

PROCEDURAL TERRAIN GENERATION

Pranav Nair – CS334 Final Project

OVERVIEW

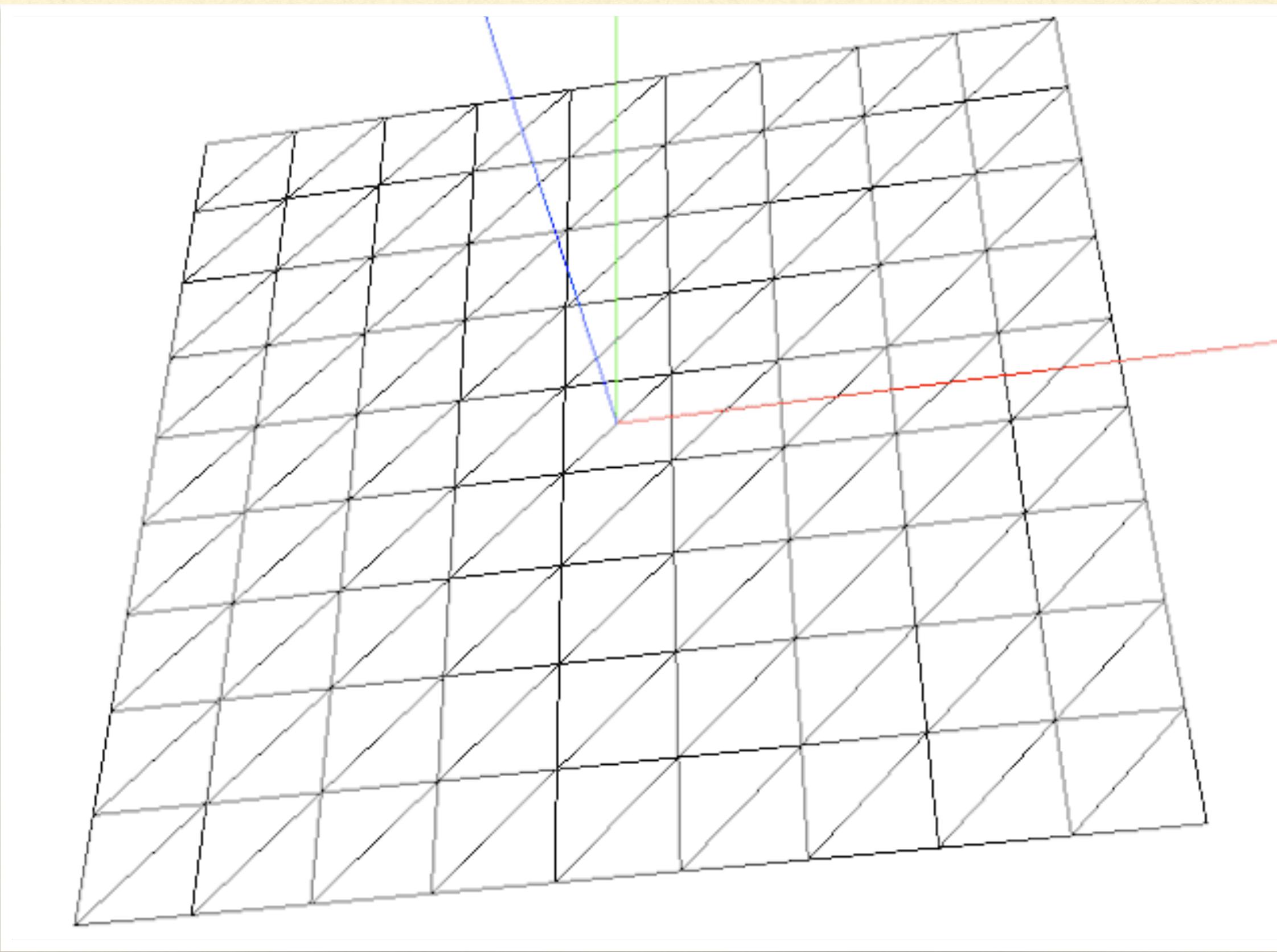
- Project components

- Mesh creation
- Core procedural generation
