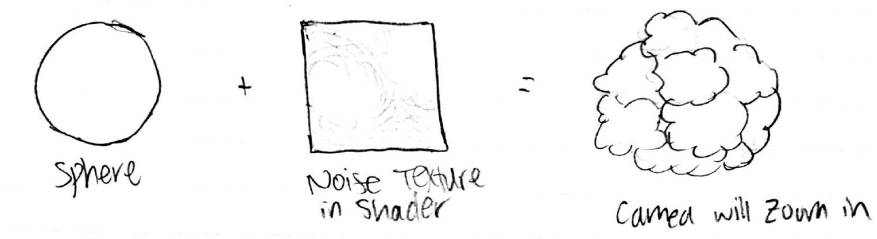
Part 2.1

I think the visual effect for cloud simulations will require camera movement, noise, light calculations, and ray marching. The simulation would simulate the view of the camera moving through many layers of clouds, which would be similar to an airplane flying through clouds. The light calculations would affect the lighting, color of the sky, and clouds to create a realistic effect. Ray marching would calculate how close the nearest surface is from the object creating this cloud simulation.

The shader for cloud simulations may contain properties that control the camera and have variables that can be updated through scripts. Interpolation, or the smoothstep function, in Unity would create natural blending of different type of clouds, such as dense and light, and result in depth. For lighting, a phong shader would most likely achieve the desired lighting calculations.

To implement cloud simulations in Unity using HLSL shaders, I would create a shader, possibly through Unity's Shader graph, that have properties that work with the camera, noise, light, and ray marching, and take in an input noise texture to give the clouds randomness. The shader would also take in color input to increase customizability, such as a slider to change the color of the clouds or sky. This would be applied to a static object such as a sphere because the shader will contain properties that alter the camera's movement. This movement can be controlled to move through a sphere and result in a cloud simulation. The shader will work with a script to update for camera movement.

Sketch



Images of the Effect

Shader applied to sphere

Camera starts zooming in

Part 2.2

Group Members:

Jeanette Mui
Yuming Li
Brian B.
Jeremy Green