robinmamie / **2019_cs341** Private

Add visualization of Perlin 3D to assignment 9

Browse files

master

robinmamie committed 13 days ago 1 parent eb5eea1 commit 2009802334c0888fa87bd90b1d59845461e281ec

Showing **12 changed files** with **128 additions** and **12 deletions**. Unified | Split

1 assignment_9/scripts_win/run_fbm.bat

Load diff
This file was deleted.

1 assignment_9/scripts_win/run_map.bat

Load diff
This file was deleted.

1 assignment_9/scripts_win/run_marble.bat

Load diff
This file was deleted.

1 assignment_9/scripts_win/run_perlin_1d.bat

Load diff
This file was deleted.

1 assignment_9/scripts_win/run_perlin_2d.bat

Load diff
This file was deleted.

1 assignment_9/scripts_win/run_turbulence.bat

Load diff
This file was deleted.

1 assignment_9/scripts_win/run_wood.bat

Load diff
This file was deleted.

1 assignment_9/scripts_win/terrain.bat

Load diff
This file was deleted.

15 assignment_9/src/render/ShaderViewer.cpp

```
25          , viewer_velocity(0)
26          , viewer_scale(1.0)
27          , should_redraw(true)
28  {
29  }
30
97                                  viewer_velocity.y +=
change;
98                                  break;
99                          }
100                 }
101         }
102
222         shader_to_display.use();
223         shader_to_display.set_uniform("viewer_position",
viewer_position);
224         shader_to_display.set_uniform("viewer_scale",
viewer_scale * aspect_ratio);
225
226         fullscreen_quad.draw();
227         glBindFramebuffer(GL_FRAMEBUFFER, 0);
```

```
25          , viewer_velocity(0)
26          , viewer_scale(1.0)
27          , should_redraw(true)
28  +       , time_3d(0.0)
29  {
30  }
31
98                                  viewer_velocity.y +=
change;
99                                  break;
100                         }
101 +                       case GLFW_KEY_E:
102 +                       {
103 +                               time_3d += 1e-2;
104 +                   should_redraw = true;
105 +                               break;
106 +                       }
107 +                       case GLFW_KEY_R:
108 +                       {
109 +                               time_3d -= 1e-2;
110 +                   should_redraw = true;
111 +                               break;
112 +                       }
113                 }
114         }
115
235         shader_to_display.use();
236         shader_to_display.set_uniform("viewer_position",
viewer_position);
237         shader_to_display.set_uniform("viewer_scale",
viewer_scale * aspect_ratio);
238 +       shader_to_display.set_uniform("time", time_3d);
239 +       std::cout << "Time: " << time_3d << std::endl;
240
241         fullscreen_quad.draw();
242         glBindFramebuffer(GL_FRAMEBUFFER, 0);
```

1 assignment_9/src/render/ShaderViewer.h

```
71          vec2 viewer_position;
72          vec2 viewer_velocity;
73          float viewer_scale;
74  };
75
76
```

```
71          vec2 viewer_position;
72          vec2 viewer_velocity;
73          float viewer_scale;
74  +       float time_3d;
75  };
76
77
```

9 assignment_9/src/shaders/display_fbm.frag

```
 in vec2 v2f_tex_coords;
 out vec4 f_color;      // Final color output produced by fragment shader.
                                        // (Can name this anything you want...)


-float perlin_fbm(vec2 point); // Implemented in noise.frag


 void main() {
-
-       float noise_val = perlin_fbm(v2f_tex_coords) + 0.5;

        f_color = vec4(noise_val, noise_val, noise_val, 1.0);
-}
```

```
 in vec2 v2f_tex_coords;
 out vec4 f_color;      // Final color output produced by fragment shader.
                                        // (Can name this anything you want...)
+uniform float time;

+float perlin_fbm_3d(vec3 point); // Implemented in noise.frag

 void main() {
+
+       float noise_val = perlin_fbm_3d(vec3(v2f_tex_coords, time)) + 0.5;
        f_color = vec4(noise_val, noise_val, noise_val, 1.0);
+}
```

