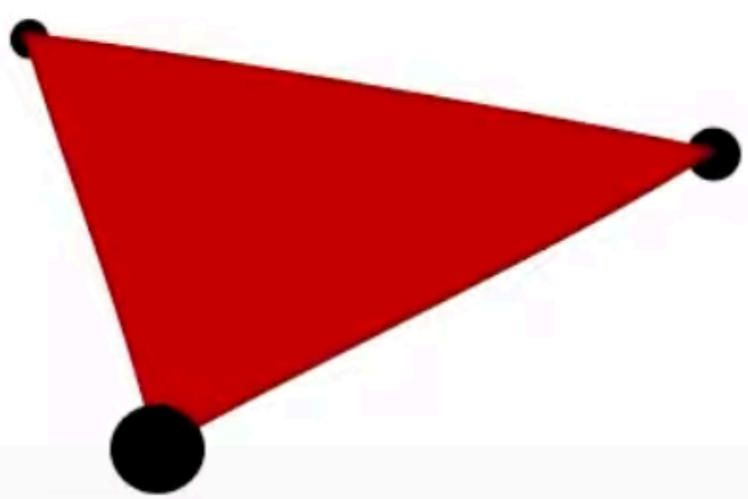
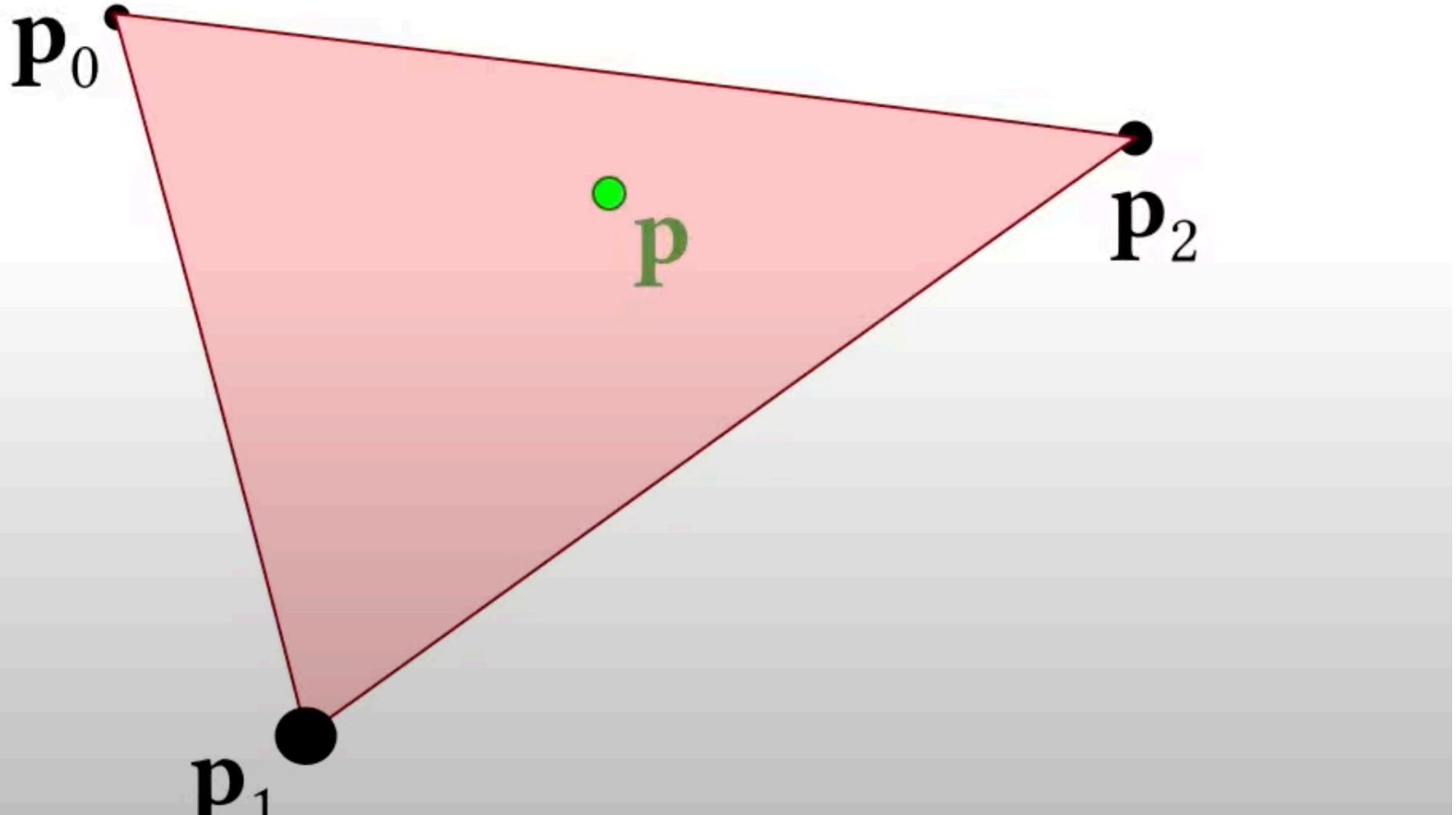
Triangular Meshes

Triangles

Triangles

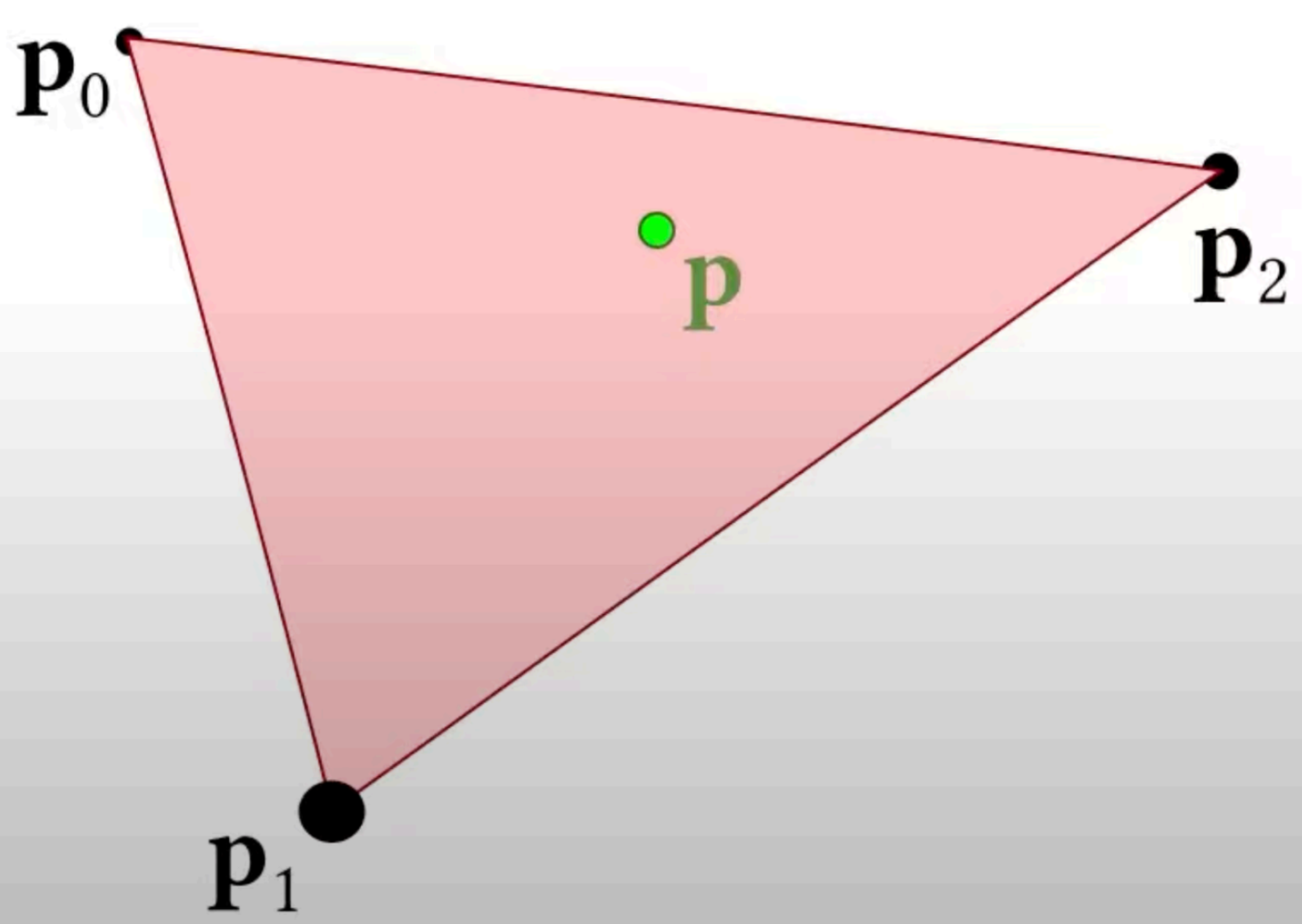


Triangles



Barycentric Coordinates

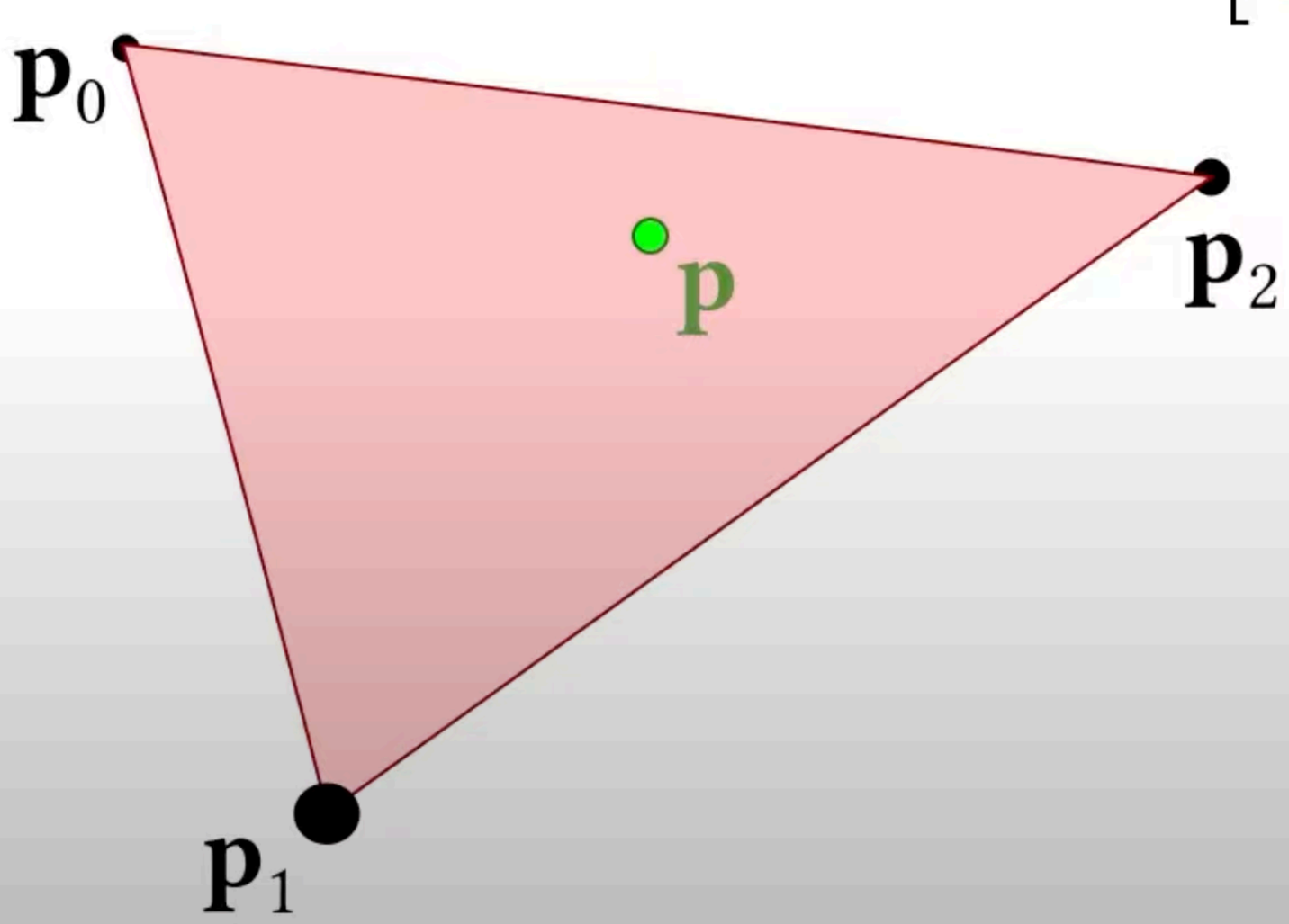
$$\mathbf{p} = \alpha\,\mathbf{p}_0 + \beta\,\mathbf{p}_1 + \gamma\,\mathbf{p}_2$$



Barycentric Coordinates

$$\mathbf{p} = \alpha\,\mathbf{p}_0 + \beta\,\mathbf{p}_1 + \gamma\,\mathbf{p}_2$$

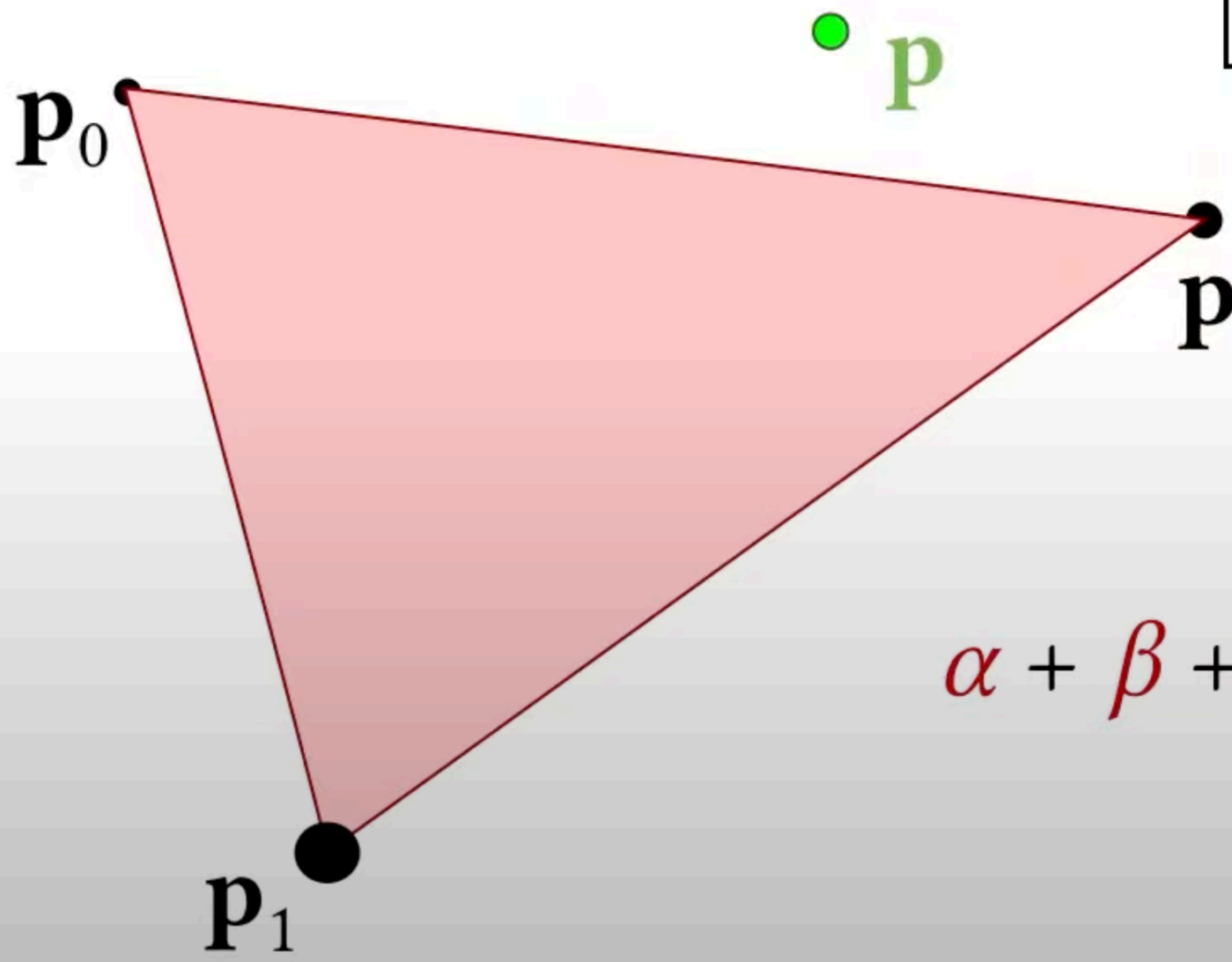
$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$$



Barycentric Coordinates

$$\mathbf{p} = \alpha\,\mathbf{p}_0 + \beta\,\mathbf{p}_1 + \gamma\,\mathbf{p}_2$$

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$$

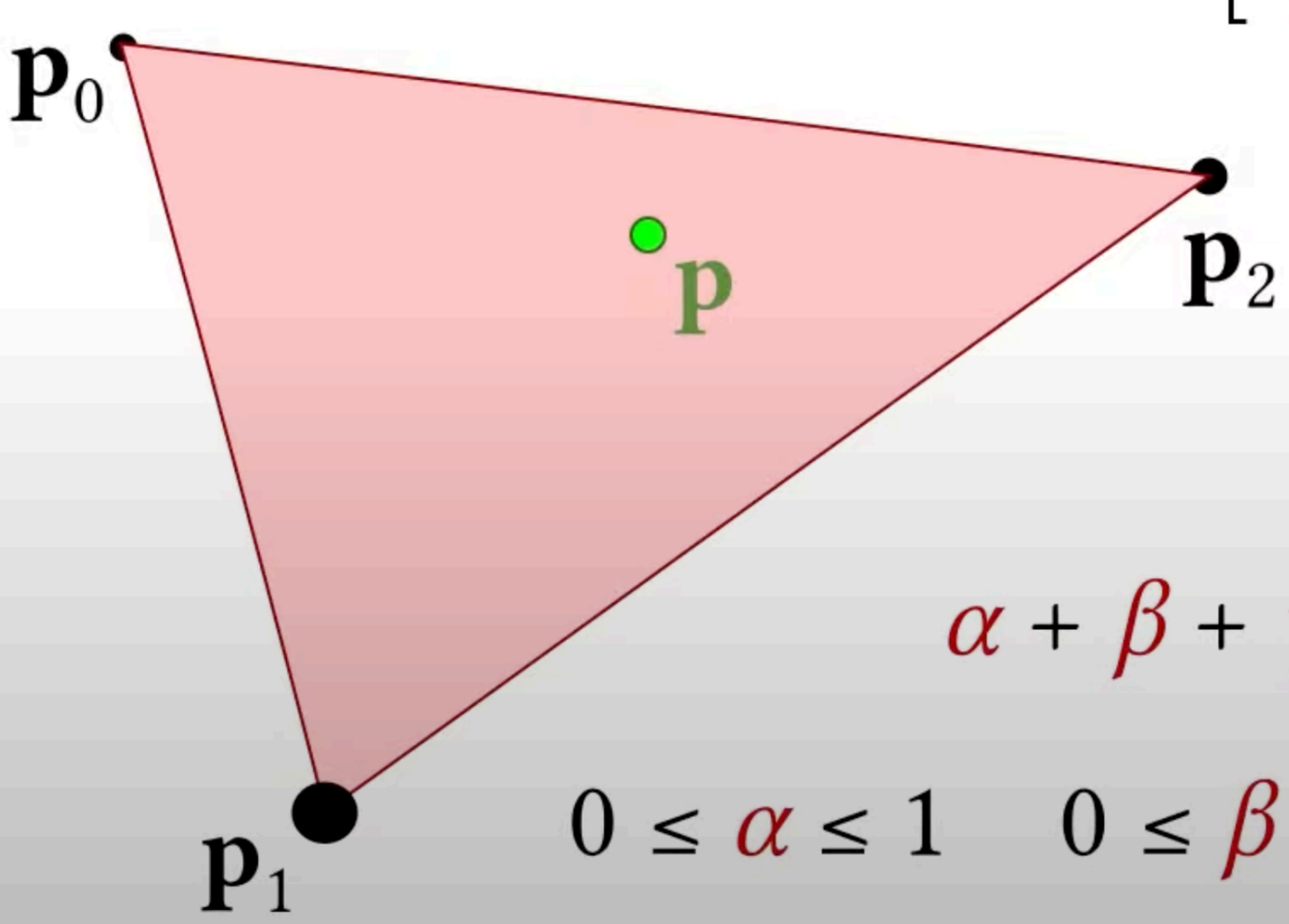


$$\alpha + \beta + \gamma = 1$$

Barycentric Coordinates

$$\mathbf{p} = \alpha\,\mathbf{p}_0 + \beta\,\mathbf{p}_1 + \gamma\,\mathbf{p}_2$$

