Graphics

Lab 3

---

Name / Ahmed Abdallah Abo-Eleid

ID / 19015274

---

## Problem statement:

- You are required to implement an application that draws a 3-legged stool
- 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.

## Code:

### 1) Global variables

```
// Globals.
static unsigned int aClyinderLeg;
static unsigned int aCircle;
static bool solidMode = false;
static float rotationAngleX = 0.0f;
static float rotationAngleY = 0.0f;
static float rotationAngleZ = 0.0f;
```

2) Setup function : which I used to create list responsible for drawing cylinder which I will use to draw legs when I call this list , and I do also for seat of chair

```
// Initialization routine.
void setup(void)
{
    aClyinderLeg = glGenLists(1); // Return a list index.
    // Begin create a display list.
    glNewList(aClyinderLeg, GL_COMPILE);
    glutSolidCylinder(3.0, 30.0, 10, 10);
    glEndList();
    // End create a display list.

    aCircle = glGenLists(1);
    glNewList(aCircle, GL_COMPILE);
    glutSolidCylinder(30.0, 2.0, 10, 10);
    glEndList();

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

3) DrawScene function: I call list for drawing 3 legs and call another list to draw seat, before calling list I do required translation & rotation for this object I will draw.in first of this function I translate all scene to center and rotate it then I return it to same place again.

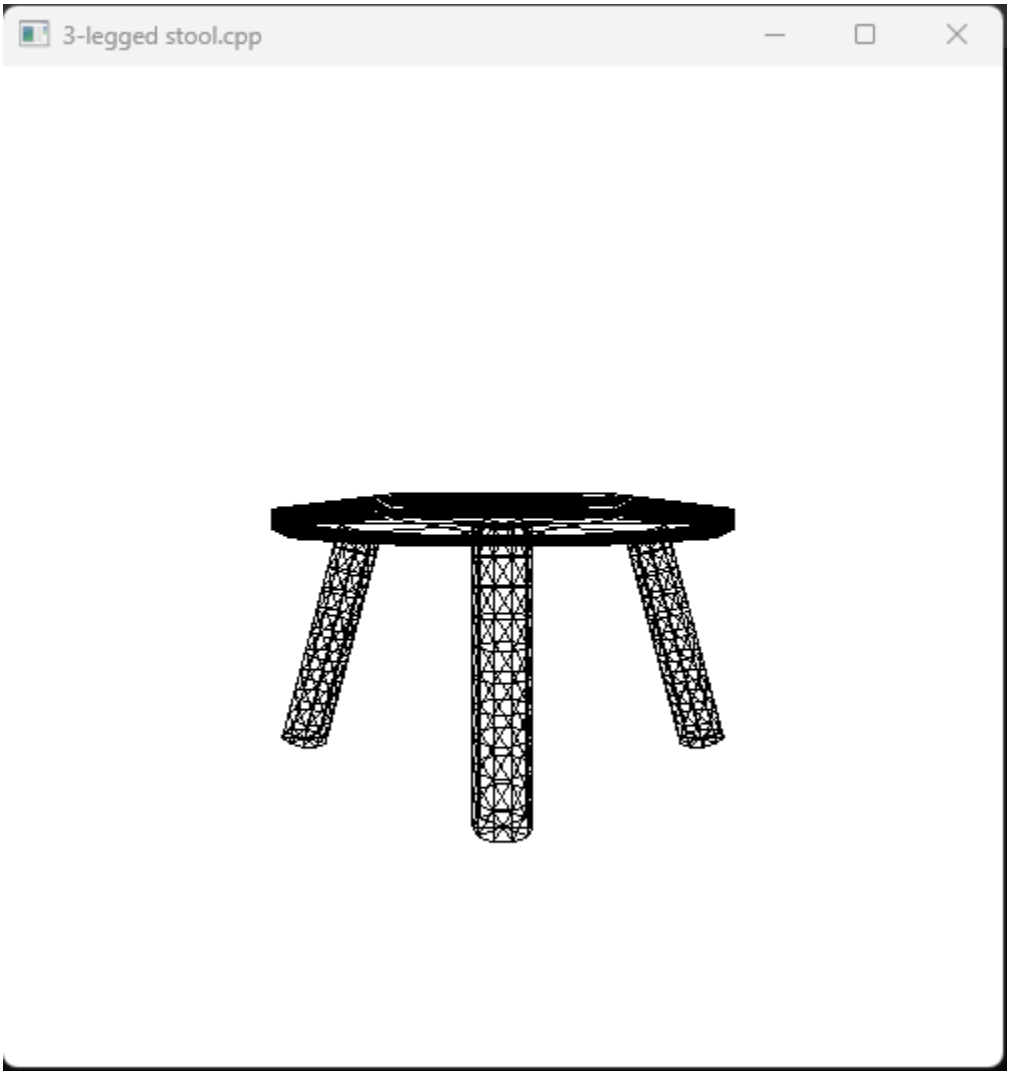
```
// Drawing routine.
void drawScene(void)
{
    glClear(GL_COLOR_BUFFER_BIT);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();

    glTranslatef(0.0, 5.0, -70.0);
    glRotatef(rotationAngleX, 1.0f, 0.0f, 0.0f);
    glRotatef(rotationAngleY, 0.0f, 1.0f, 0.0f);
    glRotatef(rotationAngleZ, 0.0f, 0.0f, 1.0f);
    glTranslatef(0.0, -5.0, 70.0);
    if (solidMode) {
        glPolygonMode(GL_FRONT_AND_BACK, GL_FILL);
    }
    else {
        glPolygonMode(GL_FRONT_AND_BACK, GL_LINE);
    }

    // Draw seat
    glColor3f(0.0, 0.0, 0.0);
    glPushMatrix();
    glTranslatef(0.0, 6.0, -70.0);
    glRotatef(90.0, 1, 0, 0);
    glCallList(aCircle); // Execute display list.
    glPopMatrix();

    // Draw first leg
    glPushMatrix();
    glTranslatef(-20.0, 5.0, -70.0);
    glRotatef(90.0,1,0,0);
    glRotatef(-15.0,0,1,0);
    glCallList(aClyinderLeg); // Execute display list.
    glPopMatrix();

    // Draw second leg
    glColor3f(0.0, 0.0, 0.0);
    glPushMatrix();
    glTranslatef(0.0, 4.0, -50.0);
    glRotatef(90.0,1,0,0);
    glCallList(aClyinderLeg); // Execute display list.
    glPopMatrix();

    // Draw third leg
    glColor3f(0.0, 0.0, 0.0);
    glPushMatrix();
    glTranslatef(20.0, 5.0, -70.0);
    glRotatef(90.0,1,0,0);
    glRotatef(15.0,0,1,0);
    glCallList(aClyinderLeg); // Execute display list.
    glPopMatrix();

