CS 4731: Computer Graphics
Final review

Emmanuel Agu

Exam Overview

- Thursday, Oct. 16, in-class
- Will cover lectures 13-24 (12 lectures)
- Can bring:
  - One page cheat-sheet
  - Calculator
- Will test:
  - Theoretical concepts
  - Mathematics
  - Algorithms
  - Programming
  - OpenGL knowledge (program structure and some commands)

What really am I Testing?

- Understanding of on concepts (NOT only programming)
- That you can plug in numbers by hand to check your programs
- Understanding of programming (pseudocode/syntax)
- That you did the projects
- That you understand what you did in projects

General Advise

- **Read your projects** and refresh memory of what you did
- **Read the slides**: worst case – if you understand slides, you're more than 50% prepared
- I've placed actual .ppt slides on web to make it clearer
- Focus on **Mathematical results, concepts, algorithms**
- Plug numbers: calculate by hand
- Midterm was calculation-intensive, final exam more algorithmic
- Should be able to **predict subtle changes** to algorithm.. What ifs?..
- **Past exams**: my exams will be most similar to last year's exam
- Every lecture has references. Look at refs to focus reading

Grading Policy

- Will still do **ALL** grading myself
- Give you all the points, take away only what I have to
- In time constraints, laying out outline of solution gets you healthy chunk of points
- Try to write something for each question

General Notes!

- This exam is more algorithmic than first
- Need to read and understand subtleties of algorithm
- Some concepts permeate multiple topics
  - E.g. Interpolation
  - Make sure you know those topics

Projection matrices

- Projection matrices
  - Perspective projection
  - Orthographic projection
  - How to derive them, plug numbers
  - If parameters changed, how would derivations change?

3D clipping, viewport transf, illumination

- 3D clipping (Liang-Barsky algorithm)
- Viewport transformation in 3D
- Illumination models
  - Light types (point, extended, etc)
  - Global vs local illumination
  - Ambient, diffuse, specular
  - Phong light model
  - OpenGL lighting, material commands
  - Shading (flat, gouraud, phong, etc)

Texturing, HSR, Shadows

- Texturing
  - Flat, curved surface
  - OpenGL texturing commands
- Hidden surface removal
  - Z-buffer
  - OpenGL HSR commands
  - Others (painters algorithm, backface culling, etc)
- Shadows
  - Shadows as texture (projection)
  - Shadow buffer approach
  - Umbra, penumbra

Raster graphics

- Raster graphics
  - Pixel manipulation
  - RGBpixmap class
  - Image manipulation (dissolve, alpha blending, etc)
  - Line drawing (simple DDA, Bresenham's algorithm)
  - Filling pixel regions (recursive fill, using coherence, etc)
  - Polygon-defined regions
  - Antialiasing (prefiltering, supersampling, openGL, etc)

Curves, Color

- Curves
  - Representations (implicit, explicit, parametric, etc)
  - Continuity
  - Interactive curve design (interpolation, etc)
  - De Casteljau algorithm
  - Bezier curves, B-splines, NURBs
- Color
  - Hue, saturation, wavelengths, etc
  - The eye, cones, tristimulus theory, etc
  - Color spaces (CIE, RGB, HSV, etc)
  - Color quantization, gamma correction, device color gamuts

Final Words

- You've all worked very hard
- Graphics is a hard topic…
- Symptoms of hardness:
  - No book does good job explaining all topics
  - Professor is a slave driver, tough to understand
  - TA's don't help enough
  - Friends can't help much either
- Hopefully, you've learned a lot
- If you've worked hard, tried, I will curve to make you pass
- If you're doing well, finish strong!!
- Good luck!