

Asn5: 5c

Shadertoy.com

Favorite Example: "Folding Space"

Why: This example creates mesmerizing visuals by simulating folding space. It implements advanced shader techniques to create the illusion of depth and movement within a 2D space.

Source Code: The source code is available directly on the ShaderToy website, allowing for a deep dive into the shader code. I understand the general structure and some of the techniques used, but some parts, such as the specific math functions used for the folding effect, would require more understanding.

Other Topics: I'd love to delve deeper into shader mathematics, especially techniques for creating complex visual effects like folding space. Additionally, understanding optimizations for performance in shader code would be valuable.

threejs.org/examples/

Favorite Example: "WebGL - Materials - Cubemap"

Why: This example demonstrates the use of cubemaps for realistic reflections in a 3D scene. It's impressive because it showcases how shaders can be used to achieve photorealistic effects.

Source Code: Three.js provides source code for its examples directly on the website, making it accessible for learning. I understand the structure of the code and how Three.js integrates with WebGL, but I would need more understanding of the specifics of cubemap implementation and shader techniques for reflections.

Other Topics: I'd be interested in learning more about advanced lighting techniques, such as global illumination and physically-based rendering, as well as how to optimize performance in complex 3D scenes.

Fun with WebGL 2.0

Favorite Example: "Menger Sponge"

Why: This example showcases the beauty of fractals in a real-time interactive environment. It's impressive because it renders a complex 3D fractal using WebGL, demonstrating the power of GPU computing.

Source Code: The source code for this example is provided directly on the website. I understand the overall structure of the code, but I would need more understanding of the mathematical concepts behind fractals and how they're implemented in shaders.

Other Topics: Exploring more about fractals and procedural generation would be fascinating, as well as learning about techniques for optimizing performance when rendering complex geometry in WebGL.

Shader Showdown Example

Favorite Example: "Asteroid Field"

Why: This example uses procedural generation and clever shader techniques to simulate an asteroid field in real-time. It's impressive because it creates a dynamic and visually stunning scene using minimal code.

Source Code: The source code is available directly on the Shader Showdown website. I understand the basic structure of the code, but I would need more understanding of the specific techniques used for procedural generation and optimizing performance in shader code.

Other Topics: Learning more about procedural content generation and creative coding techniques for real-time graphics would be valuable, as well as understanding how to integrate audio and other interactive elements into shader-based visuals.