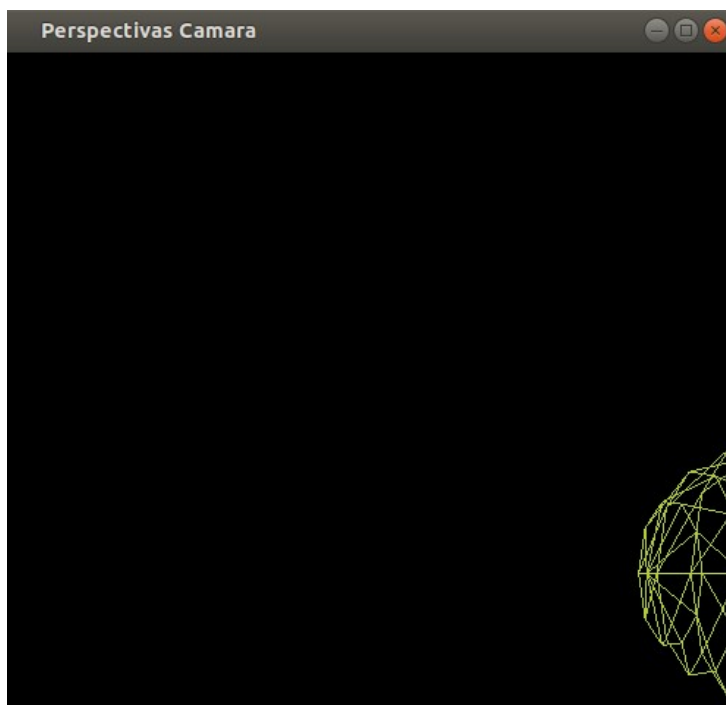
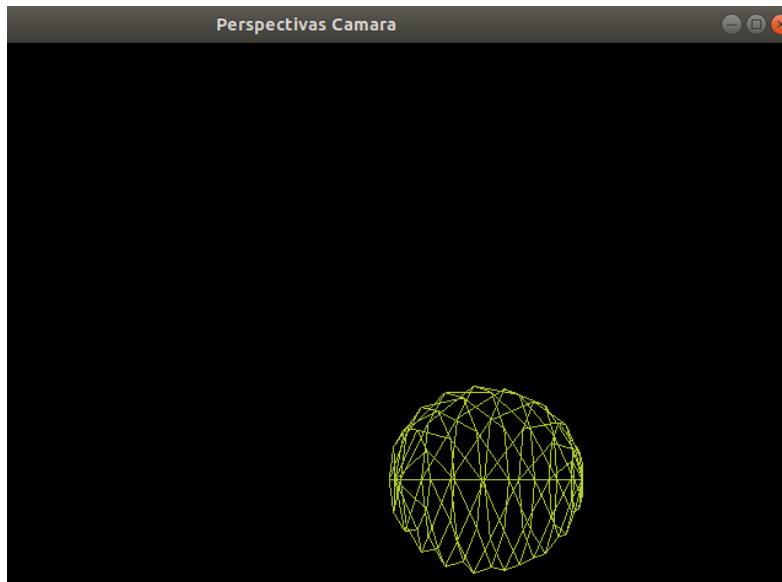


Movimiento en 3D

1.-Mover la cámara con el mouse:

Se apreta la tecla F3:

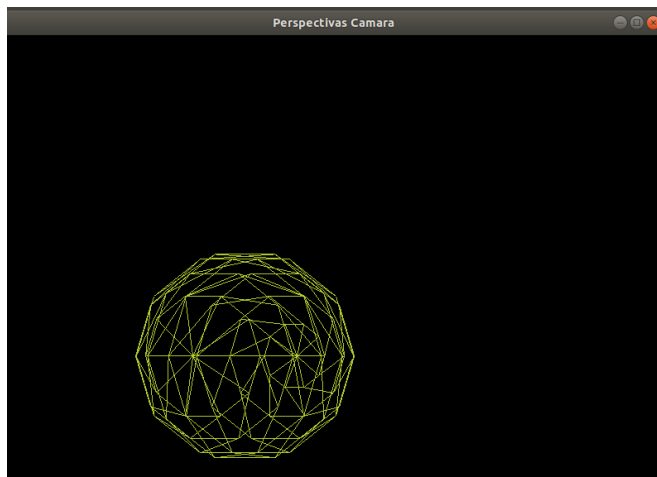
```
85     if(key==GLUT_KEY_F3){
86         cout<<"Mover camara con mouse"<<endl;
87         glutPassiveMotionFunc(MouseMotion);
88         LOCAL_MyCamera->camMovimiento = CAM_PASEAR;
89         LOCAL_MyCamera->camAtY = 0;
90         LOCAL_MyCamera->camViewY = 0;
91         SetDependentParametersCamera( LOCAL_MyCamera );
92     }
```



