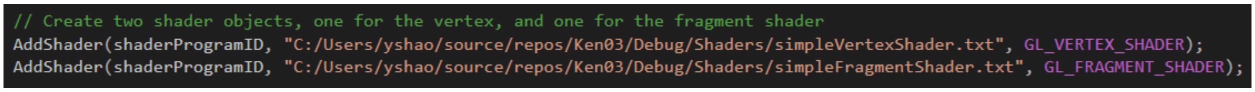
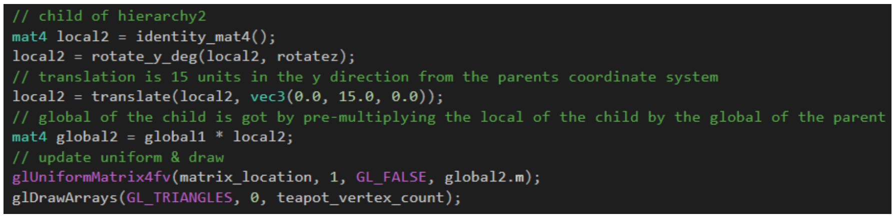
Pre-setting

Initially, the file was not able to execute at all. Therefore, I have downloaded simpleVertexShader and simpleFragmentShader, and I linked the location paths of these two text files to AddShader functions, the code is as below:



```
// Create two shader objects, one for the vertex, and one for the fragment shader
AddShader(shaderProgramID, "C:/Users/yshao/source/repos/Ken03/Debug/Shaders/simpleVertexShader.txt", GL_VERTEX_SHADER);
AddShader(shaderProgramID, "C:/Users/yshao/source/repos/Ken03/Debug/Shaders/simpleFragmentShader.txt", GL_FRAGMENT_SHADER);
```

In the child of hierarchy part: in order to implement the structure of at least 5 teapots. I duplicated the code into another 3 copy and pasted them below the duplicated code.



```
// child of hierarchy2
mat4 local2 = identity_mat4();
local2 = rotate_y_deg(local2, rotatez);
// translation is 15 units in the y direction from the parents coordinate system
local2 = translate(local2, vec3(0.0, 15.0, 0.0));
// global of the child is got by pre-multiplying the local of the child by the global of the parent
mat4 global2 = global1 * local2;
// update uniform & draw
glUniformMatrix4fv(matrix_location, 1, GL_FALSE, global2.m);
glDrawArrays(GL_TRIANGLES, 0, teapot_vertex_count);
```

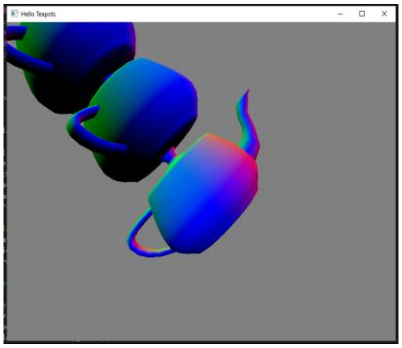
Moreover, the integer after local should be changed to the next integer respectively. For example, suppose it is the third teapot, it should be wrote as local3 and mat4 global3 = global2 * local3.Suppose it is the forth teapot, it should be wrote as local4 and mat4 global4 = global3 * local4; respectively.

After writing the code of another 3 chid of hierarchies, there are 5 teapots in stack totoally.

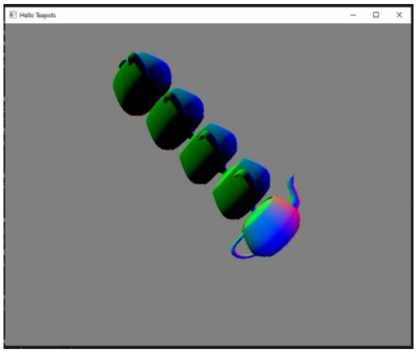
(globalN = global(N-1)+localN, N is the hierarchy level of the teapots)

However, after executing, I realized that the image is too big while displaying multiple teapots. To solve this problem, in the Root of hierarchy part of the display function, I wrote scale to reduce the size. I changed the vec3 vertex values from 1 to 0.5, seen as below:

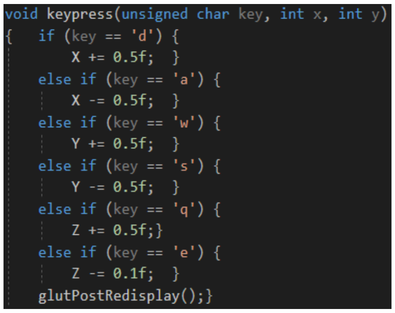
```
local1 = scale(local1, vec3(0.5, 0.5, 0.5f));
```



Before Scaling



After Scaling



```
void keypress(unsigned char key, int x, int y)
{   if (key == 'd') {
        X += 0.5f;  }
    else if (key == 'a') {
        X -= 0.5f;  }
    else if (key == 'w') {
        Y += 0.5f;  }
    else if (key == 's') {
        Y -= 0.5f;  }
    else if (key == 'q') {
        Z += 0.5f;}
    else if (key == 'e') {
        Z -= 0.1f;  }
    glutPostRedisplay();}
```

