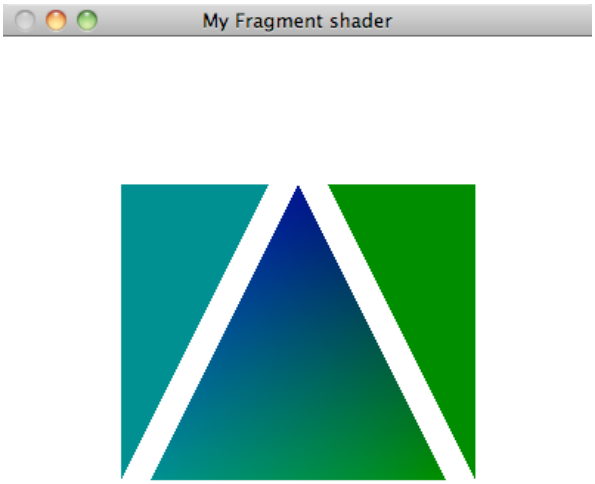
Lab #3 – cpe471
Further introduction to OpenGL and GLSL – Fragment shaders

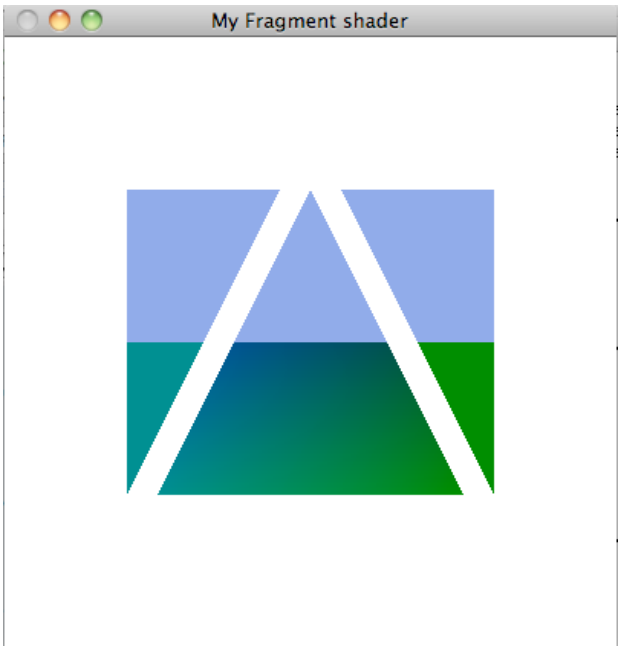
Today we will explore adding a fragment shader to our graphics programs and practice window coordinate transforms and fragement shader effects.

Starting from the code on the web page for lab 3, please read the "InstallShader" function and notice what is different about this version compared to past versions (in Lab 1).

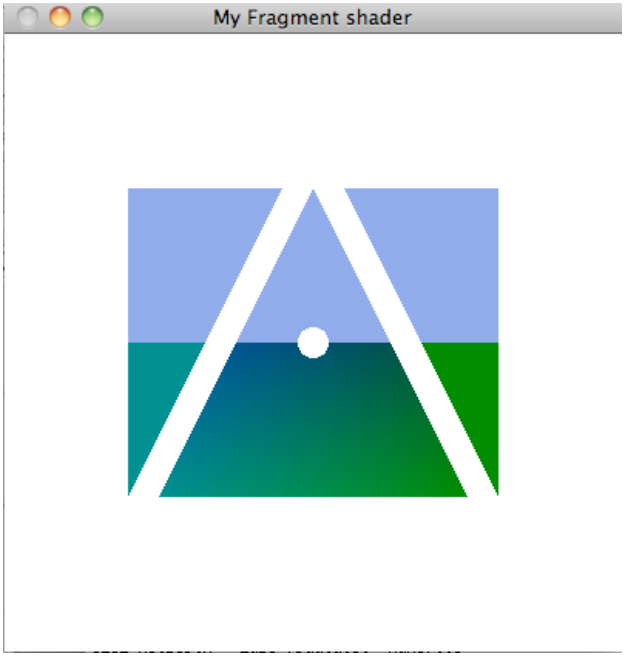
Also notice that this lab comes with two files lab3_vert.glsl and lab3_frag.glsl
Without any modifications, the program should draw something like this:



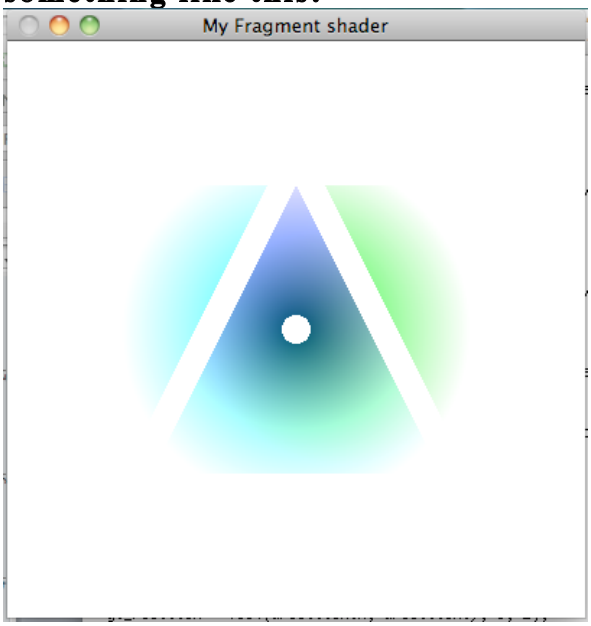
Task 1:
Modify the fragment shader so that any fragments having y values greater then half the window size are drawn as light blue instead of mixed greenish-blue colors (use GLSL gl_FragCoord.x and gl_FragCoord.y to test the position, which are in window coordinates!) and then set gl_FragColor. Your output should look something like:



Task 2:
Modify the program and shader so that for the center triangle, any pixels that are less then 10 pixels away from the center of the window are discarded. You can use the following GLSL calls: 'discard' and 'distance'. Your results should look something like this:



Task 3:
Finally, modify the fragment shader, such that all other pixels fade into white in a circular pattern around this central point (hint think about "adding" a white color to the fragment color as it gets further away from the center of the triangle). Your result should look something like this:



Lastly, think about how you could have this type of fade effect work for each triangle individually (something like this):