***PRACTICA # 6***

#include <Gl/glut.h>

GLfloat anguloCubox = 0.0f;

GLfloat anguloCuboy = 0.0f;

GLfloat anguloEsfera = 0.0f;

GLint ancho=400;

GLint alto=400;

int hazperspec=0;

void reshape(int width, int height)

{

glViewport(0, 0, width, height);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

if(hazperspec)

gluPerspective(60.0f, (GLfloat)width/(GLfloat)height, 1.0f, 20.0f);

else

glOrtho(-4,4,-4,4,1,10);

glMatrixMode(GL\_MODELVIEW);

ancho=width;

alto=height;

}

void dibujoCubo(void)

{

glColor3f(1.0f, 0.0f,0.0f);

glBegin(GL\_QUADS); // CARA FRONTAL

glVertex3f(-1.0f,-1.0f,1.0f);

glVertex3f(1.0f,-1.0f,1.0f);

glVertex3f(1.0f,1.0f,1.0f);

glVertex3f(-1.0f,1.0f,1.0f);

glEnd();

glColor3f(0.0f, 1.0f,0.0f);

glBegin(GL\_QUADS); // CARA TRASERA

glVertex3f(1.0f,-1.0f,-1.0f);

glVertex3f(-1.0f,-1.0f,-1.0f);

glVertex3f(-1.0f,1.0f,-1.0f);

glVertex3f(1.0f,1.0f,-1.0f);

glEnd();

glColor3f(0.0f, 0.0f,1.0f);

glBegin(GL\_QUADS); // CARA LATERAL IZDA

glVertex3f(-1.0f,-1.0f,-1.0f);

glVertex3f(-1.0f,-1.0f,1.0f);

glVertex3f(-1.0f,1.0f,1.0f);

glVertex3f(-1.0f,1.0f,-1.0f);

glEnd();

glColor3f(1.0f, 1.0f,0.0f);

glBegin(GL\_QUADS); // CARA LATERAL DCHA

glVertex3f(1.0f,-1.0f,1.0f);

glVertex3f(1.0f,-1.0f,-1.0f);

glVertex3f(1.0f,1.0f,-1.0f);

glVertex3f(1.0f,1.0f,1.0f);

glEnd();

glColor3f(1.0f, 1.0f,1.0f);

glBegin(GL\_QUADS); // CARA ARRIBA

glVertex3f(-1.0f,1.0f,1.0f);

glVertex3f(1.0f,1.0f,1.0f);

glVertex3f(1.0f,1.0f,-1.0f);

glVertex3f(-1.0f,1.0f,-1.0f);

glEnd();

glColor3f(1.0f, 0.0f,1.0f);

glBegin(GL\_QUADS); // CARA ABAJO

glVertex3f(1.0f,-1.0f,-1.0);

glVertex3f(1.0f,-1.0f,1.0);

glVertex3f(-1.0f,-1.0f,1.0);

glVertex3f(-1.0f,-1.0f,-1.0);

glEnd();

}

void display()

{

glClear(GL\_COLOR\_BUFFER\_BIT | GL\_DEPTH\_BUFFER\_BIT);

glLoadIdentity();

glTranslatef(0.0f, 0.0f, -5.0f);

glRotatef(anguloCubox, 1.0f,0.0f,0.0f);

glRotatef(anguloCuboy, 0.0f,1.0f,0.0f);

dibujoCubo();

glLoadIdentity();

glTranslatef(0.0f, 0.0f, -5.0f);

glRotatef(anguloEsfera, 0.0f,1.0f,0.0f);

glTranslatef(3.0f, 0.0f, 0.0f);

glColor3f(1.0f,0.0f,0.0f);

glutWireSphere(0.5f,8,8);

glFlush();

glutSwapBuffers();

anguloCubox += 0.1f;

anguloCuboy += 0.1f;

anguloEsfera += 0.2f;

}

void Init()

{

glClearColor(0,0,0,0);

glEnable(GL\_DEPTH\_TEST);

ancho = 400;

alto = 400;

}

void idle()

{

display();

}

void keyboard(unsigned char key, int x, int y)

{

switch(key)

{

case 'p':

case 'P':

hazperspec=1;

reshape(ancho,alto);

break;

case 'o':

case 'O':

hazperspec=0;

reshape(ancho,alto);

break;

case 27: // escape

exit(0);

break;

}

}

int main(int argc, char\*\* argv)

{

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_DOUBLE | GLUT\_RGB);

glutInitWindowPosition(100, 100);

glutInitWindowSize(ancho, alto);

glutCreateWindow("PRACTICA 6");

Init();

glutDisplayFunc(display);

glutReshapeFunc(reshape);

glutIdleFunc(idle);

glutKeyboardFunc(keyboard);

glutMainLoop();

return 0;

}