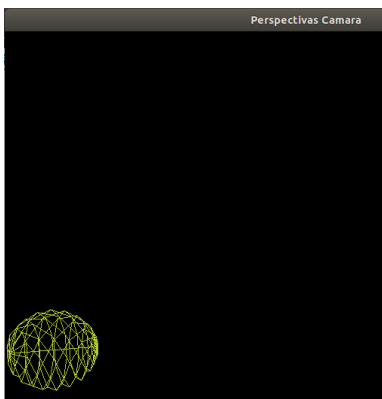
2.- Cambiar la perspectiva solo se apretá las siguientes teclas :

```
if(key==GLUT_KEY_F4){  
    if(LOCAL_MyCamera->camProjection == CAM_CONIC){  
        LOCAL_MyCamera->x1=-5;//  
        LOCAL_MyCamera->x2=7;////aumenta enflaquece  
        LOCAL_MyCamera->y1=-5;//  
        LOCAL_MyCamera->y2=6;//  
        LOCAL_MyCamera->z1=-20;  
        LOCAL_MyCamera->z2=500;  
        LOCAL_MyCamera->camProjection = CAM_PARALLEL;  
        cout<<"Modo de Proyección Ortogonal"<<endl;  
    }else{  
        cout<<"Modo de Proyección Perspectiva"<<endl;  
        LOCAL_MyCamera->camProjection = CAM_CONIC;  
    }  
}
```

- F2 para cambiar a ortogonal



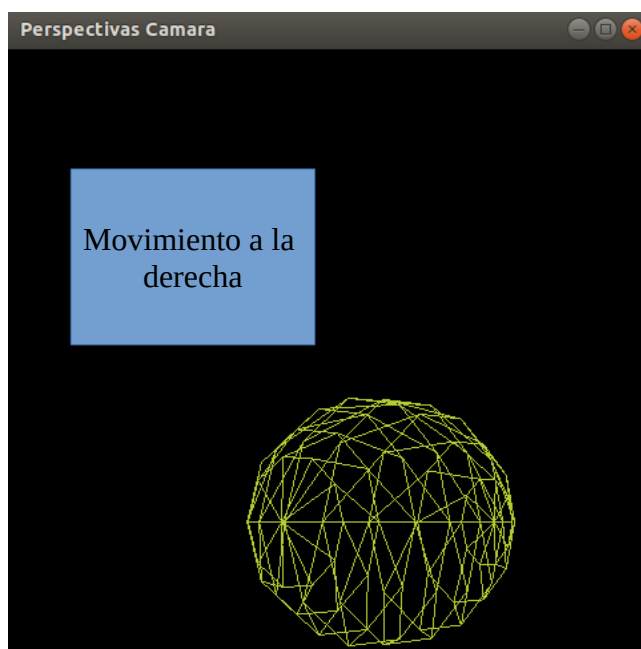
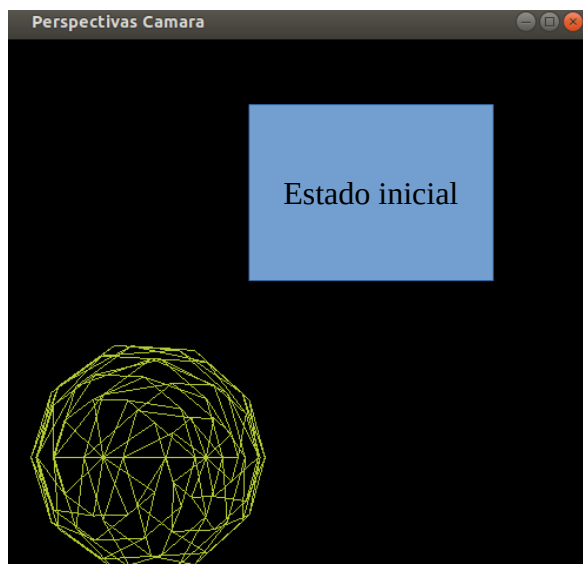
- F7 para cambiar a perspectiva(se observa más pequeño)

