CMP301 Project Notes

Things to do

1. Fix the multiple lights on the scenery so it looks better DONE
2. Rewrite the input normals so it’s completely my own DONE
3. Make sure I’m doing the best way to combine shaders DONE
4. Change the terrain shader to change the colour/texture at certain y-coordinates(done by others) DONE
5. Blend two textures using the height-based shader DONE
6. Add the remaining lights needed DONE
   1. POINT DONE
   2. DIRECTION DONE
   3. SPOTLIGHT DONE
7. Add in appropriate shadows IN PROGRESS
   1. MAKE SURE THERE ARE ENOUGH SHADOWS FOR LIGHTS WITH DIRECTIONS DONE
   2. MAKE SURE THE SPOT LIGHTS CAN PRODUCE A SHADOW DONE
   3. MAKE SURE ALL THE STUFF IS NOT JUST DEBUG BUT TAKES IN ALL THE LIGHTS CLEAN IT UP!!!!!!
   4. CHECK THE LIGHTS PROPERLY
   5. APPARENTLY ALL GEOMETRY MUST CAST SHADOWS ASK FOLK
8. Add specular and attenuation values where needed appropriately(Done in labs before) DONE
9. Use the shader headers DONE
10. Add post process bloom effect/ Hazy blur DONE
11. Add in the use of a Render to target DONE
12. Add in model(will be covered in class) DONE
13. Add in dynamic tessellation (In a book 13.5 and another book chapter 9) IN PROGRESS
    1. GOT TO TESSELATE A NON-FLAT SURFACE USE WHAT MICHEAL GAVE
14. Make sure appropriate GUI elements are added
15. Comment the code
16. Write Report
17. Add in the use of the geometry shader
18. Add in the use of the compute shader
19. Add in other stuff if needed
20. IF TIME add in grass shader

Notes

* Need to check if it’s even worth using a circle mesh as I’d have to make it (so depending how quickly I can do that I might just leave it)
* Might be a good plan to maybe just assume were gonna be square for this one
* Got to include the geometry shader in the project (should try putting the code from the slides into the code and seeing what comes out)
* Got it working but is not required but that is how you would go about making the grass
* Need to find a way for the sine wave to start from the centre and work its way out to the edges
* Gerstner Wave looks sick
* This would be a great way to show of vertex manipulation if I can get it working and that’s a very big IF
  + Really need to check if it would be worth the work to get it working because it would be a big risk, but it could drastically improve my grade, but I need to check that
* Need to ask how to properly include tessellation ,shadows ,lighting and vertex manipulation in one scene.
* Get to work on the dynamic tessellation
  + Can work with what I’ve currently got because the main clump of stuff is done within the hull and domain shaders
* I should really be more thoughtful about what is required So get the base stuff first and then add in the tessellation and the extra stuff like Gerstner waves and grass shader as its not required to pass.
* So instead, I should get the attenuation and the specular values working first and then work on the lighting for the waves along with the specular and attenuation values
* GOT A BLOOMING SHADER HEADER YAY!
* Not too sure how well the normals are working on the waves will have to look into that
* The water shader is getting one less light passed into it than the terrain shader for some reason
  + Seems to be better but could be improved but I don’t think it would take very long for a lecturer to get it to look quite right but it could take me a while and honestly its good as it is.
* Need to add in the shadows to the scene. What do I need to do?
  + 1. Create a depth buffer/map
    - Create a depth map for each light that is on and active
    - Might be better to create an array of depth maps to be passed in
  + 2. Create the shadow shader
    - Remember when setting the shadows that you are passing in lights that specifically have a directional value otherwise it will not work
    - Might need to think longer about it because if a light needs a direction for it to work were gonna need to work out which lights need to have a shadow and then put those ones in(Or work around it)
    - Not sure immediately how I would apply that to geometry might be better to apply it to stuff like models and stuff
    - Could try and cast the shadow over the water mesh. NO DOES NOT WORK
* For some reason the water mesh texture is always the same as the main mesh texture which is really odd. Fixed it
* Need to get a new height map though
* Will also need to raise the water mesh a little bit, so it doesn’t clip through the main mesh
* Get started on the bloom effect next
  + Here’s a cool lens flare post process effect <https://www.shadertoy.com/view/4sX3Rs>
  + Another cool one here <https://www.shadertoy.com/view/lsBGDK>
  + Need to find out how to properly translate some of the shadertoy stuff to directx11 I’m pretty sure they linked the blur example at some point which we used in directx11 so start looking there
  + There’s also a question mark button which details this stuff
  + FUCK IT ADD A SMALL HAZEY BLUR AND CALL IT A DAY YOU’VE ALREADY DONE THE HARD WORK FUCK IT
  + FUCK I CAN’T DO THAT!
  + So, this is what I’ve figured out about shadertoy and how to translate it maybe?
* Text

