COMPUTER GRAPHICS

LAB#5

RAMY AHMED EL SAYED

19015649

PROBLEM STATEMENT

You are required to implement an application that draws a 3-legged stool as shown in figure 1. For the legs, first create one in a display list and then draw it three times rotated appropriately using appropriate transformation.

Your application should handle user input at runtime as follows:

- When user presses ‘x’ / ‘X’. the 3-legged stool should rotate around x-axis in CW/CCW manner respectively.
- When user presses ‘y’ / ‘Y’. the 3-legged stool should rotate around y-axis in CW/CCW manner respectively.
- When user presses ‘z’ / ‘Z’. the 3-legged stool should rotate around z-axis in CW/CCW manner respectively.
- When user presses ‘space’. Toggle drawing mode of the 3-legged between drawing a wireframe object or a solid one.

You can use code in display list and transformation as reference.

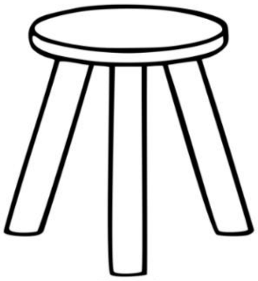


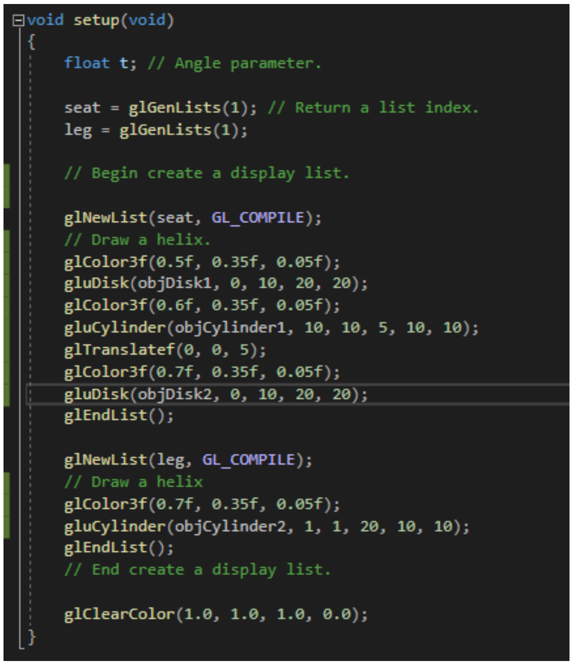
Figure 1

CODE DESCRIPTION

The code is divided into two main phase:

- Setup Phase
- Loop Phase

Setup Phase

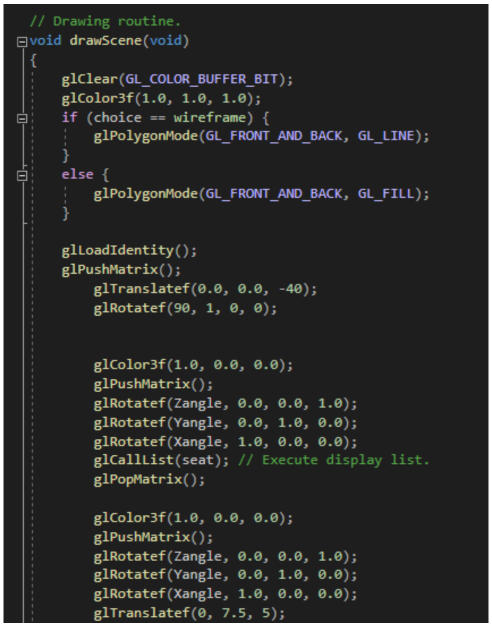


```
void setup(void)
{
    float t; // Angle parameter.

    seat = glGenLists(1); // Return a list index.
    leg = glGenLists(1);

    // Begin create a display list.

    glNewList(seat, GL_COMPILE);
    // Draw a helix.
    glColor3f(0.5f, 0.35f, 0.05f);
    gluDisk(objDisk1, 0, 10, 20, 20);
    glColor3f(0.6f, 0.35f, 0.05f);
    gluCylinder(objCylinder1, 10, 10, 5, 10, 10);
    glTranslatef(0, 0, 5);
    glColor3f(0.7f, 0.35f, 0.05f);
    gluDisk(objDisk2, 0, 10, 20, 20);
    glEndList();

    glNewList(leg, GL_COMPILE);
    // Draw a helix
    glColor3f(0.7f, 0.35f, 0.05f);
    gluCylinder(objCylinder2, 1, 1, 20, 10, 10);
    glEndList();
    // End create a display list.

    glClearColor(1.0, 1.0, 1.0, 0.0);
}
```

The code sequence proceeds as follow:

- Initialize two display lists: the seat and the legs.
- In the first display list, a seat is created using two disks and a cylinder.
- In the second display list, a leg is created using cylinders.