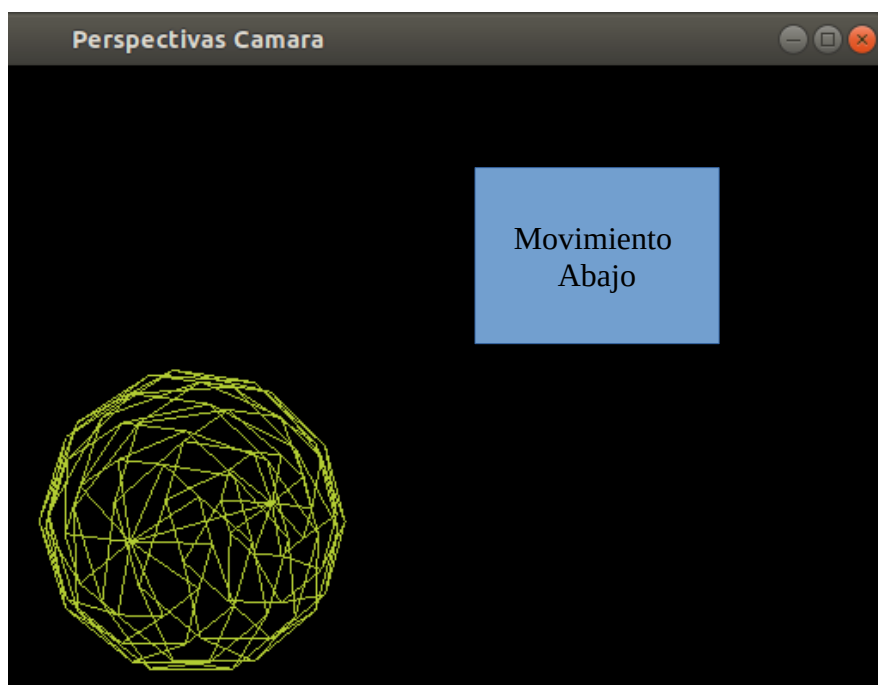
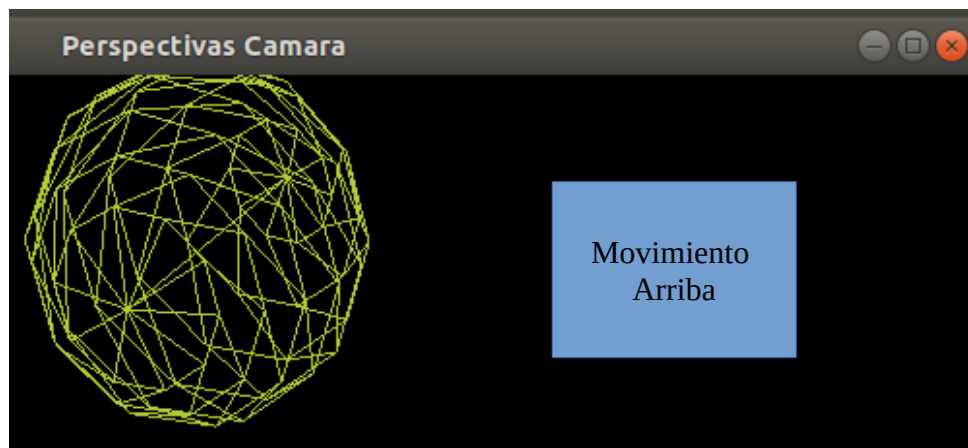
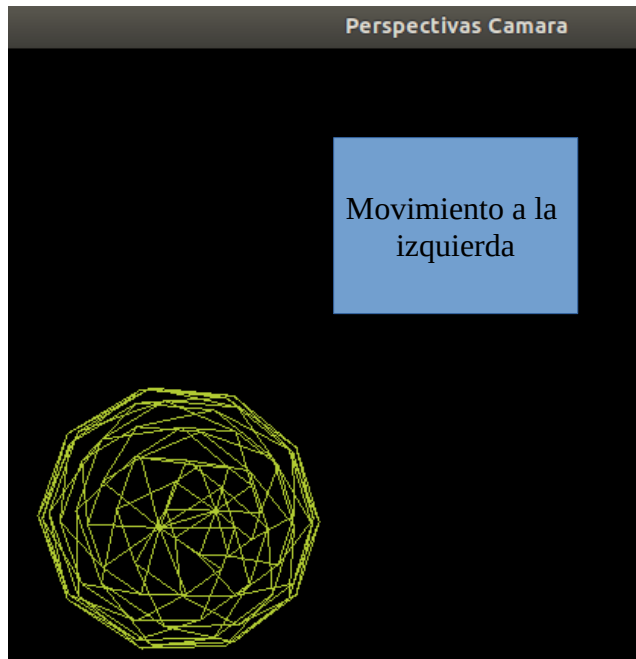


