

We decided to implement multiple fractal patterns in glsl, and texture them to the given objects. Ben worked on the fractal patterns and demo pipelines implementation, Stephen worked on the demo curves implementation. Unfortunately, we were not able to get our glsl files working with our own render passes, but we are able to display our fractal work within the given drawPhong\_multi\_forward\_mrt\_fs4x file. This project is a real-time effect that is applied to the objects in the scene, and we consider this to be an example of an intermediate real-time effect, such as was described in the project requirements. As for the other requirements, we did not complete this fractal effect in an earlier assignment, and we would use both vertex (fractalPattern\_vs) and fragment (drawFractalPattern\_fs) shaders if our own render passes had worked. Our project still uses vertex and fragment shaders (drawPhong\_multi\_forward\_mrt\_fs4x and passTangentBasis\_transform\_instanced\_vs4x), but we were forced to add to existing shaders instead of using our own.

GLSL:

Mandelbrot fractal: drawPhong\_multi\_forward\_mrt\_fs4x, lines 267-284

Mandelbrot fractal projection: drawPhong\_multi\_forward\_mrt\_fs4x, lines 288-305

Julia fractal: drawPhong\_multi\_forward\_mrt\_fs4x, lines 308-325

Fractal noise: drawPhong\_multi\_forward\_mrt\_fs4x, lines 160-234

These are also in our drawFractalPattern\_fs shader, but we were unable to generate our own render passes

CSS:

Demo\_Curves.h: added target names and pass name (lines 114, 146-150)

DemoState.h: added fractal demo state shader program (lines 361-363), added framebuffer for fractal(line 403)

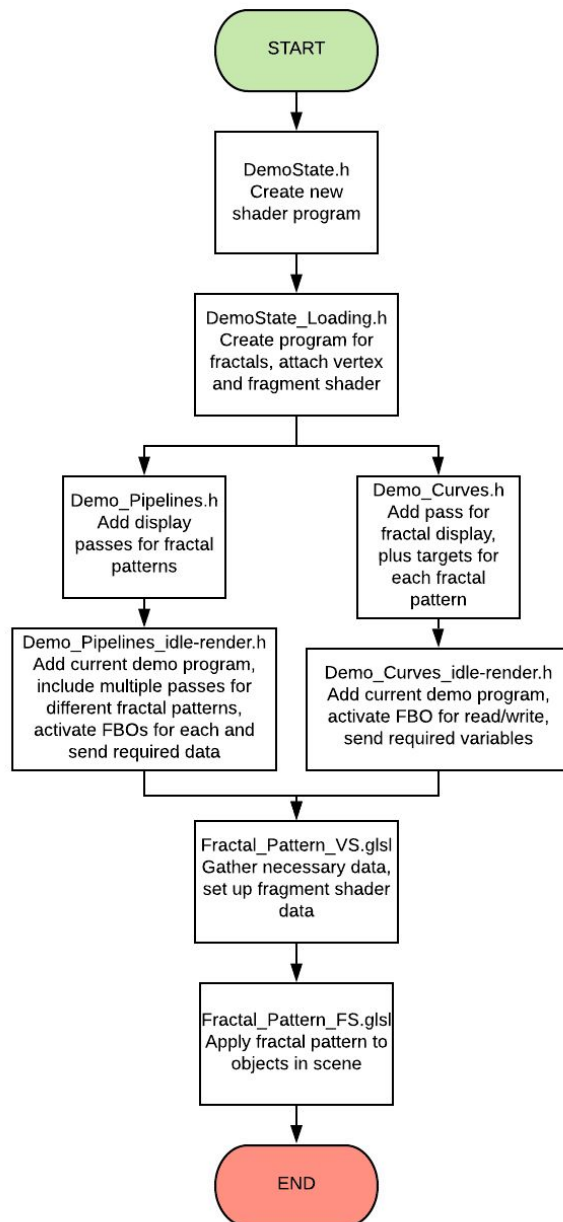
Demo\_Pipelines.h: added additional render passes and targets for fractals (lines 116-119, 158-160)

Demo\_Pipelines\_idle\_render.c: added render pass in pipeline section (lines 774-799, 839-841)

Demo\_Curves\_initialize.c: initialized target index and count for fractals (lines 61, 76)

Demo\_Curves\_idle\_render.c: added pass and target names for fractal mode(lines 93, 119-122, 138), added text for targets displayed (lines 119-122), added fbo for fractals (lines 293, 311), added render pass in curves section (lines 648-658, 702)

DemoState\_loading.c: added fractal shaders to shader list(lines 436,483, 511, 550), made program and attached shaders to it(lines 733-736), added fbo for fractal(lines 1030-1033)



UML: