

Live Lecture Chat Window

November 6, 2024

14:15:00 For the Final Project, would collision/hit detection, spawning projectiles on command, and mobility be enough?

Yes, if there is sufficient graphics to show it off.

14:28:23 Is it difficult to model reflections? I was thinking of chrome ball moving on a checkerboard.

You could probably write a shader that would map a moving checkerboard onto the sphere that would make it look like it is being reflected.

14:29:16 Has anyone ever modeled a maze? Like a ball that can move when you move the maze. Similar to a pinball machine, I guess?

Yes, that is a cool idea. Something like manipulating the eye position to move through the maze?

14:32:11 Do you know if it is possible with the lighting knowledge we have to imitate "blacklighting" and objects that glow in ultraviolet light?

Anything with glow is tricky. There is a technique used for it called "Bloom".

14:33:52 Would something like an object traveling through a hyperspace/wormhole tunnel work (assuming lots of animation, etc. to make it look good)?

I think that would be a very cool FP. Maybe mimic the relativity distortion you would get as you approach the speed of light?

14:35:02 Can we give multiple proposals if we haven't decided on one?

Yes, if you also give me a sense about how the tie will be broken.

14:48:29 I'm a bit confused about the difference between the `GL_ARRAY_BUFFER` and `GL_ELEMENT_ARRAY_BUFFER`? `GL_ARRAY_BUFFER` is the list of vertices, right?

The `GL_ARRAY_BUFFER` is the table that holds the vertices. The `GL_ELEMENT_ARRAY_BUFFER` is the table that holds the indices of the vertices that are to be connected.

14:54:24 In `glDrawElements`, what is the final argument?

It is from which vertex you are starting the drawing. Game/simulation companies like to pack multiple objects into the same vertex buffer, so that argument can tell OpenGL to start at an arbitrary location in the table.

15:00:07 Are shaders the ancestors of CUDA?

CUDA is a compute-only system to accelerate AI, Simulation, etc. However, NVIDIA did create a shader language called Cg at one time. I think they abandoned it when HLSL (DirectX) and GLSL emerged.