CS334: Assignment #0 – Warm-up

Out: Jan 16, 2008
Back/Due: Jan 25, 2008

Objective:
This objective of this assignment is a simple warm-up program to help setup your programming and graphics environment. This assignment will require you to setup a shell OpenGL, GLUT, and GLUI program using Windows. It is to your benefit to write the program modularly and with a clean setup so as to facilitate subsequent assignments. You have one week but it should take you much less time.

Summary:
The assignment is to implement a program which draws a simple 2D screen saver like program. The "screen-saver" consists of a pair of points bouncing within the confines of the windows and a line being drawn between the points. To draw the line, simply draw the pixels in between the two points. While you will use OpenGL, GLUT, and GLUI, the only OpenGL command you can use is "glDrawPixels". The first point should start at a random position and with a random (but reasonable velocity vector). The points should reflect off the edge of the window and continue moving indefinitely.

Specifics:
(0) Setup your programming environment to compile OpenGL/GLUT/GLUI programs. You might have to install GLUT on your machine. GLUI is a simple library you link with and OpenGL is most likely already installed. Instructions were given in class on how/where to obtain GLUT/GLUI (it is free).
(1) Upon starting the program, a window should 'pop-up' for drawing and a GLUI GUI should appear. The GUI should at least contain the buttons "start", "stop", "reset' and "reverse". The buttons will perform their indicated functionality. Reset implies generate a new random position and initial velocity vector. Reverse implies multiplying the velocity by -1. The initial velocity vector and magnitude must be random but "reasonable" – e.g., each time the program runs, the points will move in a different initial direction and speed.
(2) For this assignment, you may use the necessary GLUT+GLUI commands but for OpenGL you may *only* use glDrawPixels() which directly draws pixels to the screen. To draw a "line", you will have to implement a for-loop to fill-up a pixel buffer and then display it using glDrawPixels.

Code Template:
To help with this assignment, below is a template you must use to setup your program.

```
#include <GL/gl.h>
#include <GL/glut.h>
#include <GL/glui.h>
#include <stdio.h>
#include <math.h>

void display(void)
{
```

```
        // screen saver logic

        // copy my buffer to real framebuffer
        glClearColor(0,0,0,1);
        glClear(GL_COLOR_BUFFER_BIT);
        glMatrixMode(GL_PROJECTION);
        glLoadIdentity();
        glOrtho(0, width, 0, height, 1, -1);
        glMatrixMode(GL_MODELVIEW);
        glLoadIdentity();
        glRasterPos2i(0, 0);
        glDrawPixels(width, height, GL_RGB, GL_UNSIGNED_BYTE, mybuffer);
        glutSwapBuffers();
        glutPostRedisplay();
}

void MakeGUI
{
        // make my GUI using GLUI
}

int main(int argc, char* argv[])
{
        glutInit(&argc, argv);
        glutInitDisplayMode(GLUT_RGB | GLUT_DOUBLE);
        glutInitWindowSize(512, 512);
        WindowID = glutCreateWindow(argv[0]);
        MakeGUI();
        glutDisplayFunc(display);
        glutMainLoop();
        return 0;
}
```

To give in the assignment, please use Purdue's Blackboard Vista. The TA will provide details in PSO if you are not familiar with this system. Your complete project (project files, source code and precompiled executable) is due by 10:29am on the due date. It is your responsibility to make sure the assignment is delivered on time. **Hint: don't wait until the last moment to hand in the assignment**.

For grading, the program will be run with no command line parameters and the code will be looked as well for proper functionality. If the program does not compile, zero points will be given.

```
C:\ <your program>
<window pops up, square bounces, etc>
<ctrl-c to end>
```

If you have more questions, please see myself or the TA.