or -1 degree, respectively.
   3.3. Action for spherical parts (id – 2, 4)
      3.3.1. Once a part is selected the user can rotate the object as follows: rotate the part using keys z and x around the x-axis (pitch rotation) by +1 or -1 degree, respectively.

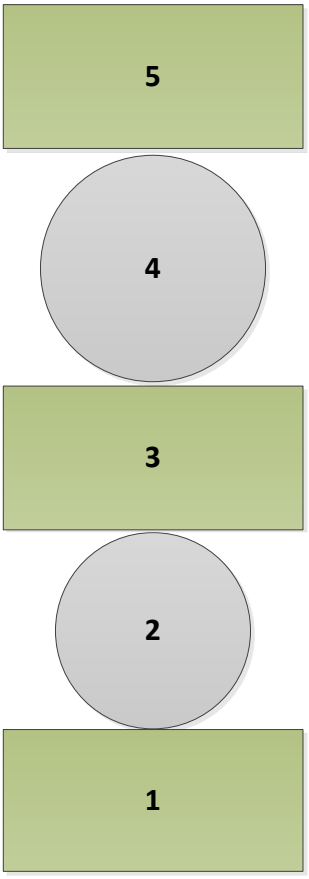

Figure 2: Robot arm manipulator.