1.-Movimiento del teclado

```
119  if (key==GLUT_KEY_UP){
120      cout<<"Movimiento Arriba"<<endl;
121      PanCamera(LOCAL_MyCamera,0,-1);
122  }
123  if (key==GLUT_KEY_DOWN){
124      cout<<"Movimiento Abajo"<<endl;
125      PanCamera(LOCAL_MyCamera,0,1);
126  }
127  if (key==GLUT_KEY_LEFT){
128      cout<<"Movimiento Izquierda"<<endl;
129      PanCamera(LOCAL_MyCamera,1,0);
130  }
131  if (key==GLUT_KEY_RIGHT){
132      cout<<"Movimiento Derecha"<<endl;
133      PanCamera(LOCAL_MyCamera,-1,0);
134  }
```

a)Moviendo con direccionales





b) Con las funciones especiales

Con la tecla F2: Se mueve el objeto en proyección

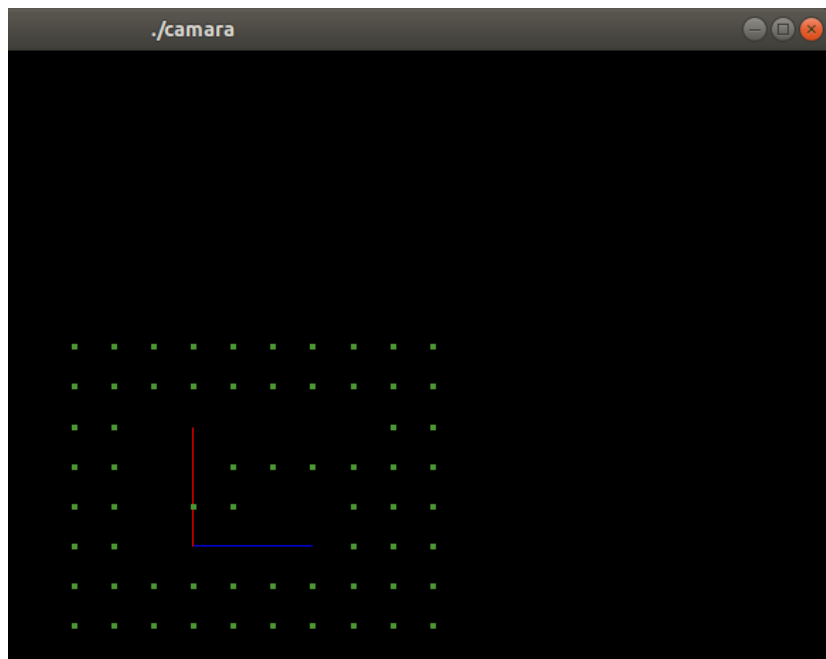
```
if(key==GLUT_KEY_F3){
    cout<<"Mover camara con mouse"<<endl;
    glutPassiveMotionFunc(MouseMotion);
    LOCAL_MyCamera->camMovimiento = CAM_PASEAR;
    LOCAL_MyCamera->camAtY = 0;
    LOCAL_MyCamera->camViewY = 0;
    SetDependentParametersCamera( LOCAL_MyCamera );
}
```

```
if(key==GLUT_KEY_F1){
    cout<<"Camara parada"<<endl;
    glutPassiveMotionFunc(MouseMotion);
    LOCAL_MyCamera->camMovimiento = CAM_STOP;
}
if(key==GLUT_KEY_F2){
    cout<<"Mover objeto en proyeccion"<<endl;
    glutPassiveMotionFunc(Examinar);
    LOCAL_MyCamera->camMovimiento = CAM_EXAMINAR;
}
```

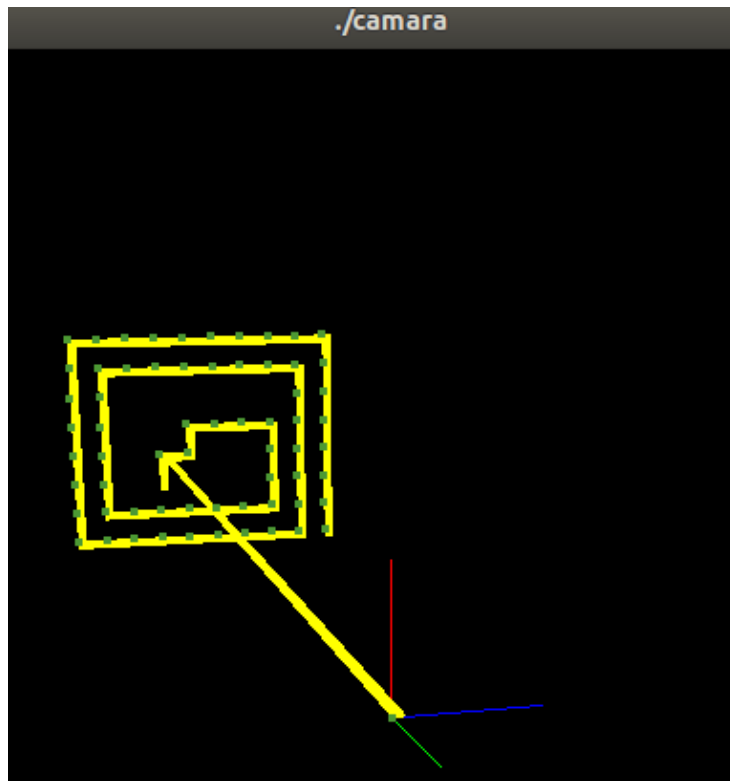
- Con F3 y se activa con click derecho, se puede observar que se mueve la cámara en la dirección del puntero.
- Con F1 se para la cámara.
- Con F2, se puede hacer girar la esfera.

5.- Avance de la tarea 3 Rastreo

Primer paso teclear de arriba a abajo



Se traza su rastro.



En la practica 3 especifica la 5.4 hacer un helicoidal.

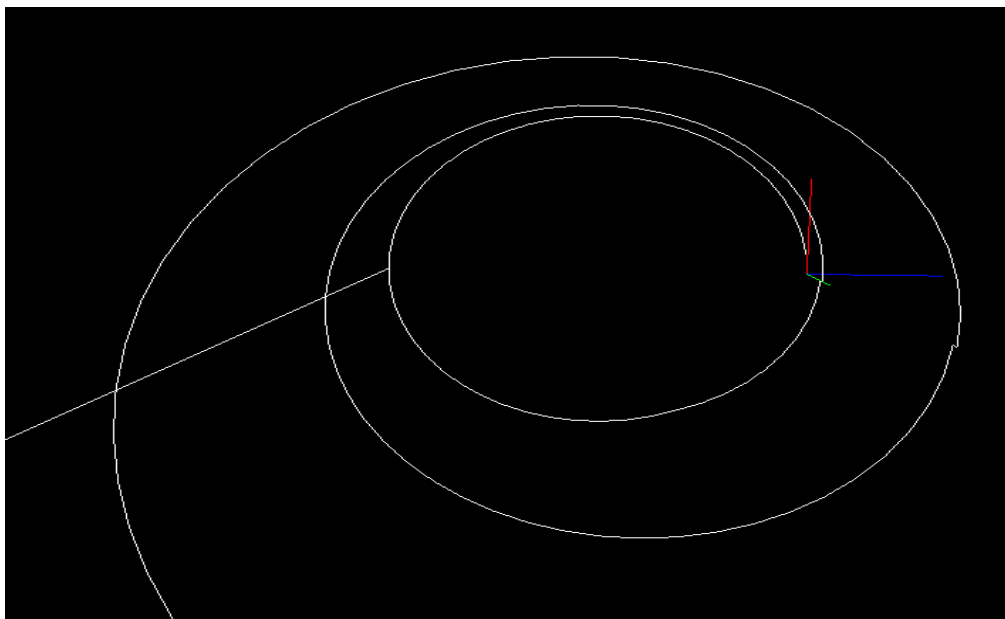


Ilustración 1: Donde la recta roja representa el eje y la recta de color azul representa x y por ultimo la recta verde representa la z.

