



# Final Exam Review

## Computer Graphics 1

4003-570-01 (ugrad)

4003-760-01 (grad)

Spring Quarter 20063

## Reading Coverage

Computer Graphics with OpenGL (Third Edition), Hearn & Baker

All Chapters from Midterm (Ch 1-6, 11)

All course notes.

The following chapters will be covered more heavily on the test as they occurred after the midterm:

Chapter 13 – Computer Animation

Chapter 14 – Hierarchical Modeling

Chapter 7 – Three Dimensional Viewing

Chapter 12 – Color Models and Color Applications

Chapter 10 – Illumination Models and Surface Rendering Methods

Chapter 8 – Three Dimensional Object Representation

Chapter 9 – Visible surface detection methods

## Major Topics

- 2-D and 3-D Transformations
- 3-D Viewing
- Color
- Lighting
- Materials / Textures
- Curves and Surfaces
- Particle Systems
- Visible surface detection methods

## Things to Know

- Know all material from the Midterm exam.
- 2-D Transformations: Translate, Scale, Rotate Shearing
  - What are the transformation equations/matrices for scale, rotate (clockwise vs. counterclockwise), and translate?
  - Be able to apply transformations.
  - Why do we need matrices/composite matrices?
  - Does it matter in which order you apply the transformations?
  - How to scale or rotate an item about a point (x,y) that is not the origin?
- 3-D Transformations

- Understand how the matrices are constructed for scale, rotates and translates
  - What are homogeneous coordinates and why are they required for 3-D transformations?
  - What the cross and dot products of matrices mean with special values like 0
- World to camera/eye coordinate transformation – why?
  - Why two coordinate systems?
  - Information needed to establish camera/eye position.
  - Steps in world to camera/eye transformations.
- Viewing Parameters
  - Camera/eye position, “look at”, “up”
  - Purpose of each parameter.
  - How coordinates are specified and in what coordinate system.
- Visible surface detection methods
  - Object vs. image space
  - Algorithms used for visible surface detection
  - Z-buffer and A-buffer concepts
  - BSP trees: what they are and how they’re used, be able to draw a simple tree
- 3-D View Volumes
  - What’s the difference between Parallel/Orthographic and Perspective Projections?
  - How to specify in OpenGL
  - Coordinate systems used for each
  - Effects of varying transformations on different projects
- OpenGL Graphics Pipeline
  - Understand the entire pipeline from 3-D World Coordinates to 2-D Screen Coordinates
  - Know the purpose of the Modelview, Projection and Viewport matrices
- Animation
  - How/why are stacks manipulated in order to achieve 3-D animation?
  - Be able to follow a coding example that pushes and pops matrices.
- Color
  - Know these definitions: hue, saturation, lightness/value, halftoning, additive vs subtractive reproduction
- Light
  - What is ambient, diffuse reflection, specular reflection and shininess?
  - What is the role of normals in computing light for a given surface?
  - Know the general concepts of the Phong Illumination Model (not details)
  - What is the difference between a position and directional light source?
  - How is a spotlight simulated in OpenGL?
  - How do transformations of the modelview matrix effect lighting?
- Materials
  - What are color materials and how do they work in conjunction with light?
  - What is texture mapping and how is it done in OpenGL (generally)?
  - What is texture space and how are texture coordinates defined (generally)?
  - Are textures passed through the same pipeline as geometric primitives?
  - In OpenGL, what does it mean if a texture is resident and how are different textures bound to different world objects?
  - Texture Filtering: What is meant by magnification vs. minification
  - Know these terms: texel, clamping, repeating, mipmapping
- Curves & Surfaces
  - Know the definitions: curve, surface, parametric equation, control point, continuity, evaluator, NURBS
  - What are the differences between the 4 types of continuity when joining 2 curves?
  - You don’t need to know the OpenGL calls, but generally understand how a curve/surface is constructed and how materials/textures are applied to that surface
  - Recognize the common curves used in graphics and why they are commonly used
- Particle Systems
  - What is the difference between a particle and a particle system?
  - How are particles represented?