

CS334/ECE30834: Assignment #0 – Cook it!

Out: January 14, 2022

Back/Due: January 24, 2022, 11:29 AM

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 programming environment for this and future assignments, using OpenGL with FreeGLUT or Qt5 on Windows or Linux. 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. The FreeGLUT framework is bare-bones, but easy to setup, whereas the Qt framework is more powerful, but also more difficult. If you want to use a different framework (e.g., GLFW, SFML, etc.), contact the TAs with your request (note that you still must use OpenGL within your framework).

Summary:

The assignment is to implement a program which draws a simple screen saver like program. The “screen-saver” consists of bouncing a loaded 3D object inside an imaginary box. You may use whatever OpenGL commands you wish but probably you just need to add some simple matrix math operations. The object should start at a random location and move at a “reasonable speed”. You do not need to make the collisions with the walls precise, nor the response physically correct, just plausible.

Specifics:

- (0) Start with the templates from the course website (choose FreeGLUT or Qt). Both templates implement the same features (the files `glstate.*`, `mesh.*`, and `util.*` are identical). You may develop under Windows or Linux (the TAs do not have a Mac to grade with – if your only dev environment is MacOS, please realize that you may encounter difficulties and the TAs may not be able to help). For either template, see the README.txt for details on compiling and running.
- (1) Compile and run the template. Left-click and drag to rotate the object, and use the scroll wheel to zoom in and out. On the FreeGLUT template, right-click to see a menu of different objects to view. In Qt, there is a dropdown to the left. **Read the code to understand how it works.** Don't hesitate to ask questions! Then, make the changes below.
- (2) When the program starts, a random initial location and velocity for the loaded object should be assigned. You may use the GUI to add options, (e.g., a reset button). The templates come with a simple `.obj` loader. You may also hardcode to load one object file.
- (3) The template code does not automatically update the screen each frame, only when the view changes. For a moving object, you will need to repeatedly tell the windowing framework to redraw the screen. Use the `idle()` function with `glutPostRedisplay()` in FreeGLUT, or use a `QTimer` with `timeout 0` for this in Qt (see the comments in `glview.cpp`).

- (4) One way to implement the wall-bouncing is when the object centroid passes a wall, snap it to the wall and reflect the velocity vector about the wall (e.g., if a “vertical wall in the YZ plane, then swap the sign of the x-component of the velocity).
- (5) You should replace the template code functionality with your implementation. For example, it may not make sense to have camera controls for a screen-saver, so you can disable the mouse movements.

Turn-in:

To give in the assignment, please use Brightspace. Give in a zip file with your complete project (project files, source code, and precompiled executable). The assignment is due BEFORE class on the due date. It is your responsibility to make sure the assignment is delivered/dated before it is due. If you wish to receive confirmation of receipt, please ask by email in advance.

Don't wait until the last moment to hand in the assignment!

For grading, the program will be compiled on Linux and run from the terminal (with Visual Studio as a fallback – please try to avoid platform-specific code (e.g., don't `#include <windows.h>`)), run without command line arguments, and the code will be inspected. If the program does not compile, zero points will be given.

If you have more questions, please ask on Piazza!