Human GPU #0014 – Filling pixels

Ok, so, you liked painting and filling squares, human?

Our previous square in #0013 was 30x30px, and it filled a surface of 900 pixels.

The fragment shader was executed (in your mind) 900 times, and the vertex shader was executed 6 times.

Few more things about fragment shader:

Don't pay too much attention to the `precision highp float;`; Fragment shader lacks of default precision format, so we must specify one.

The RGBA values expressed in gl_FragColor spans from 0 to 1. For instance `rgb(233, 10, 30)` will be expressed as `vec3(233.0 / 255.0, 10.0 / 255.0, 30.0 / 255.0)`

The fragment shader can access to uniforms, like the vertex shader does.

Draw two triangles, and fill them!

Buffers

```
{
  "data": [
    -0.5, -0.5, 2.0, -0.5, 0.5, 2.0,
    0.5, 0.5, 1.0, -0.5, -0.5, 2.0,
    0.5, -0.5, 1.0, 0.5, 0.5, 1.0
  ]
}
```

Attributes

```
{ "aPosition": { "buffer": "data", "size": 3 } }
```

Uniforms

```
{ "uColor": [1.0, 0.0, 0.0] }
```

Vertex shader

```
attribute vec3 aPosition;

void main() {
  gl_Position = vec4(aPosition.xy, 0.0, aPosition.z);
}
```

Fragment shader

```
precision highp float;
uniform vec3 uColor;

void main() {
  gl_FragColor = vec4(uColor.rgb, 1.0);
}
```