CPSC 414, Project 0

Out: Mon 8 Sep 2003
Due: Mon 16 Sep 2003

For this project, you will compile and run a simple OpenGL demo application on the CPSC 011 workstations, then make some changes to it. The template code is available from http://www.ugrad.cs.ubc.ca/ cs414/Downloads/openGLDemo.tar, which contains the two files `openGLDemo.cpp` and `Makefile`. The three steps of this project are:

1. compile and run the template on the CPSC 011 workstations
2. modify DisplayCallback so that the program draws a dodecahedron instead of a triangle. A dodecahedron is a 12-sided Platonic solid with each face an identical pentagon (5-sided surface where all edges are of equal length).
3. add an event handling callback so the object changes color every time the user clicks the mouse in the window.

In this week's labs, the TAs will discuss the high-level structure of an OpenGL/GLUT program.

Handin: do not hand in, this project will not be graded.

```
openGLDemo.cpp:

#include <GL/glut.h>

void DisplayCallback()
{
    // clear the color buffer
    glClear( GL_COLOR_BUFFER_BIT );

    // draw a triangle into the buffer
        glBegin( GL_TRIANGLES );
          glVertex3f( 0.0f, 0.5f, 0.0f );
          glVertex3f( -0.5f, -0.5f, 0.0f );
          glVertex3f( 0.5f, -0.5f, 0.0f );
        glEnd();

    // draw the buffer to the screen
    glutSwapBuffers();
}


int main(int argc, char **argv)
{
    // create window and rendering context
    glutInit( &argc, argv );
    glutInitDisplayMode( GLUT_RGB | GLUT_DOUBLE );
        glutInitWindowSize( 640, 480 );
        glutCreateWindow( ``openGLDemo'' );

    // register display callback
        glutDisplayFunc( DisplayCallback );

        glMatrixMode( GL_PROJECTION );
    glLoadIdentity();
        glMatrixMode( GL_MODELVIEW );
        glLoadIdentity();

    glViewport( 0, 0, 640, 480 );

    // pass control over to GLUT
        glutMainLoop();

        return 0;       // never reached
}
```