    glFlush();
}
```

4) KeyInput function: I handle user input to rotate object around x or y or z , and handle user input to toggle between wireframe and solid one.

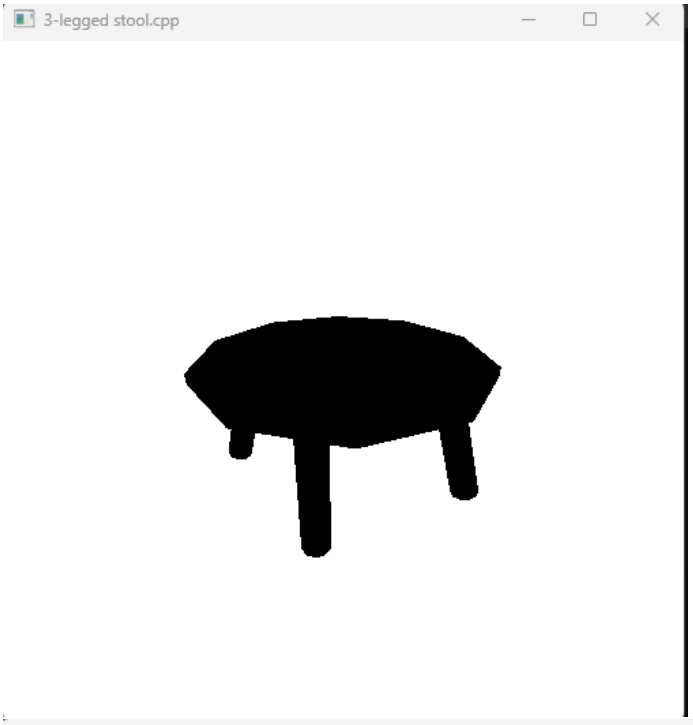
```
void keyInput(unsigned char key, int x, int y)
{
    switch (key)
    {
        case 'x':
            rotationAngleX -= 5.0f;
            break;
        case 'X':
            rotationAngleX += 5.0f;
            break;
        case 'y':
            rotationAngleY -= 5.0f;
            break;
        case 'Y':
            rotationAngleY += 5.0f;
            break;
        case 'z':
            rotationAngleZ -= 5.0f;
