Implement animation principles for any object

\*/

#inc#include <GL/glut.h>

GLfloat gfPosX = 0.0;

GLfloat gfDeltaX = .001;

void Draw() {

glClear(GL\_COLOR\_BUFFER\_BIT);

glColor3f(1.0, 1.0, 1.0);

glBegin(GL\_LINES);

glVertex3f(gfPosX, 0.25, 0.0);

glVertex3f(1.0 - gfPosX, 0.75,0.0);

glEnd();

glFlush();

gfPosX += gfDeltaX;

if (gfPosX >= 1.0 || gfPosX <= 0.0) {

gfDeltaX = -gfDeltaX;

}

glutPostRedisplay();

}

void Initialize() {

glClearColor(0.0, 0.0, 0.0, 0.0);

glMatrixMode(GL\_PROJECTION);

glLoadIdentity();

glOrtho(0.0, 1.0, 0.0, 1.0, -1.0, 1.0);

}

int main(int iArgc, char\*\* cppArgv) {

glutInit(&iArgc, cppArgv);

glutInitDisplayMode(GLUT\_SINGLE | GLUT\_RGB);

glutInitWindowSize(250, 250);

glutInitWindowPosition(200, 200);

glutCreateWindow("Animation");

Initialize();

glutDisplayFunc(Draw);

glutMainLoop();

return 0;

}

//OUTPUT:



