CSC 305 Assignment 2 Requirements

Due March 1, 2019 11:55 PM

In assignment 2, we will create a small procedural terrain based on Perlin noise, texture the terrain based on height, and place some objects on the terrain following the local normal vector of the terrain.

Assignment 2 and 3 are divided into two parts. Correctly completing all features the Basic Requirements part will award you a grade up to 70% of the assignment. To get 100%, you must selectively implement a few features in the Advanced Requirements part, whose solutions are not completely covered in lectures and requires original research. The Advanced Requirements are open-ended, and you are welcomed to propose and work on your own preferred features. You should come to discuss your own idea with the TA or the instructor first; and a point value will be assigned to your proposed feature after we discuss it.

When you submit your assignment 2 and 3, you need to include a list of feature points that you claim you have completed, and your TA will ask you questions about all feature points you listed during the grading lab session. Downloading an existing Unity package/asset for a feature is not considered as completing that feature by yourself.

Assignment 2 – Basic Requirements (70%)

1. Generate a square Perlin noise patch with sampling resolution 250x250 and frequency 10. (Hint: if you aren't sure what was the result of your Perlin noise generator, you can visualize this 250x250 2D float array as an image using the code from Assignment 1) (20%)
2. Generate a plane mesh that's divided into a 250x250 grid, with each grid cell's rectangle conventionally divided into 2 triangles. (10%)
3. Using the Perlin noise patch generated in step 1 as a height map, elevate vertices generated in (2) to form a procedural terrain. (10%)
4. Write a vertex/fragment shader pair of the terrain, which can texture the procedure terrain with at least 4 different textures according to height. (Hint: You must compute the height in the object space of the terrain model, which means if you move your terrain mesh up and down in the Scene Editor, the texture distribution does not change). (10%)

5. Implement the Phong-Blinn reflection model in the fragment shader, treating the texture colour as surface albedo (Hint: you need camera position in the fragment shader for specular reflection). (10%)
6. Place a few 3D models onto the procedural terrain, elevate them as the terrain goes up, and rotate them so they seem to be upright on their local surface. (Hint: you need to calculate the local normal of the procedure terrain). (10%)

Assignment 2 – Advanced Requirements (up to 30%)

- Make a moveable camera so you can fly through the scene and examine your terrain from any point of view. (5%)
- Scatter more models on to your mini-terrain, for example, other houses and little towers. When place then randomly, write code to ensure that they don't overlap each other. (5%)
- Extend the procedural terrain to infinity (or to a very large area such that it looks like infinite) while keeping a reasonable rendering performance. (10%)
- Add reflections to the water that not only reflects the skybox but also the mountains/placed 3D models above the water. (10%)
- Add ripple effect to the water. (5%)
- Desaturate everything in the scene according to a fragment's distance to the camera, such that far away terrain & objects appear grey-ish and things closer to the camera have more vibrant colour. (Hint: you need a method to adjust a pixel's saturation only, without changing its hue or luminance.) (10%)
- Add moving sun and clouds in the skybox, create a day/night cycle. (10%)
- Create atmospheric rendering effects for the environment – adjust the sunlight colour, cloud colour, reflection and shadows according to the simulated time of the day and the sun's height from the horizon. (10%)
- Add interactive terrain editing features: when a user clicks on part of the terrain, s/he can use the mouse to drag the terrain up and down. Surface features such as water definition, texturing and small models must be updated in real time according to the editing. (10%)
- Feel free to propose your own ideas and improvements for Advanced Requirements. Your instructor/TA will assign a point value after discussing the idea with you.