  Description automatically generatedSome interesting stuff on how to do the water
* There is also this paper <https://mmikk.github.io/papers3d/mm_sfgrad_bump.pdf>
* Also, Illya Knows aswell
* Test the normals of the water mesh using a spotlight that I need to include anyway
* Could move the number of lights to headers so I wouldn’t have to change each appearance of it
* Spotlight is lighting up scene
* Water meshes normals are still out of whack
* As it turns out the normals are not working. FUN – Looks like the other one now
* Looks like the specular and attenuation stuff will need to be reworked as it didn’t seem like it was working anyway so that’s not as huge as having to rework the lighting
  + Specifically, its looking like the attenuation and speculation doesn’t work with the directional light
  + Problem is that the specularity doesn’t seem to want to work with the directional lighting
  + Need to add the specular value not multiply
* Ended up reducing the normals of the lerp dividing them by 50.
* Ended up reducing both the normals by dividing them by 50. Going to need to explain that
* But that as it turns out completely flattens the normals more or less so its technically incorrect
* Going to need a flat surface for the shadows to work I think. Put the models on some floating platform and then have the shadows be created and set on that platform
* Needed to cross multiply the water meshes normals not lerp you dumb
* From my understanding the depth of field effect can be created by having a blur that is only applied to stuff far enough away on the z-axis
  + Bokeh depth of field might be what I need
  + This here <https://stackoverflow.com/questions/3946902/rendering-depth-in-hlsl> might be the answer as it is how to visualize the depth meaning that this will give depth a value which we can then use to blur stuff beyond a certain depth
  + Okay so I do calculate the depth for the shadow map as that’s what that does I need to find a way to use that to blur stuff based on the cameras position and then blur it. So, I need to combine how the shadow map gets the depth of stuff and then blur it if it is past a certain point
  + Something is wrong with the depth of field shader it only returns a single colour
  + Need to give it the camera positional data aswell
  + The shaders depth value is always 1 it never changes
  + I have no idea how to get this working. I’m going to have to ask again
  + Make the depth of field shader more similar to the depth shader
  + AND NOW ITS BROKEN AGAIN FUCK IT I DO NOT CARE GET ANY POST PROCESSING EFFECT YOU CAN GET
  + FUCK IT IM DOING SHADOWS TOMMOROW AT LEAST THAT’S SOMETHING DIFFERENT
* Getting some help from Erin on the depth of field stuff and I feel like it is getting closer to being what I need it to be. However, it is still unfortunately not working but I have emailed her about it.
  + I need to create a new depth shader that will apply the same height map calculations as the terrain shader to a depth map
  + And another one that can perform the height map manipulation the same as the water
  + Models and terrain may need different settings so I should pass something in to let it know what it is
  + Setup the GUI so it can be turned on and off
* Got models which are able to cast shadows properly in E9 or week 9 shadows which need to be moved to my project
* When doing the dynamic tessellation do the same thing you did with the normals and make it your own and make sure you understand fully how it works (Currently trying to get it to work in E8 aka week 8 tessellation).
  + The domain shader stays the same as it was previously for standard tessellation as it does for dynamic tessellation
  + Need to put the tessellation in the terrain shader and water mesh shaders for it to work and if that really doesn’t work have a cube and dynamically tesselate that.
  + Tessellation needs it to be a specific mesh, so I’ll just have it as another thing in the scene. So, I need to add in the special tessellation mesh
* I need to check if the specular values are working properly
  + Best way to properly check would be with either a cube or sphere mesh
* Create an update function that will you know apply all the changes from the Gui and hold the timer stuff
* The lighting and the shadows must be combined. The shadow shader doesn’t have the correct lighting and the light shader doesn’t have the correct shadows
  + COMBINE THEM
* Looks better when it is not saturated. Don’t saturate at once do it as you get the lighting

What I need to do now!(The stuff in orange will most likely take me the longest and will be the most difficult):

1. Put in the models on a higher flat plane to allow for proper shadows (Can do it on heightmap)
2. Need a post processing effect implemented
3. Need to add dynamic tessellation
4. Add control through the use of a Gui and comment the code
5. Write the report