WebGL Lecture Room

wfjv99

Implementation

I assembled a basic lecture room consisting of chairs, tables, a whiteboard and a floor, using triangle primitives (wfjv99.htm). The chairs can be rotated dynamically using the 'o' and 'p' keys and the height of the whiteboard can be adjusted dynamically using the keys 1-9 (“induce changes to dynamic objects”). The tables and floor are static. Also there are three views (“different orientations and positions of the virtual camera”) of the room available, accessible via the keys 'q', 'w' and 'e'. There is directional and ambient lighting present, of arbitrary colour. Three textures are used in my lecture room: wood, metal and sandstone.

The primitives instances of my scene are: two chairs and a table, the floor, and the whiteboard. The two chairs and a table are duplicated four times and arranged using two for loops which iteratively places them into two columns and two rows. There is only one instance each of the floor and the whiteboard.

Limitations

My program uses multiple buffers (a buffer for the vertex positions of each individual object, another buffer each for the positions of the normal vectors, another buffer for the texture coordinates). It would technically be more efficient to combine all these buffers into one buffer and use element indexing. There is also some duplication of pushing identical buffers to the GPU e.g. for each chair, even though they all use the same basic vertex coordinates.

I could not figure out how to get my program to wait until after the image resources had been loaded before using them as textures, so all textures are loaded as hidden <img> elements on the page. This is the reason the scrollbar is quite long.

Screenshots

1.jpg – same as view 'q', shows an above an off-to-side view of the lecture room

2.jpg – same as view 'w', shows a view of the lecture room from behind the whiteboard, down the middle of the room

3.jpg – same as view 'e', shows a view of the lecture room from the back and slightly off to the side. Also the whiteboard in this view has had its height increased and the chairs are rotated to the left of the viewer.

Public Domain Code

For matrix manipulation I used the glMatrix library (supplied as gl-matrix.js), which is very similar to the library used in the practicals. The code I have used in wfjv99.htm was adapted from examples at learningwebgl.com, including the vertex and fragment shaders which are mostly based on the vertex and fragment shaders from Lessons 5 and 7. I have commented the code and annotated types where pertinent according to JSDoc convention. The loadTexture() function is almost the exact same as Lesson 7 from learningwebgl.com. The general pattern for drawing the scene (filling a buffer, putting number_of_items and item_size attributes on the buffer object, adjusting the relevant matrices before resetting them, drawing using GL.TRIANGLES, etc) is based on Lessons 1-4 from learningwebgl.com.

References

Lecture slides 1-4

http://learningwebgl.com – Lessons 1-5 and 7.

https://www.youtube.com/watch?v=6-9XFm7XAT8 – helpful video explaining some general OpenGL concepts

http://nehe.gamedev.net/article/glsl_an_introduction/25007/ - understanding the GLSL language

http://glmatrix.net/ - glMatrix library

http://usejsdoc.org/ - JSdoc

http://abduzeedo.com/files/originals/w/wood02-texture-old.jpg – wood512.jpg texture

https://www.cs.auckland.ac.nz/~georgy/research/texture/bunch/synthetic/metal00_g.bmp.jpg – metal512.jpg texture

http://i751.photobucket.com/albums/xx152/vexx32/sandstone256.png – floor256.jpg texture