CS460

Homework 3 Report

Sinem Ozden

1. Problem Statement:

Part A: 3D viewing with an active virtual camera. The camera's position, orientation and field of view should be adjustable. Roll, Pitch, Yaw and Zoom will be implemented.

Part B: You are required to implement a series of geometric transformation to construct a "lever" object using the primitive models "cylinder" and "sphere", and perform the rotation of the "lever" in a specified orientation.

2. Algorithm Design:

2.1. I created a menu for camera functions it's a separate .cpp file. In menu function. I get the x,y coordinates of the mouse and create the menu accordingly. If it's in viewport 1 I create menu for roll+, roll-, pitch+, pitch-, yaw+, yaw-, slide+, slide-. If it's in other viewports I created a menu for levers rotation.

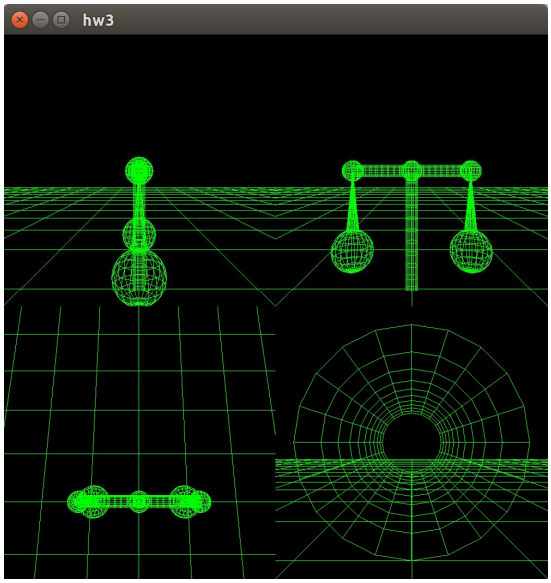
2.2. I created viewport at the different locations of the window. I rotate the camera with glRotatef. I also set the camera location with gluLookAt. Projection sets the perspective. The RenderGLScene creates the object and the background. In RenderGLScene2 I draw the lever object.

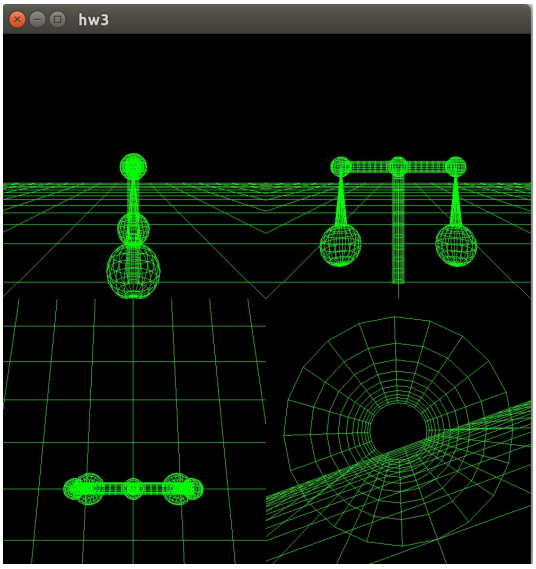
```
glViewport(width/2.0, 0, width/2.0, height/2.0); //V1
{

    projection(width/2.0, height/2.0, 1);
    gluLookAt(0, 0, m_slide, 0.0, 0.0, 0.0, 0.0, 1.0, 0.0);
   // glClear(GL_DEPTH_BUFFER_BIT|GL_COLOR_BUFFER_BIT );
    glPushMatrix();
    glRotatef(m_xtheta, 1,0,0);
    glRotatef(m_ytheta, 0,1,0);
    glRotatef(m_ztheta, 0,0,1);
    RenderGLScene(5, 5, 30);
    glPopMatrix();
glViewport(0, 0, width/2.0, height/2.0); //V2
{

   projection(width/2.0, height/2.0, 1);
   gluLookAt(0.0, 40.0, 0.0, 0.0, 0.0, 10.0, 0.0, 1.0, 0.0);
  // glClear(GL_DEPTH_BUFFER_BIT|GL_COLOR_BUFFER_BIT );
   glPushMatrix();
   RenderGLScene2(3, 3, 30);
   glPopMatrix();
}
```

