OpenGL/GLUT Example Program
CSC 461: Undergraduate Computer Graphics

This document describes how to compile the example OpenGL program (available on the course web page) under Windows XP/Vista with Visual Studio 2008.

When you begin a new project in Visual C++ (our standard programming environment), you will need to perform the following steps to ensure your program can find and use the OpenGL and GLUT APIs:

1. In the File→New... submenu, choose Project

2. In the New Project dialog, choose Visual C++ as the project type, then select Win32 Console Application from the Templates list, then select a Name, Location, and Solution Name for your new project

3. In the Win32 Application Wizard dialog, choose the Application Settings option, deselect the Pre-compiled header checkbox, select the Empty project checkbox, then select Finish to create the project

4. Add opengl.cpp to the Source Code directory in the Solution Explorer panel

5. Choose the Project→Properties menu item, then expand the Configuration Properties subtree

6. Expand the C/C++ subtree, choose the General option, and in the Additional Include Directories option in the right-hand panel (topmost option), enter the directory where you’ve stored GL/glut.h

7. Expand the Linker subtree, choose the General option, and in the Additional Library Directories option in the right-hand panel (9th option), enter the directory where you’ve stored glut.lib

8. Choose Build→Build Solution to compile and link your project

9. Copy glut.dll into the directory where your project’s executable is located (usually this is in the Debug subfolder of the project directory you specified when you created your project)

10. Choose Debug→Start Debugging to run your program

The source code for the example program can be downloaded from the course web page. You are encouraged to copy it onto your machine, then compile and link it to make sure you can get an OpenGL/GLUT program running. You can also modify the program, to see how different OpenGL commands affect the appearance of the axis (for example, try switching from perspective projection to orthographic projection, and note how the display of the axis changes when you translate, zoom, and rotate it).