#include<GL/freeglut.h>

#include<GL/gl.h>

#include<math.h>

int framenumber=0;

void drawwindmill(){

glColor3f(1,1,0);

glBegin(GL\_POLYGON);

glVertex2f(-0.5f, 0);

glVertex2f(-0.5f, 3);

glVertex2f(0.5f, 0);

glVertex2f(0.5f, 3);

glEnd();

glTranslated(0,3,0);

glColor3f(1,0,1);

glRotated(framenumber\*(180.0/45), 0, 0, 1);

for(int i=0; i<4; i++){

glRotated(90,0,0,1);

glBegin(GL\_POLYGON);

glVertex2f(0,0);

glVertex2f(4,0.5f);

glVertex2f(4,-0.5f);

glEnd();

}

}

void renderfunction(){

glClearColor(0,0,0,0);

glClear(GL\_COLOR\_BUFFER\_BIT);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

glOrtho(-1,7,-1,8,-1,1);

glMatrixMode(GL\_MODELVIEW);

glPushMatrix();

glTranslated(-0.5, 0.5, 0.0);

glScaled(1,1,1);

drawwindmill();

glPopMatrix();

glPushMatrix();

glTranslated(3.0,2.0,0.0);

glScaled(4,4,4);

drawwindmill();

glPopMatrix();

glutSwapBuffers();

}

void doFrame(int v){ //int v as parameter?why?

framenumber++;

glutPostRedisplay();

glutTimerFunc(10, doFrame,0);

}

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

glutInit(&argc, argv);

glutInitDisplayMode(GLUT\_DOUBLE);

glutInitWindowSize(400, 400);

glutInitWindowPosition(100,100);

glutDisplayFunc(renderfunction);

glutCreateWindow("animation");

glutTimerFunc(200, doFrame, 0);

glutMainLoop();

return 1;

}