Loop Phase



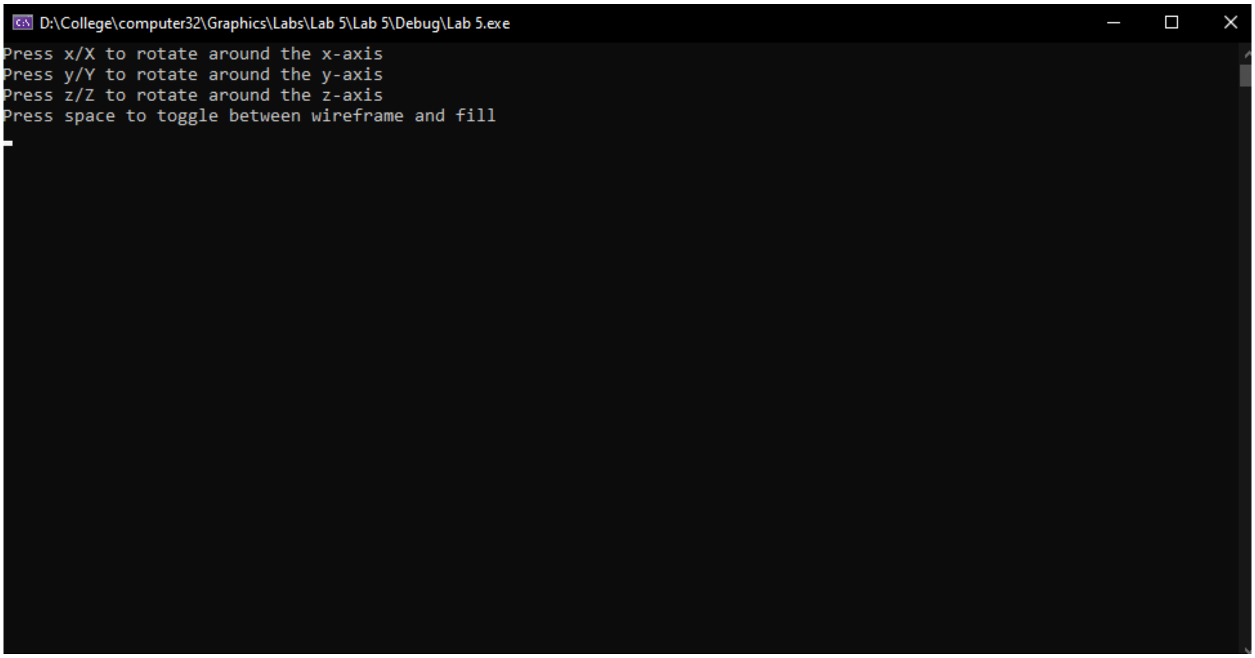
```
// Drawing routine.
void drawScene(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
    glColor3f(1.0, 1.0, 1.0);
    if (choice == wireframe) {
        glPolygonMode(GL_FRONT_AND_BACK, GL_LINE);
    }
    else {
        glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
    }

    glLoadIdentity();
    glPushMatrix();
        glTranslatef(0.0, 0.0, -40);
        glRotatef(90, 1, 0, 0);


        glColor3f(1.0, 0.0, 0.0);
        glPushMatrix();
        glRotatef(Zangle, 0.0, 0.0, 1.0);
        glRotatef(Yangle, 0.0, 1.0, 0.0);
        glRotatef(Xangle, 1.0, 0.0, 0.0);
        glCallList(seat); // Execute display list.
        glPopMatrix();

        glColor3f(1.0, 0.0, 0.0);
        glPushMatrix();
        glRotatef(Zangle, 0.0, 0.0, 1.0);
        glRotatef(Yangle, 0.0, 1.0, 0.0);
        glRotatef(Xangle, 1.0, 0.0, 0.0);
        glTranslatef(0, 7.5, 5);
```

The loop starts by specifying the drawing type depending on user input.

Then we build our chair using a hierarchical by using the MODEL_VIEW_MATRIX.

Example Of Running Code

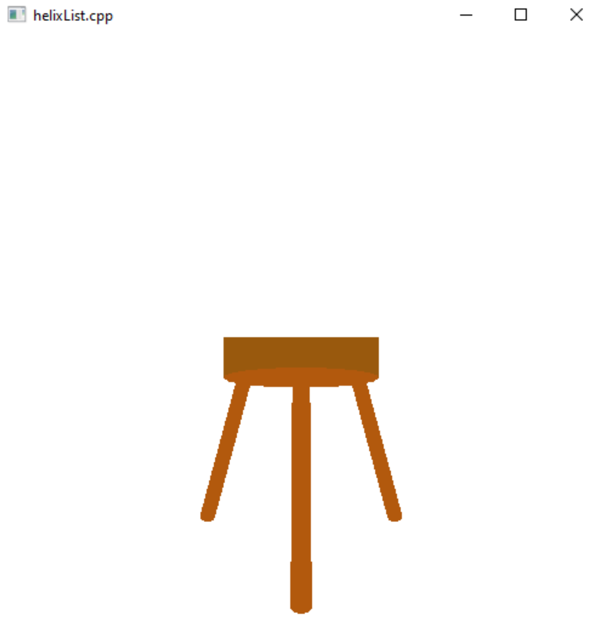
Main Screen



D:\College\computer32\Graphics\Labs\Lab 5\Lab 5\Debug\Lab 5.exe

```
Press x/X to rotate around the x-axis
Press y/Y to rotate around the y-axis
Press z/Z to rotate around the z-axis
Press space to toggle between wireframe and fill
```

CHAIR



CHALLENGES

The main challenge was to build a hierarchical model out of the chair .