- Water bodies
- Grammar

- Demos

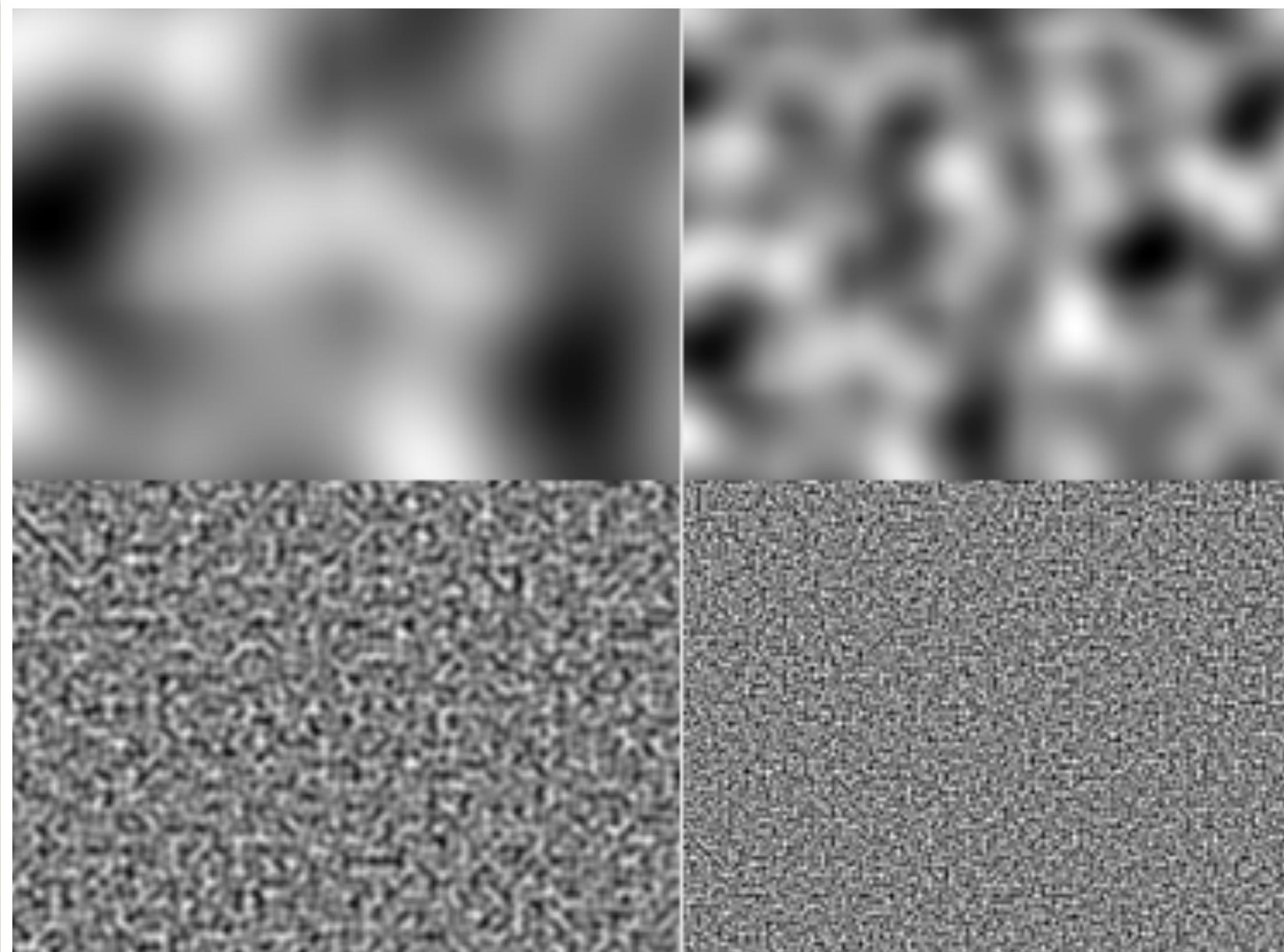
- Example terrains
- Live demo

MESH CREATION

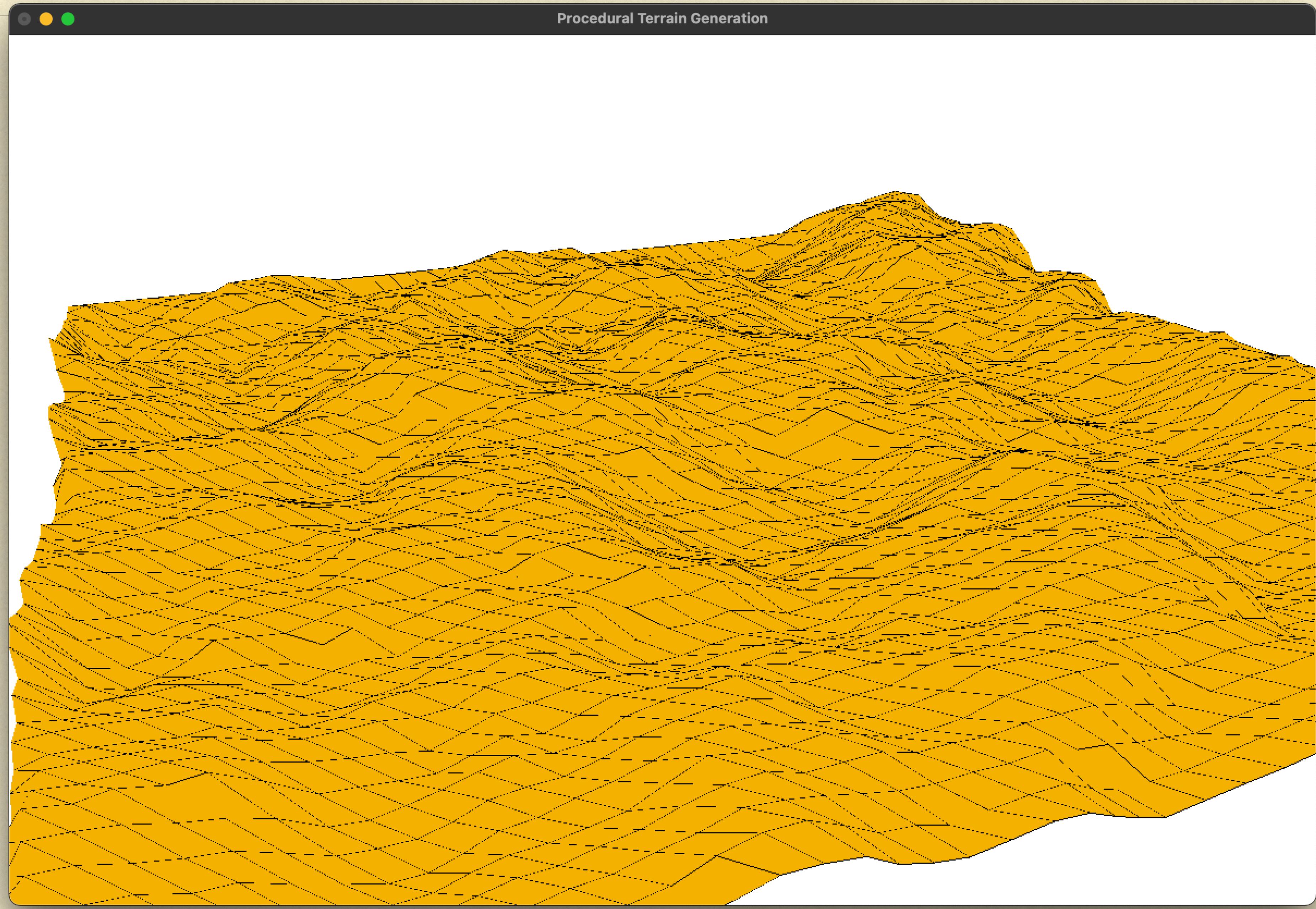


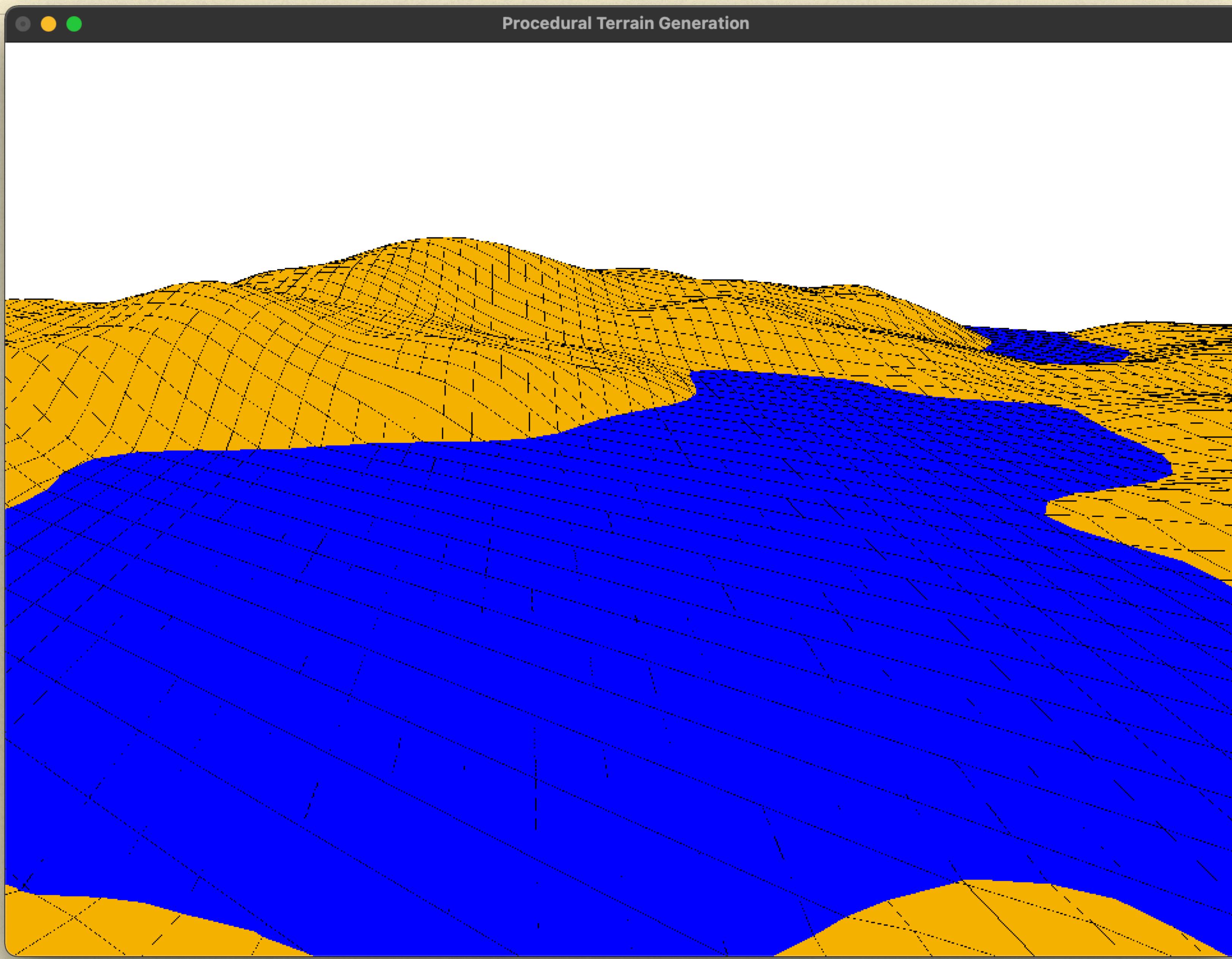
- Algorithm developed to create a $N \times N$ plane mesh using OpenGL primitives

CORE PROCEDURAL GENERATION



- 2D Perlin noise used to create height map
- Variable frequency and octaves to customize terrain
 - Corresponds to ‘mountain density’ and ‘terrain roughness’, respectively
- Reputeless’ C++ library used (<https://github.com/Reputeless/PerlinNoise>)





