Computer Graphics

Lecture 08

Introduction to OpenGL

OpenGL is a software interface to graphics hardware. This interface consists of about 150 distinct commands that you use to specify the objects and operations needed to produce interactive three-dimensional applications.

OpenGL is designed as a streamlined, hardware-independent interface to be implemented on many different hardware platforms.

The OpenGL Utility Library (GLU) provides many of the modeling features, such as quadric surfaces and NURBS curves and surfaces. GLU is a standard part of every OpenGL implementation. Also, there is a higher-level, object-oriented toolkit, Open Inventor, which is built atop OpenGL, and is available separately for many implementations of OpenGL.

Include Files

#include <stlib.h>
#include <GL/glut.h> (This includes gl.h and glu.h)
#include <stdio.h> (if using C I/O)
#include <math.h> (if using C math library)

For portability, replace the second statement by the following:

```
#ifdef __APPLE__
#include <GLUT/glut.h>
#else
#include <GL/glut.h>
#endif
```

Basic Main

```
/* Program entry point */

int main(int argc, char *argv[])
{
    glutInit(&argc, argv);
    glutInitDisplayMode (GLUT_SINGLE | GLUT_RGB);
    glutInitWindowSize (250, 250);                  //window screen
    glutInitWindowPosition (100, 100);              //window position
    glutCreateWindow ("Program1");                  //program title
    init()                                          //initialize
    glutDisplayFunc(display);                       //callback function for display
    glutReshapeFunc(resize);                        //callback for reshape
    glutKeyboardFunc(key);                          //callback function for keyboard
    glutMainLoop();                                 //loop

    return EXIT_SUCCESS;
}
```

OpenGL Display Modes

GLUT_RGBA	Bit mask to select an RGBA mode window. This is the default if neither GLUT_RGBA nor GLUT_INDEX are specified.
GLUT_RGB	An alias for GLUT_RGBA.
GLUT_INDEX	Bit mask to select a color index mode window. This overrides GLUT_RGBA if it is also specified.
GLUT_SINGLE	Bit mask to select a single buffered window. This is the default if neither GLUT_DOUBLE or GLUT_SINGLE are specified.
GLUT_DOUBLE	Bit mask to select a double buffered window. This overrides GLUT_SINGLE if it is also specified.
GLUT_ACCUM	Bit mask to select a window with an accumulation buffer
GLUT_ALPHA	Bit mask to select a window with an alpha component to the color buffer(s).
GLUT_DEPTH	Bit mask to select a window with a depth buffer.
GLUT_STENCIL	Bit mask to select a window with a stencil buffer.
GLUT_STEREO	Bit mask to select a stereo window.

Initialize Viewing Values

```
void init()
{
  glMatrixMode(GL_PROJECTION);
  glLoadIdentity();
  glOrtho(-2.5,2.75,10.5,0, -1.0, 1.0);
}
```

glLoadIdentity — replace the current matrix with the identity matrix

glMatrixMode

void glMatrixMode(GLenum mode)	
GL_MODELVIEW	Applies subsequent matrix operations to the modelview matrix stack.
GL_PROJECTION	Applies subsequent matrix operations to the projection matrix stack.
GL_TEXTURE	Applies subsequent matrix operations to the texture matrix stack.
GL_COLOR	Applies subsequent matrix operations to the color matrix stack.

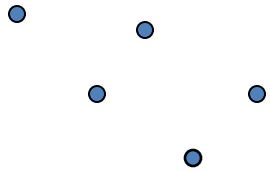
Display

```
glPushMatrix();
glRotatef(theta, 0.0f, 0.0f, 1.0f);
glBegin(GL_TRIANGLES);

    glColor3f(1.0f, 0.0f, 0.0f);
    glVertex2f(0.0f,   1.0f);
    glColor3f(0.0f, 1.0f, 0.0f);
    glVertex2f(0.87f,  -0.5f);
    glColor3f(0.0f, 0.0f, 1.0f);
    glVertex2f(-0.87f, -0.5f);

glEnd();
glPopMatrix();
SwapBuffers();
```

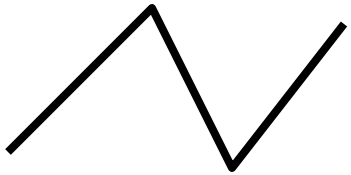
OpenGL Geometric Primitives

Value	Meaning
GL_POINTS	individual points
GL_LINES	pairs of vertices interpreted as individual line segments
GL_LINE_STRIP	series of connected line segments
GL_LINE_LOOP	same as above, with a segment added between last and first vertices
GL_TRIANGLES	triples of vertices interpreted as triangles
GL_TRIANGLE_STRIP	linked strip of triangles
GL_TRIANGLE_FAN	linked fan of triangles
GL_QUADS	quadruples of vertices interpreted as four-sided polygons
GL_QUAD_STRIP	linked strip of quadrilaterals
GL_POLYGON	boundary of a simple, convex polygon

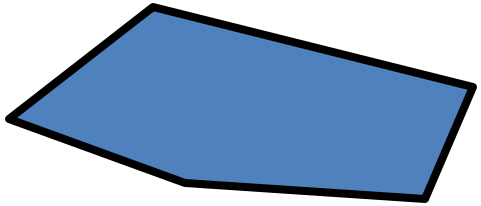
OpenGL Geometric Primitives



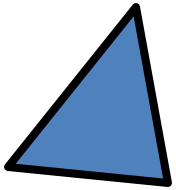
Points



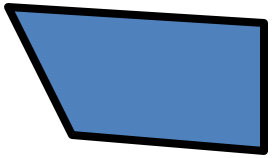
Lines



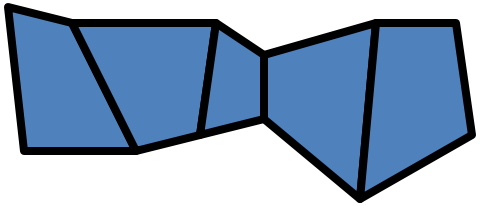
Polygon



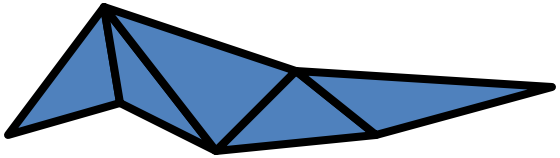
Triangle



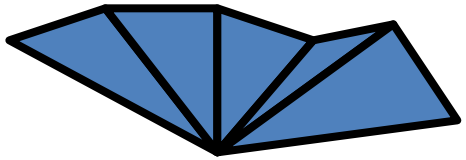
Quad



Quad Strip



Triangle Strip



Triangle Fan