


[robinmamie](#) / [2019\\_cs341](#) Private

## Add visualization of Perlin 3D to assignment 9





[Browse files](#)

master

 **robinmamie** committed 13 days ago1 parent [eb5eea1](#) commit [2009802334c0888fa87bd90b1d59845461e281ec](#)Showing **12 changed files** with **128 additions** and **12 deletions**.

Unified



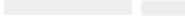

Split

▼ 1  assignment\_9/scripts\_win/run\_fbm.bat   
  
  
[Load diff](#)





This file was deleted.

▼ 1  assignment\_9/scripts\_win/run\_map.bat   
  
  
[Load diff](#)


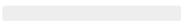
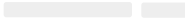

This file was deleted.

▼ 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
```

<div> <div>▼ 9</div> <div>assignment_9/src/shaders/display_fbm.frag</div> <div></div> </div> <pre> 3   in vec2 v2f_tex_coords; 4   out vec4 f_color;      // Final color output produced by                              fragment shader. 5                              // (Can name this                              anything you want...) 6 7   -float <u>perlin_fbm</u>(vec2 point); // Implemented in noise.frag 8 9   void main() { 10  - 11  -   float noise_val = <u>perlin_fbm</u>(v2f_tex_coords) + 0.5; 12 13      f_color = vec4(noise_val, noise_val, noise_val, 1.0); -}_ </pre>	<div> <div>▼ 9</div> <div>assignment_9/src/shaders/display_fbm.frag</div> <div></div> </div> <pre> 3   in vec2 v2f_tex_coords; 4   out vec4 f_color;      // Final color output produced by                              fragment shader. 5                              // (Can name this                              anything you want...) 6   +uniform float time; 7 8   +float <u>perlin_fbm_3d</u>(vec3 point); // Implemented in                              noise.frag 9 10  void main() { 11  + 12  +   float noise_val = <u>perlin_fbm_3d</u>(vec3(v2f_tex_coords, time)) + 0.5; 13      f_color = vec4(noise_val, noise_val, noise_val, 1.0); 14  +} </pre>
<div> <div>▼ 107</div> <div>assignment_9/src/shaders/noise.frag</div> <div></div> </div> <pre> 207      return mix(white, brown_dark, alph); 208  } 209 </pre>	<div> <div>▼ 107</div> <div>assignment_9/src/shaders/noise.frag</div> <div></div> </div> <pre> 207      return mix(white, brown_dark, alph); 208  } 209 210  +// constants for water 211  +const float freq_multiplier_water = 11; 212  +const float ampl_multiplier_water = 0.3; 213  + 214  +// ##### PERLIN 3D ##### 215  +// Source: https://mrl.nyu.edu/~perlin/noise/ 216  +#define N_PERM      512 217  +#define N_PERM_HALF 256 218  + 219  +const int permutation[N_PERM_HALF] = int[N_PERM_HALF] (151,160,137,91,90,15, 220  + 131,13,201,95,96,53,194,233,7,225,140,36,103,30,69,142,8,9 9,37,240,21,10,23, 221  + 190, 6,148,247,120,234,75,0,26,197,62,94,252,219,203,117,35,11, 32,57,177,33, 222  + 88,237,149,56,87,174,20,125,136,171,168, 68,175,74,165,71,134,139,48,27,166, 223  + 77,146,158,231,83,111,229,122,60,211,133,230,220,105,92,41 ,55,46,245,40,244, 224  + 102,143,54, 65,25,63,161, 1,216,80,73,209,76,132,187,208, 89,18,169,200,196, 225  + 135,130,116,188,159,86,164,100,109,198,173,186, 3,64,52,217,226,250,124,123, 226  + 5,202,38,147,118,126,255,82,85,212,207,206,59,227,47,16,58 ,17,182,189,28,42, 227  + 223,183,170,213,119,248,152, 2,44,154,163, 70,221,153,101,155,167, 43,172,9, 228  + 129,22,39,253, 19,98,108,110,79,113,224,232,178,185, 112,104,218,246,97,228, 229  + 251,34,242,193,238,210,144,12,191,179,162,241, 81,51,145,235,249,14,239,107, 230  + 49,192,214, 31,181,199,106,157,184, 84,204,176,115,121,50,45,127, 4,150,254, </pre>

```

231 +
    138,236,205,93,222,114,67,29,24,72,243,141,128,195,78,66,2
    15,61,156,180);
232 +
233 +int perm(int n) {
234 +     return n >= N_PERM_HALF ? permutation[N_PERM - n] :
        permutation[n];
235 +}
236 +
237 +float grad(int hash, vec3 point) {
238 +     // Convert lowest 4 bits of hash code into 12
        gradient directions.
239 +     int h = hash & 15;
240 +     float u = h < 8 ? point.x : point.y;
241 +     float v = h < 4 ? point.y : (h==12 || h==14 ? point.x
        : point.z);
242 +     return ((h&1) == 0 ? u : -u) + ((h&2) == 0 ? v : -v);
243 +}
244 +
245 +#define N_CUBES 255
246 +
247 +float perlin_noise_3d(vec3 p)
248 +{
249 +     // Find unit cube that contains point.
250 +     vec3 unit_cube = floor(p);
251 +     int xi = int(unit_cube.x) & N_CUBES;
252 +     int yi = int(unit_cube.y) & N_CUBES;
253 +     int zi = int(unit_cube.z) & N_CUBES;
254 +
255 +     // Find relative x, y, z of point in cube.
256 +     p -= unit_cube;
257 +
258 +     // Compute fade curves for each of x, y, z.
259 +     float u = blending_weight_poly(p.x);
260 +     float v = blending_weight_poly(p.y);
261 +     float w = blending_weight_poly(p.z);
262 +
263 +     // Hash coordinates of the 8 cube corners.
264 +     int a = perm(xi) + yi;
265 +     int aa = perm(a) + zi;
266 +     int ab = perm(a + 1) + zi;
267 +     int b = perm(xi + 1) + yi;
268 +     int ba = perm(b) + zi;
269 +     int bb = perm(b + 1) + zi;
270 +
271 +     // And add blended results from 8 orners of cube.
272 +     return mix(mix(mix(grad(perm(aa), p
        ),
273 +         grad(perm(ba), p-vec3(1,0,0)),
274 +         u),
275 +         mix(grad(perm(ab), p-vec3(0,1,0)),
276 +         grad(perm(bb), p-vec3(1,1,0)),
277 +         u),
278 +         v),
279 +         mix(mix(grad(perm(aa+1), p-vec3(0,0,1)),
280 +         grad(perm(ba+1), p-vec3(1,0,1)),
281 +         u),
282 +         mix(grad(perm(ab+1), p-vec3(0,1,1)),
283 +         grad(perm(bb+1), p-vec3(1,1,1)),
284 +         u),
285 +         v),
286 +         w);
287 +}
288 +

```

```
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`