

```
1: /*
2:  * pong.c
3:  * This is a simple double buffered program.
4:  * Use the left and right arrow keys to move the paddle.
5:  * Use the home key to re-center the paddle and return the ball "home."
6:  * The beginnings of the classic video game.
7:  */
8:
9:
10: #include <GL/glut.h>
11: #include <stdlib.h>
12:
13:
14: GLdouble paddleX = 0.0;
15: GLdouble paddleDelta = 1.0;
16:
17: GLdouble ballX = 0.0;
18: GLdouble ballY = 0.0;
19: GLdouble ballXDelta = 0.5;
20: GLdouble ballYDelta = 0.5;
21:
22: GLuint ball;
23:
24:
25: void display(void)
26: {
27:     glClear(GL_COLOR_BUFFER_BIT);
28:
29:     // Render paddle.
30:     glColor3f(1.0, 1.0, 1.0);
31:     glBegin(GL_QUADS);
32:         glVertex2d(paddleX - 10.0, -50.0);
33:         glVertex2d(paddleX - 10.0, -45.0);
34:         glVertex2d(paddleX + 10.0, -45.0);
35:         glVertex2d(paddleX + 10.0, -50.0);
36:     glEnd();
37:
38:     // Render ball.
39:     glColor3f(1.0, 0.0, 0.0);
40:     glPushMatrix();
41:     glTranslatef(ballX, ballY, 0.0);
42:     glCallList(ball);
43:     glPopMatrix();
44:
45:     glutSwapBuffers();
46: }
47:
48:
49: void init(void)
50: {
51:     GLUquadricObj *qobj;
52:
53:     glClearColor (0.0, 0.0, 0.0, 0.0);
54:     glShadeModel (GL_FLAT); // Probably unnecessary.
55:
56:     // Create the ball.
57:     ball = glGenLists(1);
58:     qobj = gluNewQuadric();
59:     glNewList(ball, GL_COMPILE);
60:         gluDisk(qobj, 0.0, 5.0, 72, 1);
61:     glEndList();
62: }
63:
64:
65: void reshape(int w, int h)
66: {
67:     // Probably needs to be fixed.
68:
69:     glViewport (0, 0, (GLsizei) w, (GLsizei) h);
70:     glMatrixMode(GL_PROJECTION);
71:     glLoadIdentity();
72:     glOrtho(-50.0, 50.0, -50.0, 50.0, -1.0, 1.0);
```

```
73:     glMatrixMode(GL_MODELVIEW);
74:     glLoadIdentity();
75: }
76:
77:
78: void keyboard(int key, int x, int y)
79: {
80:     // Limited here by keyboard repeat rate.
81:
82:     switch (key)
83:     {
84:     case GLUT_KEY_LEFT:
85:         paddleX -= paddleDelta;
86:         break;
87:
88:     case GLUT_KEY_RIGHT:
89:         paddleX += paddleDelta;
90:         break;
91:
92:     case GLUT_KEY_HOME:
93:         paddleX = 0.0;
94:         ballX = ballY = 0.0;
95:         break;
96:     }
97:     glutPostRedisplay();
98: }
99:
100:
101: void idle(void)
102: {
103:     ballX += ballXDelta;
104:     ballY += ballYDelta;
105:     glutPostRedisplay();
106: }
107:
108:
109: /*
110:  * Request double buffer display mode.
111:  * Register "special" input callback functions for arrow and home keys.
112:  */
113:
114: int main(int argc, char** argv)
115: {
116:     glutInit(&argc, argv);
117:     glutInitDisplayMode (GLUT_DOUBLE | GLUT_RGB);
118:     glutInitWindowSize (250, 250);
119:     glutInitWindowPosition (100, 100);
120:     glutCreateWindow (argv[0]);
121:     init ();
122:     glutDisplayFunc(display);
123:     glutReshapeFunc(reshape);
124:     glutSpecialFunc(keyboard);
125:     glutIdleFunc(idle);
126:     glutMainLoop();
127:     return 0; /* ANSI C requires main to return int. */
128: }
```