

Código 13: Luz Puntual

```
#include <windows.h>
#include <GL/glut.h>

float light_ambient [] = {0.5,0.7,0,0};
float light_diffuse [] = {1,1,1,0};
float light_pos [] = {0.0,2.5,0.0,1.0};

float mat_ambient [] = {0,.0,.0,0};
float mat_diffuse [] = {.5,.0,.0,0};
float mat_specular [] = {.5,.6,.1,0};
float mat_emission [] = {0,.0,.0,0};
float mat_shininess = 64;

float focus_ambient_diffuse [] = {1,1,.1,0};
float focus_emission [] = {0.5,0.7,0.1,0};

void initgl()
{
    glClearColor(0.7,0.7,0.7,0.0);

    glLightfv(GL_LIGHT0,GL_AMBIENT,light_ambient);
    glLightfv(GL_LIGHT0,GL_DIFFUSE,light_diffuse);
    glLightfv(GL_LIGHT0,GL_POSITION,light_pos);

    glEnable(GL_LIGHTING);
    glEnable(GL_LIGHT0);
    glEnable(GL_NORMALIZE);
    glEnable(GL_DEPTH_TEST);

    glMatrixMode(GL_PROJECTION);
    glLoadIdentity();
    glOrtho(-3,3,-2,4,-3,3);
    glMatrixMode(GL_MODELVIEW);
    glLoadIdentity();
}

void display(void)
{
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    glPushMatrix(); // fuente luz
    glMaterialfv(GL_FRONT,GL_AMBIENT_AND_DIFFUSE,focus_ambient_diffuse);
    glMaterialfv(GL_FRONT,GL_EMISSION,focus_emission);
    glTranslatef(light_pos[0],light_pos[1],light_pos[2]);
    glutSolidSphere(.35,15,15);
    glPopMatrix();

    glPushMatrix(); //esfera
    glMaterialfv(GL_FRONT,GL_AMBIENT,mat_ambient);
    glMaterialfv(GL_FRONT,GL_DIFFUSE,mat_diffuse);
    glMaterialfv(GL_FRONT,GL_SPECULAR,mat_specular);
    glMaterialfv(GL_FRONT,GL_EMISSION,mat_emission);
    glMaterialf(GL_FRONT,GL_SHININESS,mat_shininess);
    glTranslatef(2.4,2,0);
    glutSolidSphere(.4,20,20);
    glPopMatrix();

    glutSolidTeapot(1.3);
    glFlush();
}

int main(int argc, char ** argv)
{
    glutInitDisplayMode(GLUT_SINGLE|GLUT_RGBA|GLUT_DEPTH);
    glutInitWindowPosition(20,20);
    glutInitWindowSize(350,350);
    glutCreateWindow("Luz Puntual");
    initgl();
    glutDisplayFunc(display);
    glutMainLoop();
    return 0;
}
```