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$$

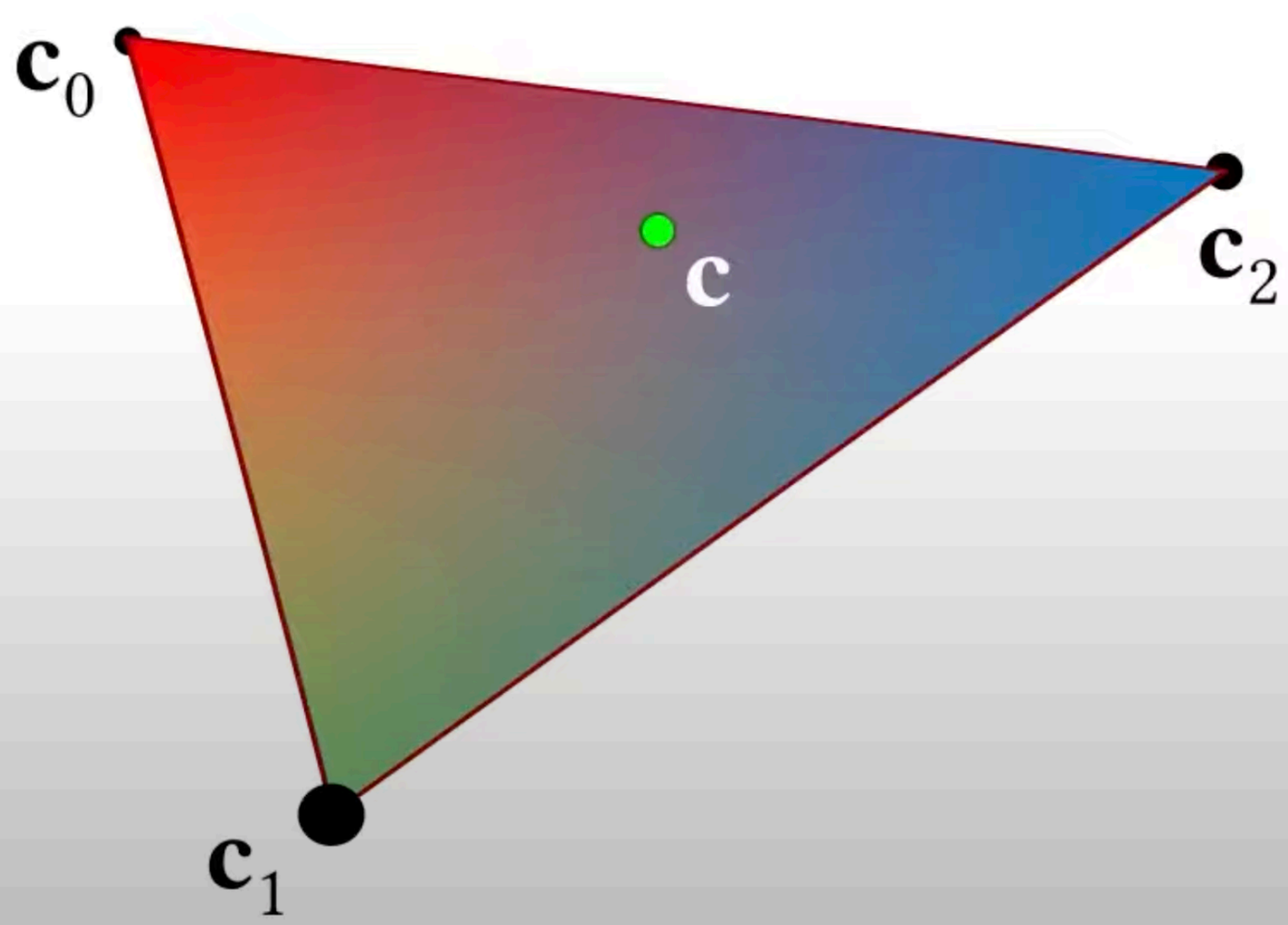


$$\alpha + \beta + \gamma = 1$$

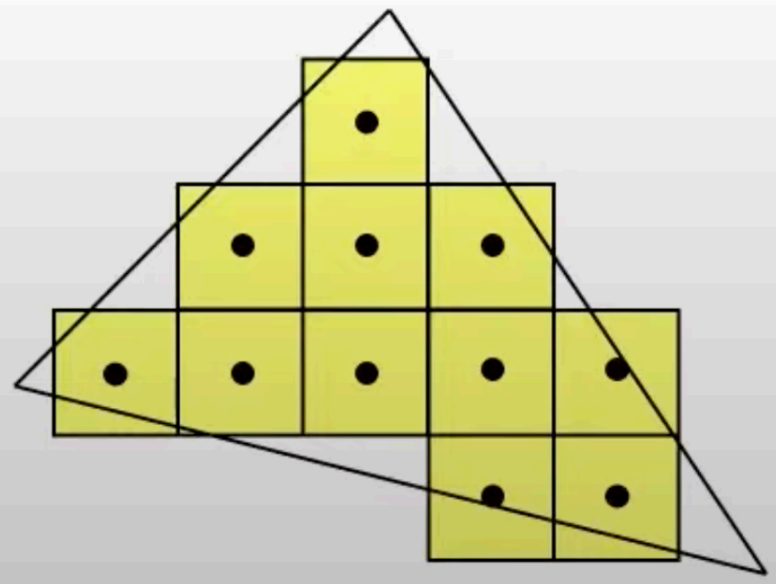
$$0 \le \alpha \le 1 \quad 0 \le \beta \le 1 \quad 0 \le \gamma \le 1$$

Barycentric Coordinates

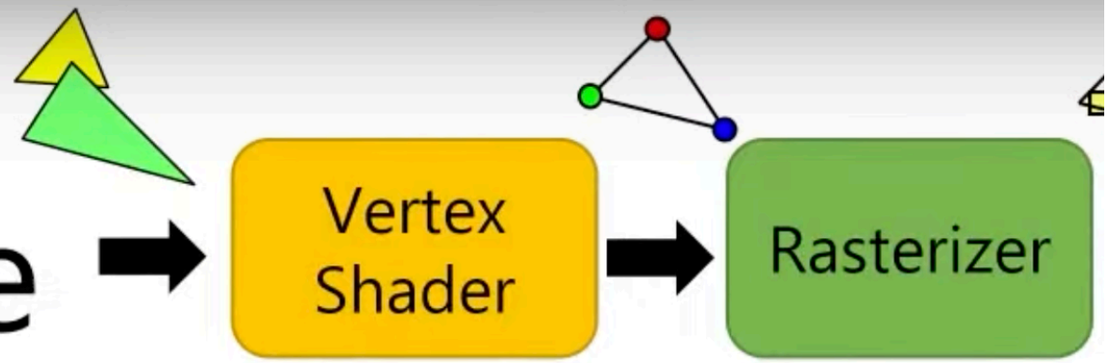
$$\mathbf{c} = \alpha\,\mathbf{c}_0 + \beta\,\mathbf{c}_1 + \gamma\,\mathbf{c}_2$$



GPU Pipeline



GPU Pipeline



Vertex Shader

GLSL

```
#version 330 core
layout(location=0) in vec3 pos;
layout(location=0) in vec4 clr;
uniform mat4 trans;
out vec4 vcolor;
void main()
{
    gl_Position = trans * vec4(pos,1);
    vcolor = clr;
}
```

Fragment Shader

GLSL

```
#version 330 core
layout(location=0) out vec4 color;
in vec4 vcolor;
void main()
{
    color = vcolor;
}
```

Triangular Meshes

- List of vertices
  - x, y, z position per vertex

- List of triangles (faces)
  - Vertex indices

OpenGL

- Vertex Attributes

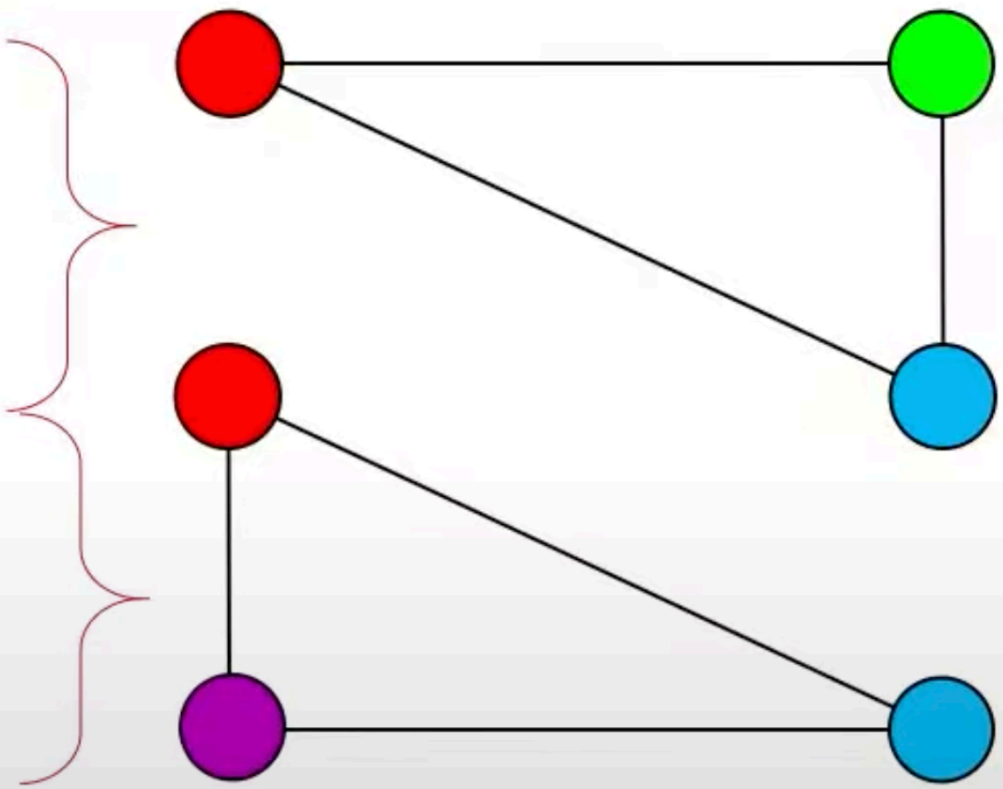
	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z
4	x	y	z
5	x	y	z