107 assignment_9/src/shaders/noise.frag

```
        return mix(white, brown_dark, alph);
 }
```

```
        return mix(white, brown_dark, alph);
 }

+// constants for water
+const float freq_multiplier_water = 11;
+const float ampl_multiplier_water = 0.3;
+
+// #### PERLIN 3D ####
+// Source: https://mrl.nyu.edu/~perlin/noise/
+#define N_PERM      512
+#define N_PERM_HALF 256
+
+const int permutation[N_PERM_HALF] = int[N_PERM_HALF]
(151,160,137,91,90,15,
+
131,13,201,95,96,53,194,233,7,225,140,36,103,30,69,142,8,99,37,240,21,10,23,
+   190,
6,148,247,120,234,75,0,26,197,62,94,252,219,203,117,35,11,32,57,177,33,
+   88,237,149,56,87,174,20,125,136,171,168,
68,175,74,165,71,134,139,48,27,166,
+
77,146,158,231,83,111,229,122,60,211,133,230,220,105,92,41,55,46,245,40,244,
+   102,143,54, 65,25,63,161,
1,216,80,73,209,76,132,187,208, 89,18,169,200,196,
+   135,130,116,188,159,86,164,100,109,198,173,186,
3,64,52,217,226,250,124,123,
+
5,202,38,147,118,126,255,82,85,212,207,206,59,227,47,16,58,17,182,189,28,42,
+   223,183,170,213,119,248,152, 2,44,154,163,
70,221,153,101,155,167, 43,172,9,
+   129,22,39,253, 19,98,108,110,79,113,224,232,178,185,
112,104,218,246,97,228,
+   251,34,242,193,238,210,144,12,191,179,162,241,
81,51,145,235,249,14,239,107,
+   49,192,214, 31,181,199,106,157,184,
84,204,176,115,121,50,45,127, 4,150,254,
```

```
+ 138,236,205,93,222,114,67,29,24,72,243,141,128,195,78,66,215,61,156,180);
+
+int perm(int n) {
+    return n >= N_PERM_HALF ? permutation[N_PERM - n] : permutation[n];
+}
+
+float grad(int hash, vec3 point) {
+    // Convert lowest 4 bits of hash code into 12 gradient directions.
+    int h = hash & 15;
+    float u = h < 8 ? point.x : point.y;
+    float v = h < 4 ? point.y : (h==12 || h==14 ? point.x : point.z);
+    return ((h&1) == 0 ? u : -u) + ((h&2) == 0 ? v : -v);
+}
+
+#define N_CUBES 255
+
+float perlin_noise_3d(vec3 p)
+{
+    // Find unit cube that contains point.
+    vec3 unit_cube = floor(p);
+    int xi = int(unit_cube.x) & N_CUBES;
+    int yi = int(unit_cube.y) & N_CUBES;
+    int zi = int(unit_cube.z) & N_CUBES;
+
+    // Find relative x, y, z of point in cube.
+    p -= unit_cube;
+
+    // Compute fade curves for each of x, y, z.
+    float u = blending_weight_poly(p.x);
+    float v = blending_weight_poly(p.y);
+    float w = blending_weight_poly(p.z);
+
+    // Hash coordinates of the 8 cube corners.
+    int a  = perm(xi)   + yi;
+    int aa = perm(a)    + zi;
+    int ab = perm(a +1) + zi;
+    int b  = perm(xi+1) + yi;
+    int ba = perm(b)    + zi;
+    int bb = perm(b +1) + zi;
+
+    // And add blended results from 8 orners of cube.
+    return mix(mix(mix(grad(perm(aa  ), p            ),
+                       grad(perm(ba  ), p-vec3(1,0,0)),
+                       u),
+                   mix(grad(perm(ab  ), p-vec3(0,1,0)),
+                       grad(perm(bb  ), p-vec3(1,1,0)),
+                       u),
+                   v),
+               mix(mix(grad(perm(aa+1), p-vec3(0,0,1)),
+                       grad(perm(ba+1), p-vec3(1,0,1)),
+                       u),
+                   mix(grad(perm(ab+1), p-vec3(0,1,1)),
+                       grad(perm(bb+1), p-vec3(1,1,1)),
+                       u),
+                   v),
+               w);
+}
+
```

```
289 +
290 +float perlin_fbm_3d(vec3 point) {
291 +    float fbm = 0.0f;
292 +    float am  = 1.0f;
293 +    float fm  = 1.0f;
294 +    for (int i = 0; i < num_octaves; ++i) {
295 +        fbm += am * perlin_noise_3d(point * fm);
296 +        am  *= ampl_multiplier;
297 +        fm  *= freq_multiplier;
298 +    }
299 +       return fbm;
300 +}
301 +
302 +float perlin_water_3d(vec3 point) {
303 +    float water = 0.0f;
304 +    float am  = 0.03f;
305 +    float fm  = 0.5f;
306 +    for (int i = 0; i < num_octaves; ++i) {
307 +        water += am * perlin_noise_3d(point * fm);
308 +        am  *= ampl_multiplier_water;
309 +        fm  *= freq_multiplier_water;
310 +    }
311 +
312 +    while(water>0.05){
313 +        water = 0.25*water;
314 +    }
315 +       return water;
316 +}
```

0 comments on commit `2009802`