

22C:151 Computer Graphics Fall 2009 – Midterm Review

The midterm will be at 9:30-10:45 on Thursday October 15

The exam will be open book & open note. No calculator will be needed, though you may bring one (especially if simplifying things like $\sin 60^\circ$ or $\cot 45^\circ$ is difficult).

The exam will mostly consist of short answer format questions. The basic idea of the exam will be to see if you understand the concepts – determining if you can code is the job of the homeworks, not the exam. At most, you will have to interpret code, not write it. None of the questions will be “trick” questions, and many of them will have simple answers. If you know the concepts well, you may be done in half an hour. Those of you who need to refer to the book for most questions will have difficulty finishing in the allotted time.

Things you may wish to review should you feel rusty:

- What is a PPM file?
- Basics of monitors. What are R, G, B, and A? How do these relate to human vision?
- Why is OpenGL called a state machine? What is a state machine?
- Basic GLUT functionality. Callback routines.
- Basic OpenGL commands for drawing
- Primitives allowable in OpenGL
- What do various parameters to `glBegin()` mean?
- Bresenham’s line algorithm
- The difference between Bresenham’s algorithm and the incremental approach (the simple approach we examined first)
- Smooth v.s. flat shading
- Interpolation
- Triangle rasterization
- Boundary conditions for triangle rasterization
- Color interpolation for triangles
- Homogeneous coordinates and their use
- Why do we need/want a homogeneous coordinate system?
- Transformations, matrices, how they’re implemented in OpenGL, and ordering of transformations
- Why do we need 4×4 matrices in OpenGL? What do the extra row and column do?
- Perspective
- What is a matrix stack? Why is it useful?
- What is a z-buffer? What does it store? How do you enable it in OpenGL?
- OpenGL lighting. What is the process of enabling it? What parameters are required? How are these parameters affected by OpenGL state?