            break;
        case 'Z':
            rotationAngleZ += 5.0f;
            break;
        case ' ':
            // Toggle between solid and wireframe mode.
            solidMode = !solidMode;
            break;
        case 27: // ESC key
            exit(0);
            break;
        default:
            break;
    }
    glutPostRedisplay();
}
```

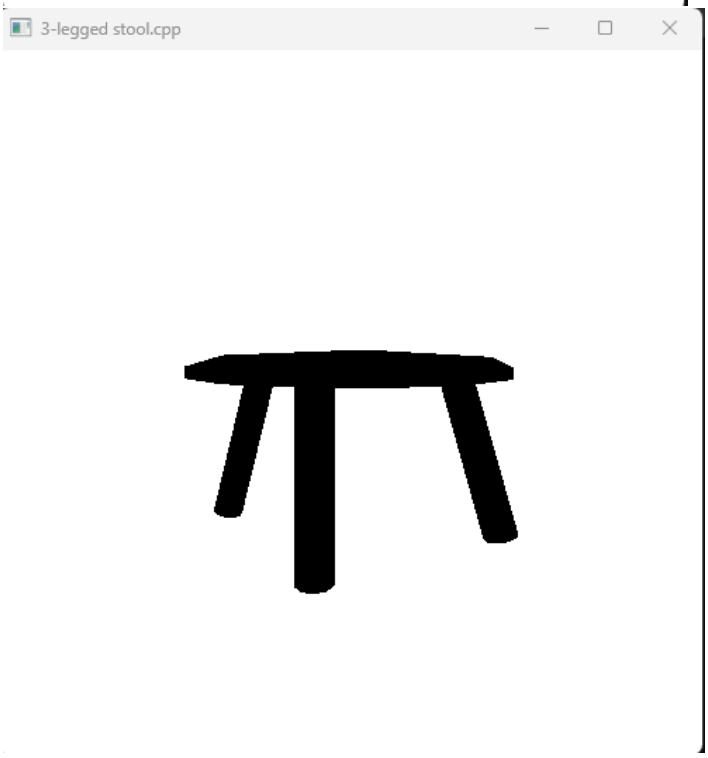
## **Sample runs**:

- Wireframe

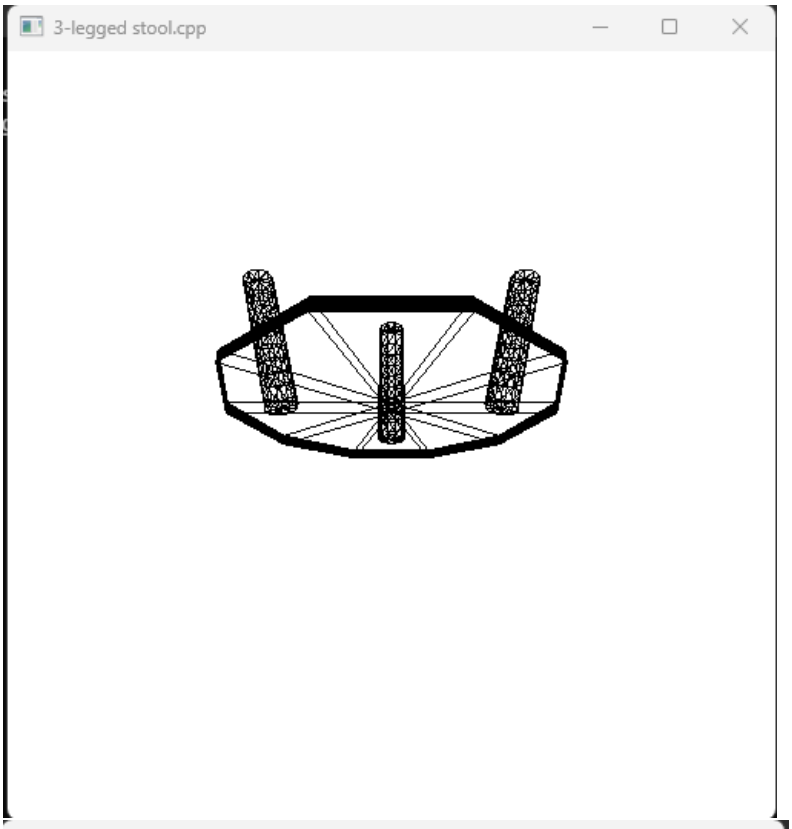


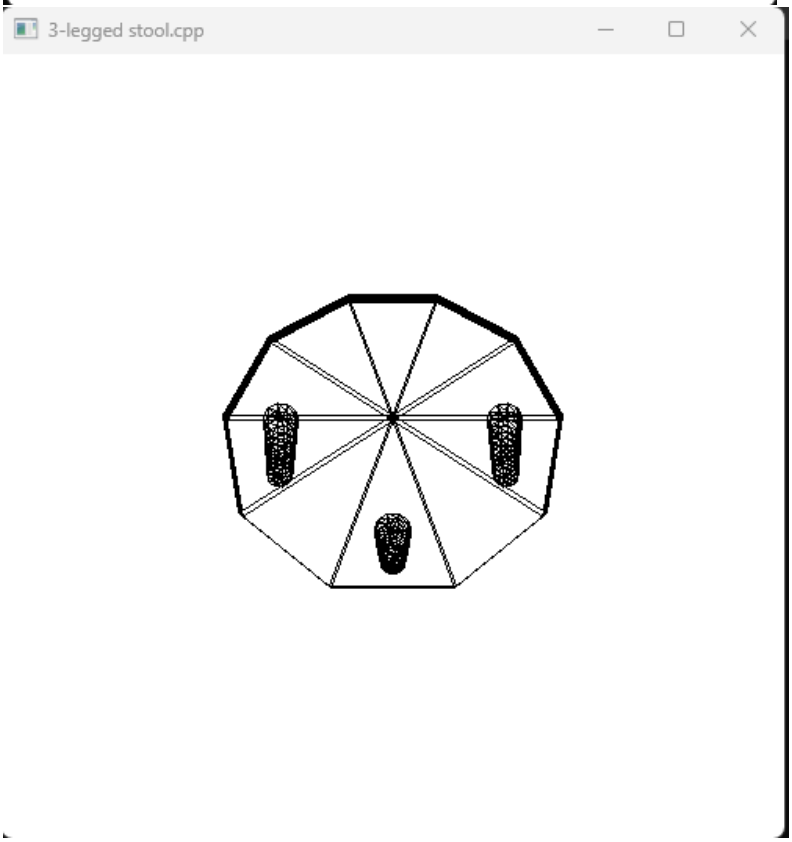
- Solid



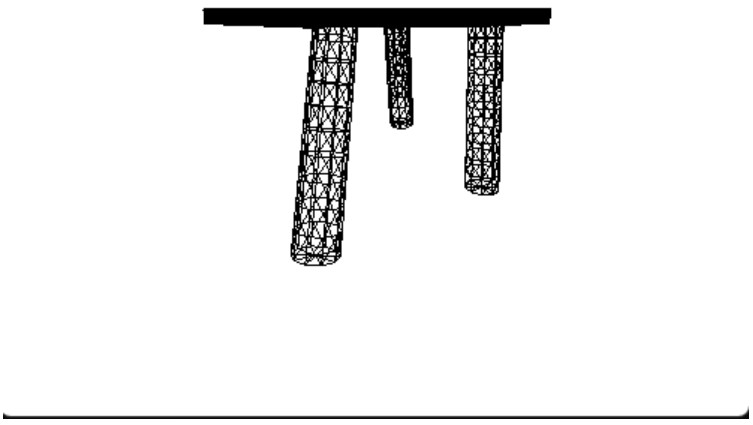



- Rotate around x



- Rotate around y

- Rotate around z