	colors			
0	r	g	b	a
1	r	g	b	a
2	r	g	b	a
3	r	g	b	a
4	r	g	b	a
5	r	g	b	a

OpenGL

- Vertex Attributes

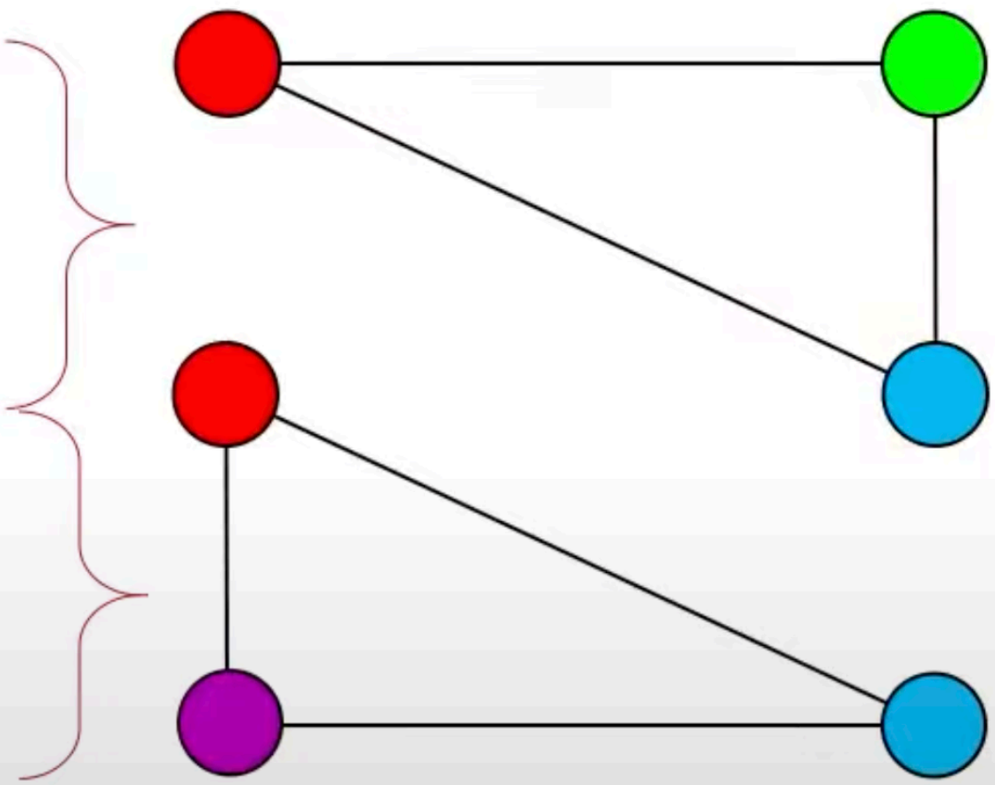
	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z
4	x	y	z
5	x	y	z



OpenGL

- Vertex Attributes

	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z
4	x	y	z
5	x	y	z



```
glDrawArrays( GL_TRIANGLES, 0, 6 );
```