Keyboard control code

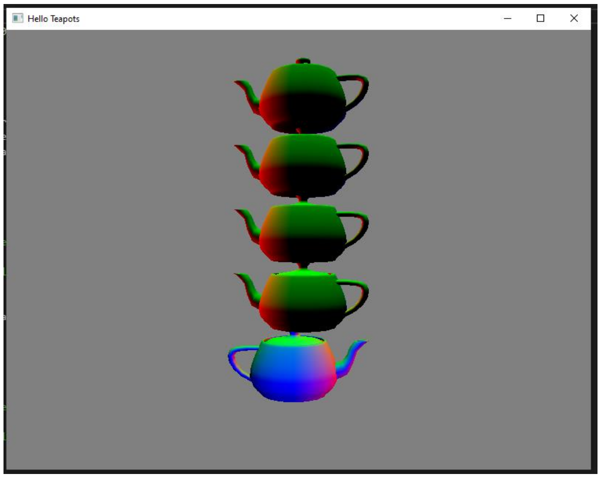
Furthermore, I want to shift the entire structure from 45 degree incline, as present, into vertical to make it looks more decent. Therefore, in display function, I changed the rotate value from 45 to 0, i.e. *local1 = rotate_z_deg(local1, 45.0f);*

**Keyboard control (Code as above)**

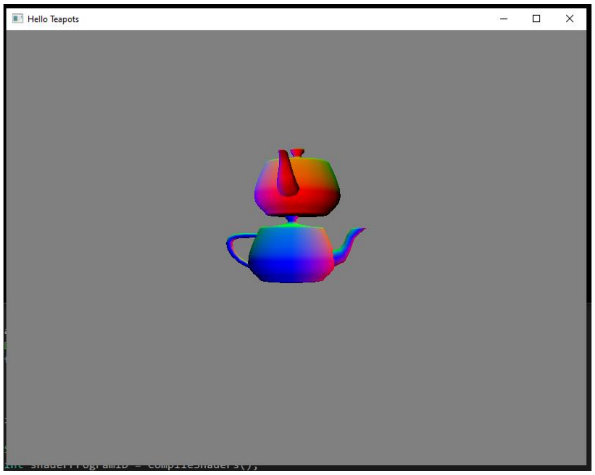
In main function, *glutKeyboardFunc(keypress)* has been written to link the keyboard control function. Then, inside keypress function, I have used if and else if to implement keypress:

Next, the rest of the outputs are as follow:

**5 teapots hierarchy** and **one to many relation:**

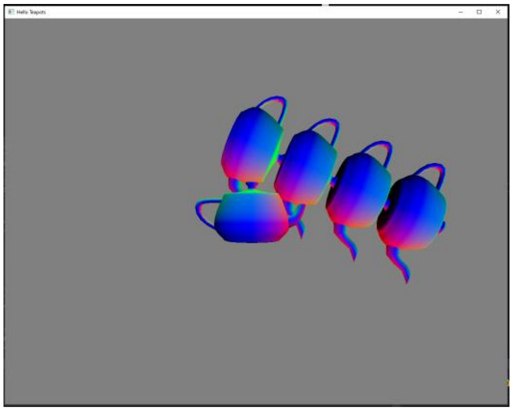


**One to one relation:**

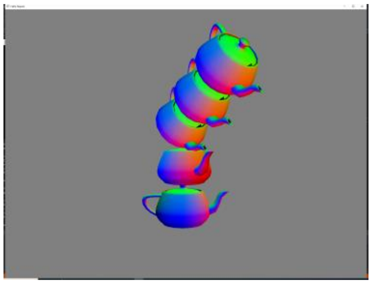


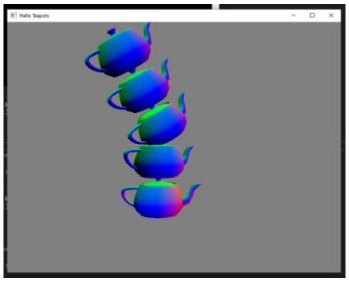
**Interesting/Inventive/Unusual Structure:**

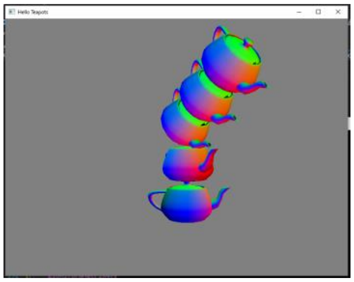
I set changed rotate axis of the second teapot from bottom from  y axis to z axis, then the entire structure do an anti-clock rotation at the center of the root teapot.



In another structure, in `child of hierarchy2, I set the rotation as local2 = rotate_y_deg(local2, rotatez), to roatae in y axis, and in child of hierarchy3, I set the roation as local3 = rotate_z_deg(local3, rotatez) to rotate in z axis. The structure become a bit complex as below:`


Hello Teapots


Hello Teapots


Hello Teapots