Notes on final exam.

120 minute comprehensive written test. See the midterm exam review sheet for topics up until midterm. We estimate that about 40% of the final will consist of topics from the midterm.

Texture mapping
Texture coordinate space

Suppose the quad below (two triangles) has texture coordinates (0, 0) at A, (1, 0) at B, (0.5, 1) at C, and (0, 1) at D. Sketch what the 2 x 2 checkerboard texture looks like when mapped to that quad.

![Checkerboard texture](image.png)

Image formats (RGB, RGBA, ...), loading pixels

Issues
1. How to determine texture coordinates
2. How to sample from a texture
3. How to use the sampled value

Wrapping options (repeat, clamp)

Filtering options – nearest, linear

Sampling and antialiasing concepts, nearest vs linear filtering
- Show/draw/explain how linear filtering works
- Why does a texture tend to look blurred with linear filtering?

Minification, magnification, mipmaps
- What artifacts do you typically see as a result of magnification?
- What is the purpose of mipmaps?

Mipmapping options – nearest/nearest, nearest/linear, linear/nearest, linear/linear
- Explain in words what these mean

Using texture value (replace, modulate, blend)
- Write a fragment shader that uses the texture value to modulate the intensity of the specular highlight

Procedural textures
- Write a fragment shader that generates a 2 x 2 checkerboard pattern (assume you have an `in` variable representing the texture coordinate)

Bump mapping
Normal maps
- How are normal vectors encoded into a texture as RGB?
- Why do normal maps typically look blue?

Height maps
- Explain in pseudocode how you’d convert a height map to normal map

Tangent (local) coordinates
- Why do we need local coordinates?

Using the inverse TBN matrix
- Assuming you are given normal and tangent vectors for each vertex, write a shader pair for doing bump mapping using the usual Phong lighting calculation in the fragment shader. You can assume that the given normal and tangent are orthogonal and normalized.
  a) Set it up to do the lighting calculation in local coordinates.
  b) Set it up to do the lighting calculation in eye coordinates. Why is this likely to be less efficient?

Cube maps, sampling from an OpenGL cube texture
- Suppose you have a model that doesn’t have texture coordinates. Show how you could apply any texture to the model using the idea of cube mapping. Assume the model’s center is at the origin.

Reflection (environment mapping) using cube map

Framebuffer objects, attachment points
Renderbuffers
Rendering to a texture
- Implement a “scene within a scene”, i.e., an image of the scene itself appears on a quad within the scene

Shadow mapping
- Explain, in words and a sketch, what steps are involved.
Rendering to a depth buffer, using sampler2DShadow

Suppose that GLSL did not have the special sampler type sampler2DShadow. Describe how you would do shadow mapping using an ordinary 2D texture.

Antialiasing strategies

Explain some of the issues with shadow mapping and the tricks/techniques used to reduce artifacts.

Concept of a matrix stack

Hierarchical models

Give an example of a hierarchical model based on one type of primitive solid (sphere, cylinder, etc.). Write down the stack operations you’d use to render it, assuming that the top matrix on the stack is always used as the model matrix.

Perlin noise (Gradient noise)

Suppose you have two textures; texture unit 0 is bound to an image texture and texture unit 2 is bound to a 2D texture containing noise values in [-1, 1] stored as values [0, 1] in the red channel. Write a fragment shader that uses noise to modulate the transparency of the first texture before applying lighting.

Write a fragment shader that uses a 3d noise texture to modulate the surface color between red and yellow.

Describe/sketch how gradient noise is generated

Octaves, 1/f noise

Why is it called “1/f”?
Suppose you have a function float noisel(float x) that returns a one-dimensional noise value for any given real number x, using a period of 1 and an amplitude of 0.5. Write a function that sums four octaves of noise starting with period 4 and gives a result in the range [-1, 1].

Turbulence

Geometry shaders

Where does a geometry shader fit in the overall OpenGL pipeline?

Billboarding

What is it for?

view plane aligned

view point aligned

Given the view matrix along with the up direction for world coordinates, derive the transformation for a view point aligned billboard a) preserving the world up direction, b) not preserving the world up direction