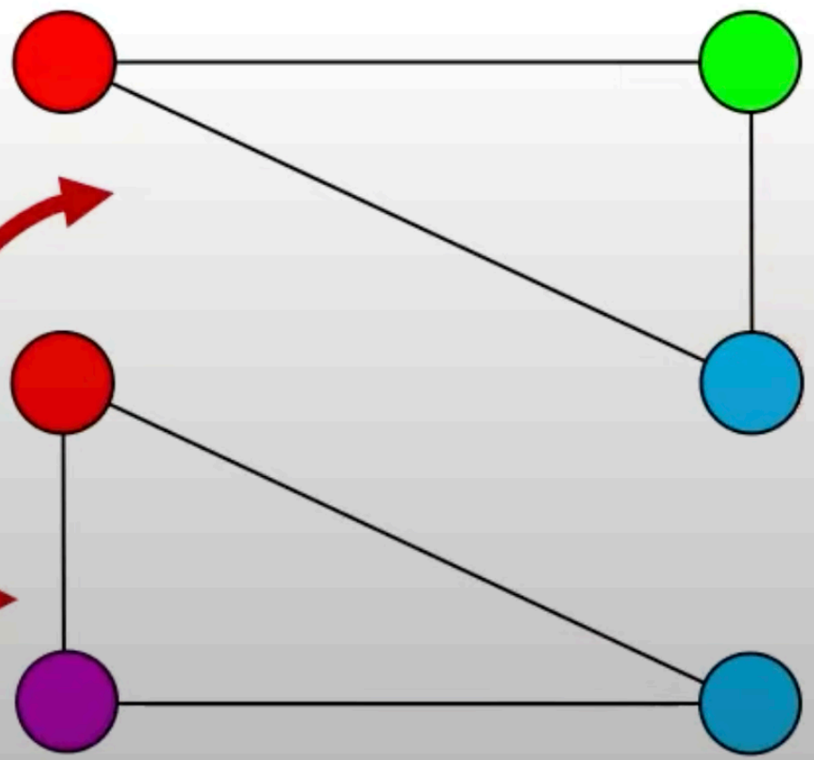
OpenGL

- Vertex Attributes

	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z

- Elements

0	0	1	2
1	0	2	3



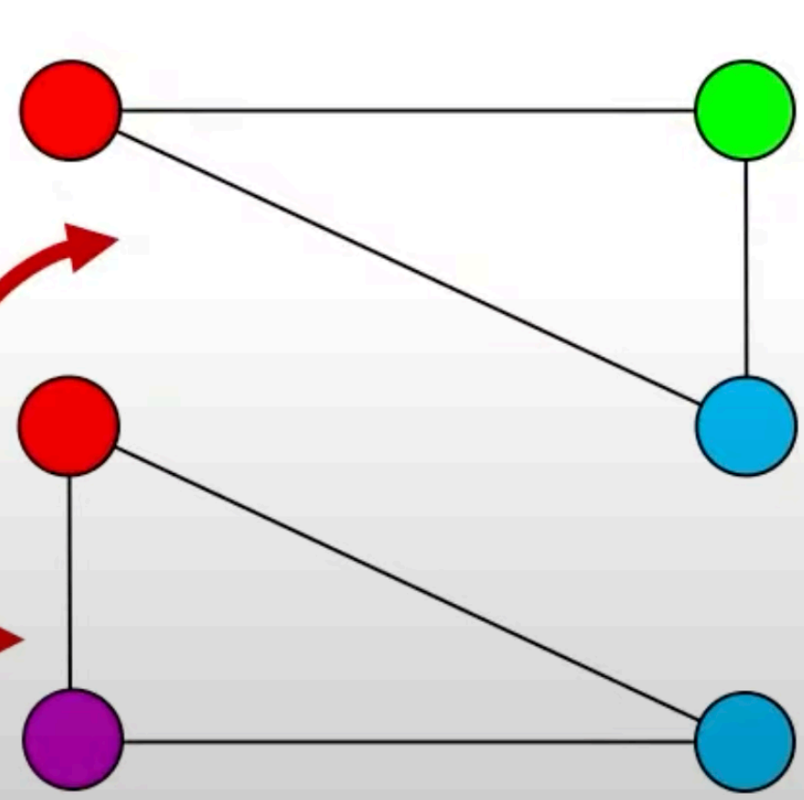
OpenGL

- Vertex Attributes

	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z

- Elements

0	0	1	2
1	0	2	3



```
glDrawArrays( GL_TRIANGLES, 0, 6 );
```

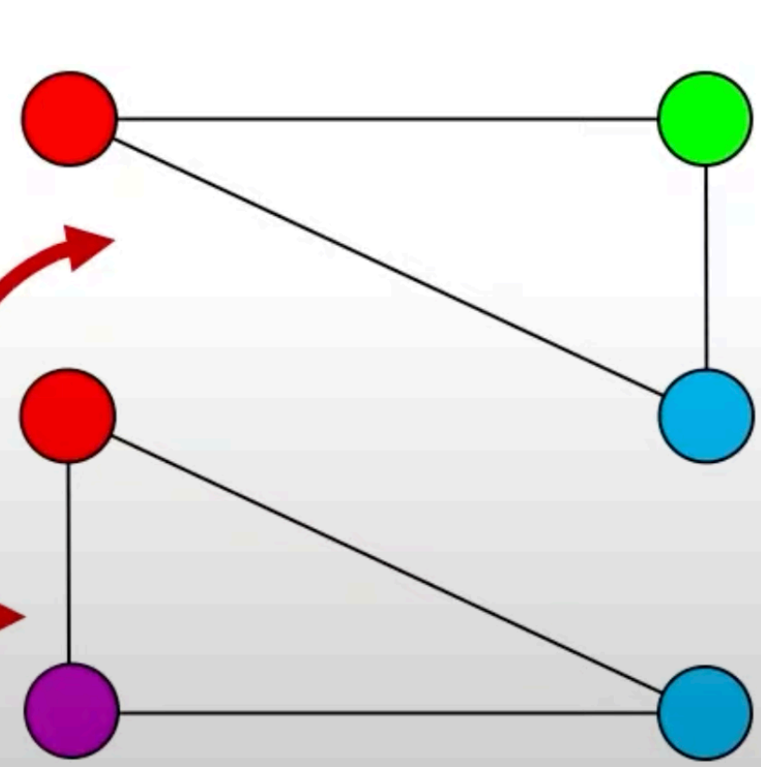
OpenGL

- Vertex Attributes

	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z

- Elements

0	0	1	2
1	0	2	3



```
glDrawElements( GL_TRIANGLES, 6, GL_UNSIGNED_INT, 0
```

Element Buffer Object

```
Unsigned int elem[] = {
                0, 1, 2,
                0, 2, 3
};
```

Element Buffer Object

```
GLuint ebuffer;
glGenBuffers( 1, & ebuffer );
glBindBuffer( GL_ELEMENT_ARRAY_BUFFER, ebuffer )
glBufferData( GL_ELEMENT_ARRAY_BUFFER,
              sizeof(unsigned int)*6,
              elem,
              GL_STATIC_DRAW );

...
glBindBuffer( GL_ELEMENT_ARRAY_BUFFER, ebuffer )
glDrawElements( GL_TRIANGLES, 6, GL_UNSIGNED_INT
                                                0);
```

OpenGL

- Vertex Attributes

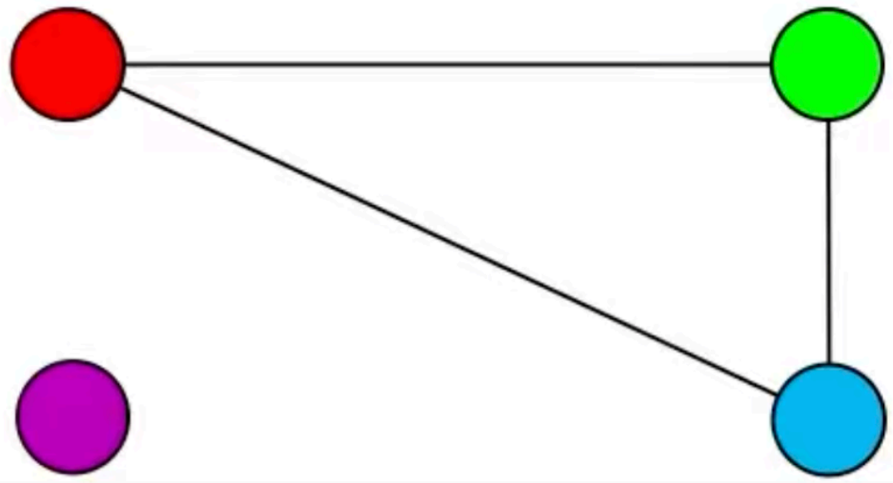
	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z

	colors			
0	r	g	b	a
1	r	g	b	a
2	r	g	b	a
3	r	g	b	a

OpenGL

- Vertex Attributes

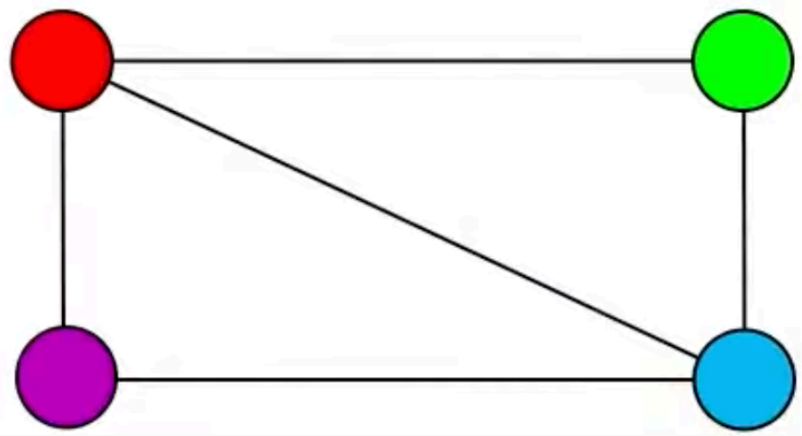
	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z



