Lab 01 Shader Reboot

**Fufillment**

* All of the requirements are met with minimal extra conveniences added (BetterDargon can fly around and look at the scene, some useful things like movement and camera speed are config values)

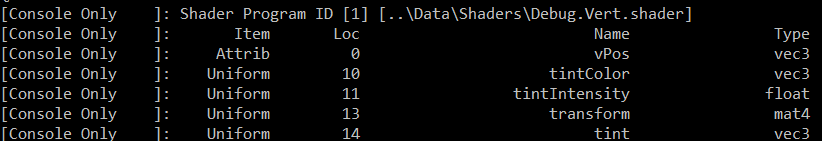
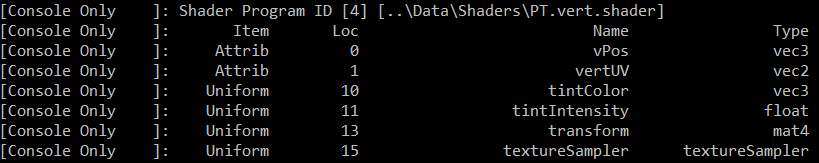
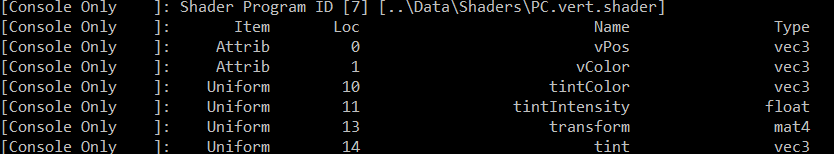
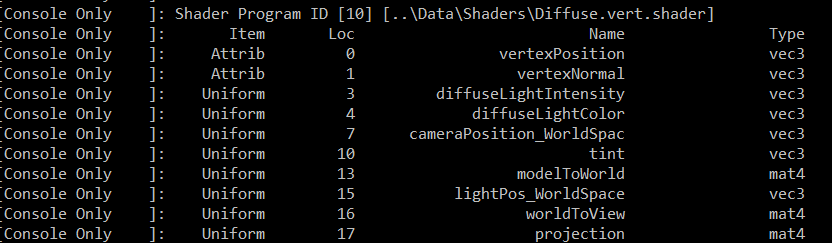
**Execution**

* I do not believe there should be any unexpected requirements for running

**Controls**

* Pressing the X key will close the application
* Pressing the P key will pause the application, pressing it again will un-pause the application
* Pressing numpad 0 will re-read the config file
* Pressing M, L, T or C will dump engine info to the console, this is pretty much exclusively used for debugging
* Pressing W will rotate BetterDargon to the left, S will rotate him to the right
* Pressing A will tilt BetterDargon forward, D backward
* Pressing Q will roll BetterDargon to the left, E to the right
* Holding space will move BetterDargon forward, in the direction he is facing, releasing will halt movement
* Right clicking and dragging the mouse will turn the camera around BetterDargon
* Scrolling in or out with the mouse wheel should zoom the camera accordingly, up to a minimum or maximum distance

**Screenshots**

* This is the output for the debug shader from the shader parser
* 
* This is the output for the PT shader, which was used for Regular Dargon in the demo, but I let it parse for testing purposes anyway
* 
* This is the output for the PC shader, which BetterDargon is using in the demo
* 
* This is the output for the diffuse shader, used by the teapots
* 
* Following is a screenshot of all three objects and all three lights
* 
* A close up of the first teapot (reflectivity = (0.0f, 0.0f, 0.9f), light color = (1.0f, 1.0f, 1.0f))
  + Note that when hit by a white light, it reflects only blue!
* 
* A close-up of the next teapot (reflectivity = (0.9f, 0.9f, 0.0f), light color = (1.0f, 1.0f, 1.0f))
  + Note that when hit by white light, it reflects only red and green light, making it appear yellow
* 
* A close up of the final teapot (reflectivity = (0.75f, 0.75f, 0.75f), light color = (1.0f, 0.0f, 0.0f))
  + Note that when hit by a red light, it reflects a portion of the red
* 

**Post-Mortem**

* .
  + I have realized, upon putting a little more time into using my engine, that the Shader Uniform thing is actually cooler than I first thought. It is really cool because I can now log what values are being passed into my Shaders from one place. This is an amazingly useful debugging technique as it makes it clear what values are not being sent/what values are being sent incorrectly.
  + I really liked how I was able to make my engine a little better along the way with this lab
    - Not only did I fix two or three bugs, but I added a material class, did the uniform thing, and refactored some code
      * I feel like, if I get a little bit of this done each lab, my engine could end up in a much better place at the end of the quarter, however, I do feel a little bad knowing that I now have a superior version of my own engine than the version I gave the others, and I worry they will run into similar issues I did with my engine.