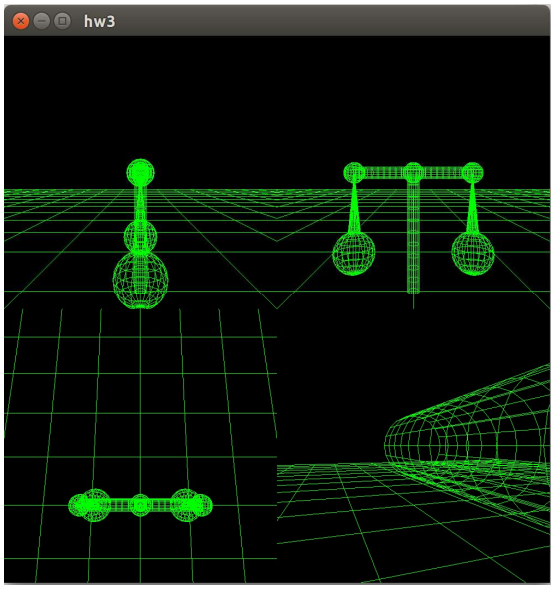
2.3. To do the rotations I use the functions to change the roll, pitch, yaw and slide. I call these functions when they clicked at menu. They change the global values according the angle and when the viewport redisplayed it uses the updated angles.

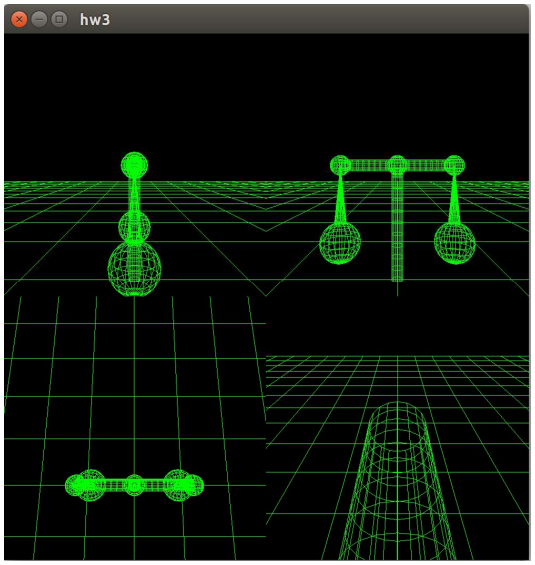
```
void RollUp(float n){
  m_ztheta+=n;
  if(m_ztheta > 360) m_ztheta=-360;
}
void RollDown(float n){
  m_ztheta-=n;
  if(m_ztheta < -360) m_ztheta=360;
}
void PitchUp(float n){
  m_ytheta+=n;
  if(m_ytheta > 360) m_ytheta=-360;
}
void PitchDown(float n){
  m_ytheta-=n;
  if(m_ytheta < -360) m_ytheta=360;
}
void YawUp(float n){
  m_xtheta+=n;
  if(m_xtheta > 360) m_xtheta=-360;
}
void YawDown(float n){
  m_xtheta-=n;
  if(m_xtheta < -360) m_xtheta=360;
}
void SlideUp(float n){
  m_slide+=n;
}
void SlideDown(float n ){
  m_slide-=n;
}
```

