

CS535: Assignment #0 – Warm-up

Out: August 30, 2005

Back/Due: September 6, 2005

Objective:

This objective of this assignment is a simple warm-up program to help setup your programming and graphics environment and to begin implementing basic vector, point, and matrix manipulation routines. This assignment will require you to setup a shell OpenGL and GLUT 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 bounces a square (2D) from within a window. The square will be drawn in wireframe (i.e., not a solid colored square but rather four line segments). Starting with an initial random direction vector and velocity, the ball should move within the window until it hits a window boundary. The ball velocity vector should be reflected off the wall. The ball should continue moving in the reflected direction until the next collision and so forth.

Specifics:

(0) You will have to implement a simple library that permits operations on points, vectors, and matrices. This will be used for the animation, collision detection, collision response, etc.

(1) The square must “touch” the wall before bouncing – it cannot bounce in mid-air.

(2) The initial velocity vector and magnitude must be random but “reasonable” – e.g., each time the program runs, the ball will move in a different direction and speed.

(3) For this assignment, you may *not* use OpenGL/GLUT functions or any other downloaded libraries for geometric calculations, manipulations, rendering, or any part of this assignment except as indicated in the below template. You must implement yourself all necessary routines.

Example Screen Dump:



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 <stdio.h>
#include <math.h>

void display(void)
{
    // collision and animation logic
    // draw my box

    // 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();
}

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

To give in the assignment, email the TA, Denny Wong (wang124@purdue.edu), your complete project (project files, source code and precompiled executable) by 4:30pm on the due date. It is your responsibility to make sure the email is delivered/dated before it is due. If you wish to receive confirmation of receipt, please ask the TA by email. If you email at 4:29:59pm on the due date and ask for confirmation and the confirmation is negative -- it will be considered late. **Hint: don't wait until the last moment to hand in the assignment.**

IMPORTANT: in your email message, please include a short message saying "This is Assignment #X" and attach your assignment inside a .ZIP file -- do not email a .EXE file as spam filters will probably filter-out the attachment.

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.