

Spring 2008

22C:196 Advanced OpenGL Rendering

Assignment 5

Due: The morning of Thursday, May 15th at 9:45 am

Please Note: This is non-negotiable, as you will all present your work to the class at 9:45 in lieu of a Final Exam. Due to summer travel I must have my grading done on Thursday May 15th.

Goal: Spend time improving and debugging your OpenGL room demo. If you did an alternative project for Homework 4, you should talk to me about what you need to do on this assignment.

Problem 1 (20 points): Fix up your Homework 4 so it is easy to use and looks good. You should invest some time in software engineering, perhaps cleaning your rendering code or improving on my framework to make it easier to use. Some of these 20 points will be discretionary on my part. Remember this assignment is essentially replacing your final, so I expect this to be a nice program you'll be proud to show off to friends. You'll need to demonstrate it to myself and the other students in the class during the Thursday Final Exam period (9:45–11:45). Things you are required to have in your program:

- Mirrors must not be 100% reflective and glass objects must not be 100% refractive (so the viewer can clearly see the shape of the mirror or glass object). This reflective or transmissive parameter should be controllable via a keyboard or menu interface.
- If you have not done so, spend some time building a new scene that is different than those I provided you. You can, however, leave complex things (e.g., mirrors) in the same spot or even leave the walls intact. However, I'd like to see everyone have a completely unique room. You might consider changing the textures, changing the shape of the room, using different types of wall decoration (instead of my relatively boring "paintings" and photos), and finding other geometry for the scene. You may wish to read the whole assignment before deciding on a room design.
- Add some dynamic objects to your scene. Please make sure it makes sense in the context of a "room." For instance, a dynamic ceiling fan would be appropriate, but a teapot orbiting the light would not be.
- Add controls for moving the user around the room and rotating the eye in place (e.g., not using a trackball). In my framework this should be relatively easily accomplished, since all the camera specifics (position, parameters, etc) are encapsulated in a single Camera class.
- Make sure that during movement the user cannot get "lost" by flying outside the walls; Some of the internals of the room should be visible at all times.
- Fix any features you "implemented" in Homework 4 that you did not get fully working.
- Use textures that are appropriate for their function. If you have some all-time favorite image, say from your favorite TV show, please do not use it as "wallpaper" for your room.
- **Additional requirements may be added shortly as I grade Homework 4 and identify specific problems.**

Problem 2 (30 points): Extend your demo program by implementing additional features. You may implement anything from Homework 4 that you *did not* turn in previously and is not required above for Problem 1, as well as anything from the following list. Groups must do more than 25 additional points; please contact me to determine how many points are required for full credit.

(5 points): Find 5 or so new complex 3D objects to add to your scene to make the room more rich and realistic. You could, for instance, add some couches, chairs, tables, lamps, etc. This will involve some searching for objects online, and perhaps some experimentation to convert them to an appropriate format (someone suggested using Blender). Make sure to use appropriate materials, textures, or shaders for all objects. This set of 5 models may not include any 3D models I have given you (though you may use those in addition). Organize these models in an "interesting" way in your scene.

- (5 points):** If you choose to add a number of additional complex models, you may choose to augment my framework to allow it to use an object's built-in material types.
- (10 points):** Add a textured spotlight to your scene (i.e., like an overhead projector). This can either be the only object light in the scene or it can be in addition to your other lights.
- (10 points):** Implement the Gooch NPR shading model, and shade a complex model with this technique (either a new model, or one I've provided).
- (20 points):** Implement real-time hatching, and apply it to a complex model. If you decide to implement this, you should come talk to me.
- (20 points):** Implement procedural texturing and add a toggle to switch between standard lighting and a procedural marble texture for one complex object. As an optional addition to this, you may consider replacing your floor with a (different color) procedurally generated marble. (This would be really cool with a partially reflective floor, one of the options mentioned in Homework 4).
- (5 points):** Add a "painting" to the wall that is not static, but instead shows some dynamic scene. (Some examples: a dynamic game of life or a few frames from a repeatable video clip to create a cheap "virtual TV.")
- (10 points):** Change your reflective mirror so it blurs reflected objects different amounts based on how far they are from the mirror (e.g., close objects are not blurry, distant objects are blurred a lot). If you want to do this, please come talk to me to discuss specifics.
- (30 points):** Implement a GPU-based volume renderer and draw a volume object in the scene. This will likely be slow, so you may wish to add a toggle to switch back to a simple stand-in (say a cube) with standard OpenGL lighting.
- (10 points):** Utilize OpenGL selection so you can click on any complex object in your room to select it and then drag it around the room. You need not perform collision detection, but when dragging, the object must stay directly under the mouse. You should be able to do this in my framework relatively easily as every Group object has a groupXForm matrix for just this purpose (Note: *xform* is short for transform).
- (30 points):** Add a curved reflector to your room scene. You should talk to me about this idea.
- (30 points):** A very open ended choice: turn your demo into a simple game.
- One idea for this might be for your mirrored room to serve as the start point in a maze in a basic 3D first-person PacMan where your goal is to run around and pick up the floating dots.
 - This is very subjective project idea, so you may not get full credit if you put it very little effort. For the "PacMan" idea, for instance, in order to get 30 points you might need either a sizable and interesting maze, a good 2D numerical timer/scoreboard floating in front of the 3D display, 3D enemy "ghosts" that chase you in some basic way, or interesting dots that pulsate or reflect/refract the scene.
 - It may be worth your time to discuss any ideas you might have with me.