

# CMPM 163 W2018

## Game Graphics & Real-time Rendering

### Homework 2 (100 pts) – Due Sunday, 2/18 at 12noon

All code will be uploaded to GitHub (or another repo), it should include a short “ReadMe” describing the project, along with one or more screenshots of the project. Assignments A and B should run from a website. Lucas will provide the instructions for submitting the homework. This time around, 10% of your grade will be based on creativity. You can also earn a total of 10 points of extra credit.

#### A. Outdoor 3D scene - (45 pts for completion and 5 pts for creativity)

Using Three.js and GLSL shaders, create a scene with at least the following elements:

- A height map that is used to represent terrain. The height map can be created from an image, or procedurally (eg, from a noise function or some other method). Additionally, you will apply one or more textures to the terrain.
- A cube map that is used to represent the sky or distant objects.
- Bodies of water (eg, where the height map is less than sea level) that reflect the sky.
- Be able to move the camera through the scene.
- Use dat.gui to control some parameters of the scene.

Extra credit (5 pts):

- In addition to having the water reflect the sky, also use a Fresnel effect to refract the light so that we can also see the bottom of the bodies of water.

#### B. Abstract scene using particles and noise - (45 pts for completion and 5 pts for creativity)

Using Three.js and GLSL shaders, create a scene with at least the following elements:

- A particle system that mimics some natural phenomenon – fire, clouds, smoke, etc – that changes or animates over time. You can utilize the GPUParticleSystem.js class, or create your own.
- A noise function that provides realistic features of natural phenomena. You can use the library at <https://github.com/stegu/webgl-noise/>, or you are welcome to find another approach of your own.
- Use dat.gui to control some of the noise or particle system parameters.

Extra credit (5 pts):

- Use textured point sprites in your scene.