OpenGL

- Vertex Attributes

	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z

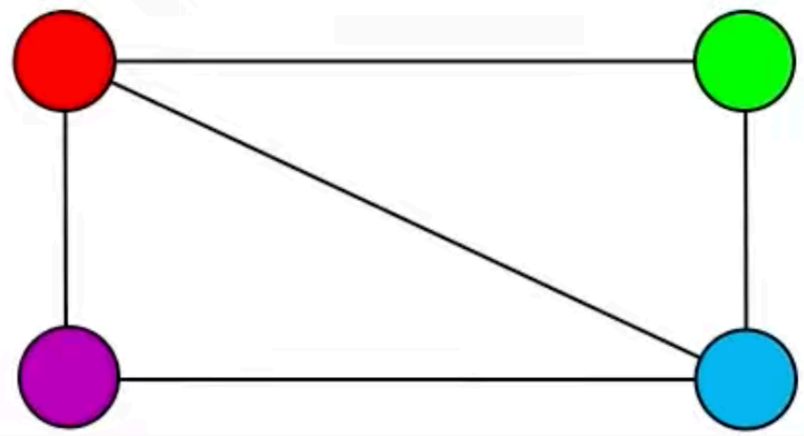


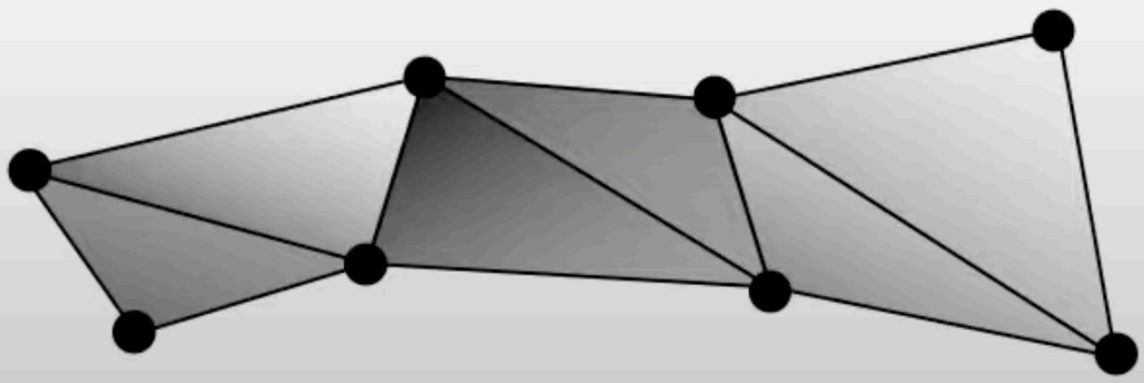
```
glDrawArrays( GL_TRIANGLE_STRIP, 0, 4 );
```

OpenGL

- Vertex Attributes

	positions		
0	x	y	z
1	x	y	z
2	x	y	z
3	x	y	z



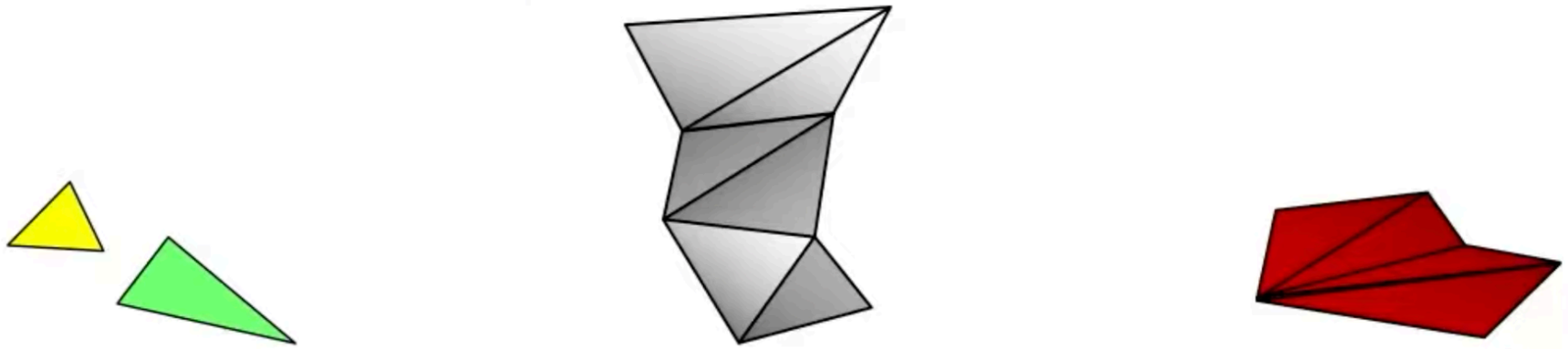


```
glDrawArrays( GL_TRIANGLE_STRIP, 0, 4 );
```

Triangle Strips

OpenGL

- Drawing Triangles



GL_TRIANGLES GL_TRIANGLE_STRIP GL_TRIANGLE_FAN

```
glDrawArrays( GL_TRIANGLES, 0, 6 );
glDrawArrays( GL_TRIANGLE_STRIP, 0, 4 );
glDrawArrays( GL_TRIANGLE_FAN, 0, 4 );
```

glDrawElements(GL_TRIANGLES, 6, GL_UNSIGNED_INT, 0);
glDrawElements(GL_TRIANGLE_STRIP, 4, GL_UNSIGNED_INT, 0);
glDrawElements(GL_TRIANGLE_FAN, 4, GL_UNSIGNED_INT, 0);