to



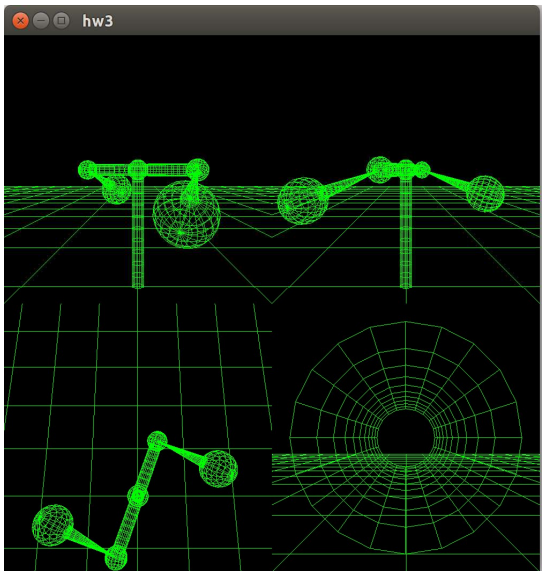


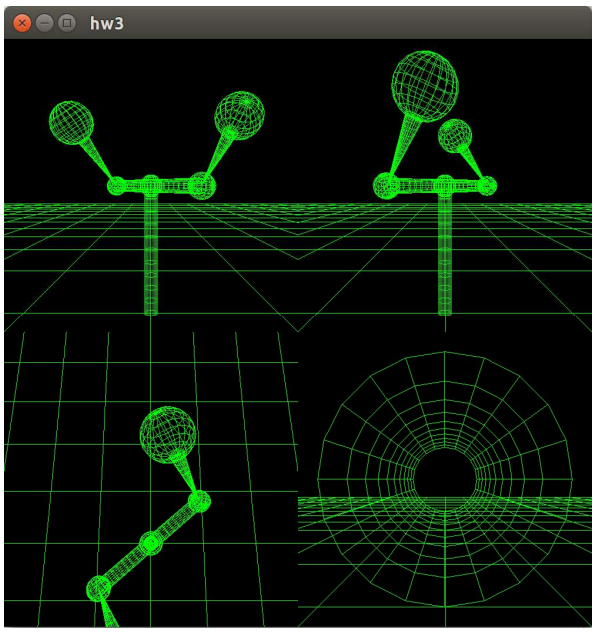


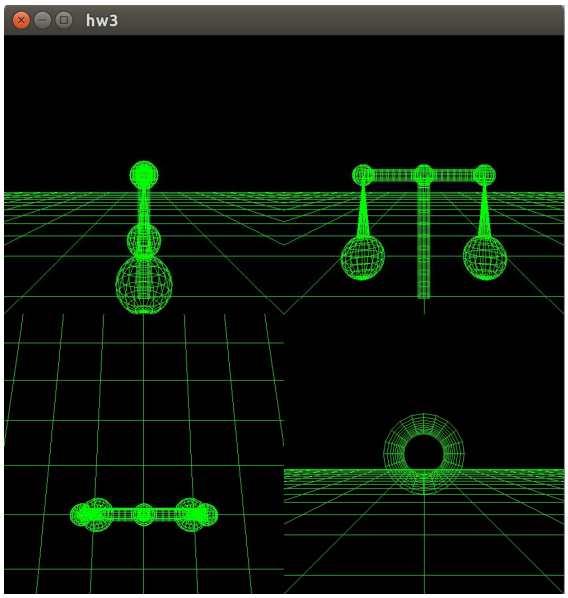


2.4. To create the lever object, I rotate and translated the sphere and cylinder object. To rotate the lever object, I rotated the s2 first this rotated the whole object. Then I rotated the b1 and then b3 in negative direction. I get the angle from the menu and increate the angle as the clicks increase.

```
//s1
glRotatef(90.0f, 1.0f, 0.0f, 0.0f);
gluCylinder(quadObj, base_radius/2, top_radius/2, 20, 20, 10);

//b2
gluSphere(quadObj, (base_radius + 0.5)/2 , 20, 10);

//s2
glRotatef(-90.0f, 1.0f, 0.0f, 0.0f);
glRotatef(90.0f, 0.0f, 1.0f, 0.0f);
glRotatef(m_lever, 0.0f, 1.0f, 0.0f);
gluCylinder(quadObj, base_radius/2, top_radius/2, 10, 20, 10);

//s3
glRotatef(180.0f, 1.0f, 0.0f, 0.0f);
gluCylinder(quadObj, base_radius/2, top_radius/2, 10, 20, 10);

//b1
glTranslated(0.0, 0.0 , 10.0);
gluSphere(quadObj, (base_radius + 0.5)/2, 20, 10);

//s4
glRotatef(-90.0f, 1.0f, 0.0f, 0.0f);
glRotatef(m_lever, 0.0f, 1.0f, 0.0f);
gluCylinder(quadObj, 0, top_radius/2, 10, 20, 10);

//b4
glTranslated(0.0, 0.0 , 10.0 + base_radius + 0.5);
gluSphere(quadObj, base_radius + 0.5, 20, 10);

//b3
glTranslated(0.0, 20.0 , -(10.0 + base_radius + 0.5));
gluSphere(quadObj, (base_radius + 0.5)/2, 20, 10);

//s5
glRotatef(-2*m_lever, 0.0f, 1.0f, 0.0f);
gluCylinder(quadObj, 0, top_radius/2, 10, 20, 10);

//b5
glTranslated(0.0, 0.0 , 10.0 + base_radius + 0.5);
gluSphere(quadObj, base_radius + 0.5, 20, 10);
```







3. To compile the first 3 parts of the assignment run these lines:
   make
   ./project