GRAMMAR

```
● ● ● Untitled — Edited

# Dimensions x z
50 50

# Terrain characteristics
# height, roughness, mountain density, water bodies, water level
15 5 3.0 1 0.0

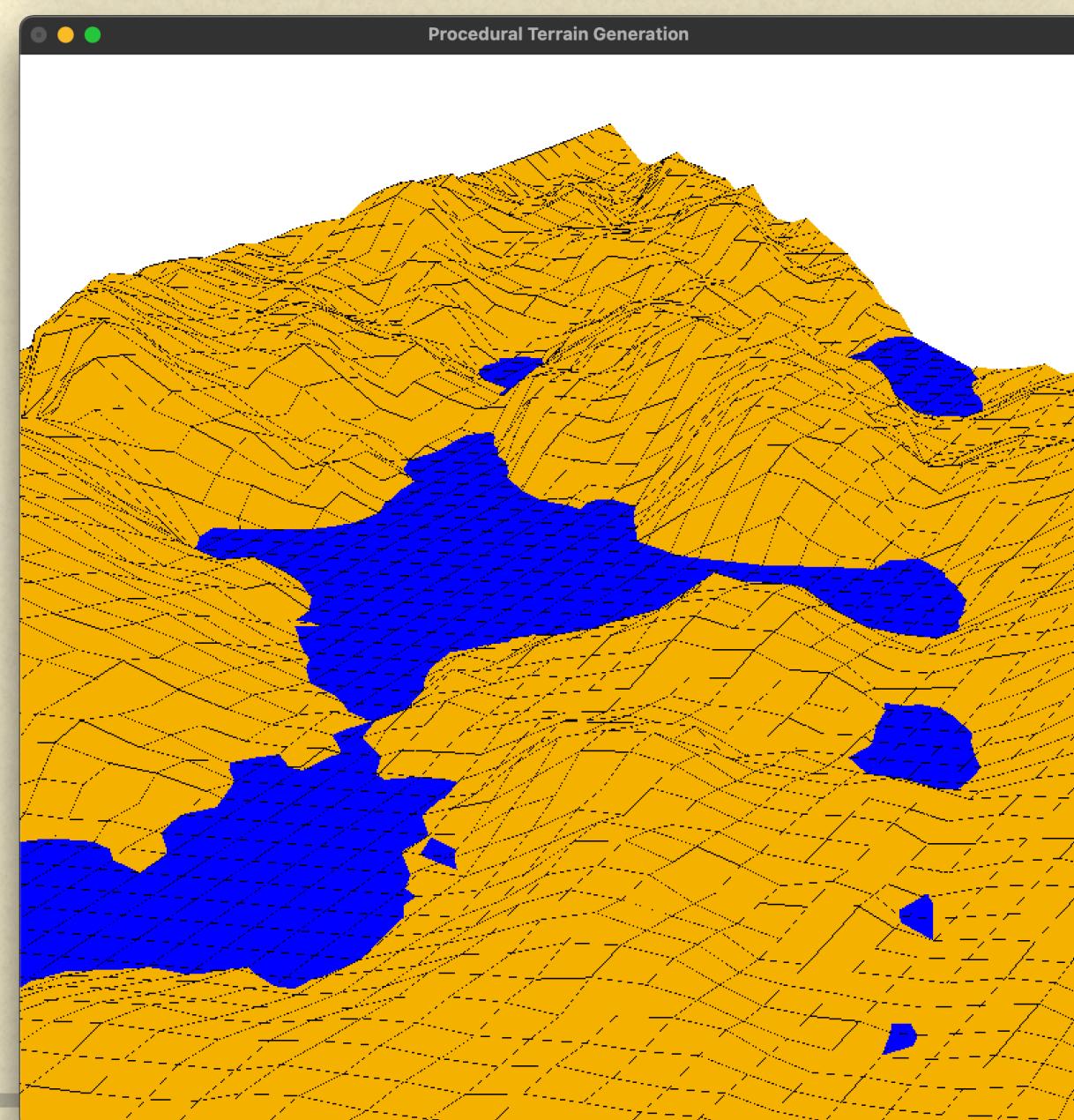
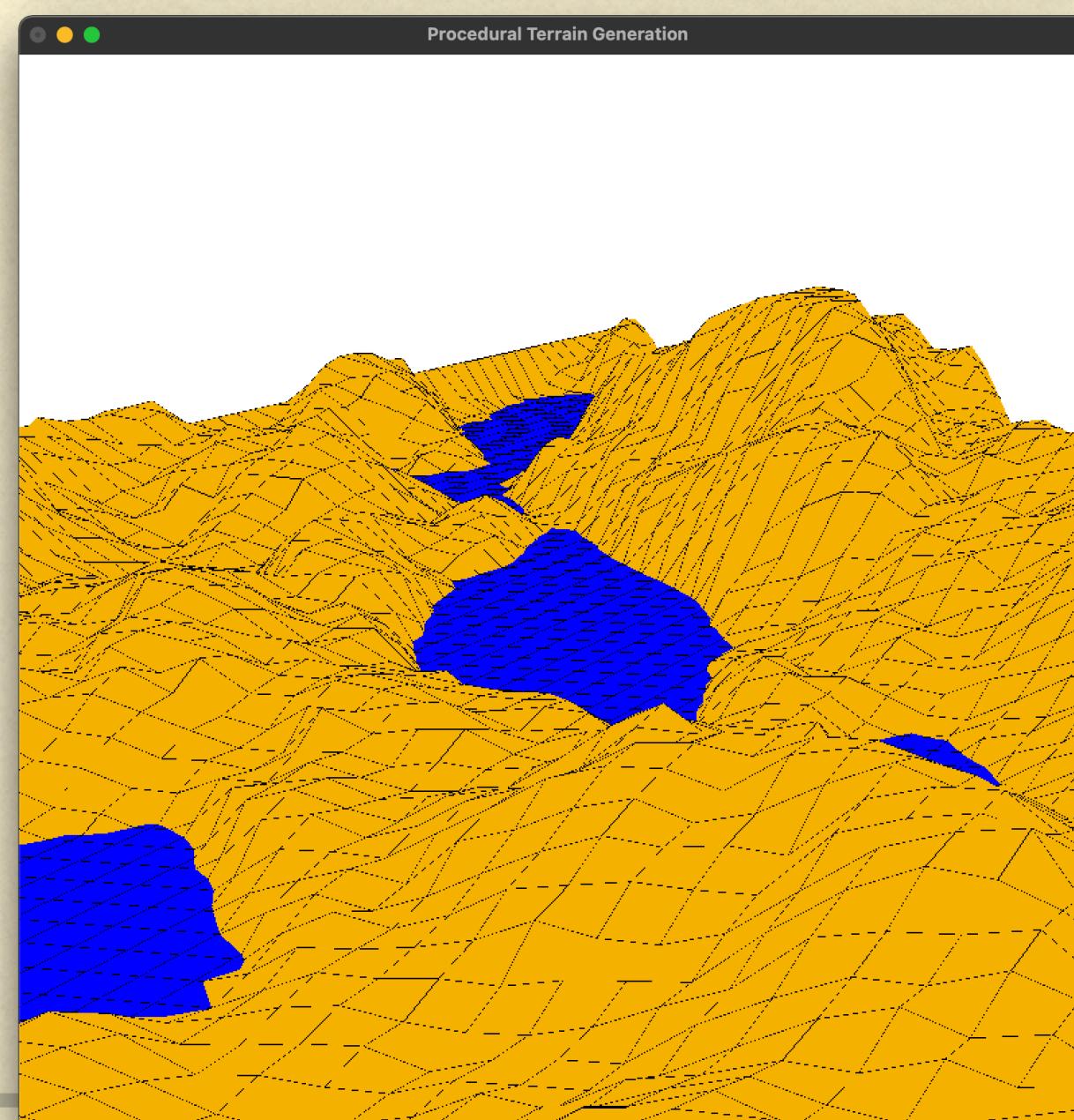
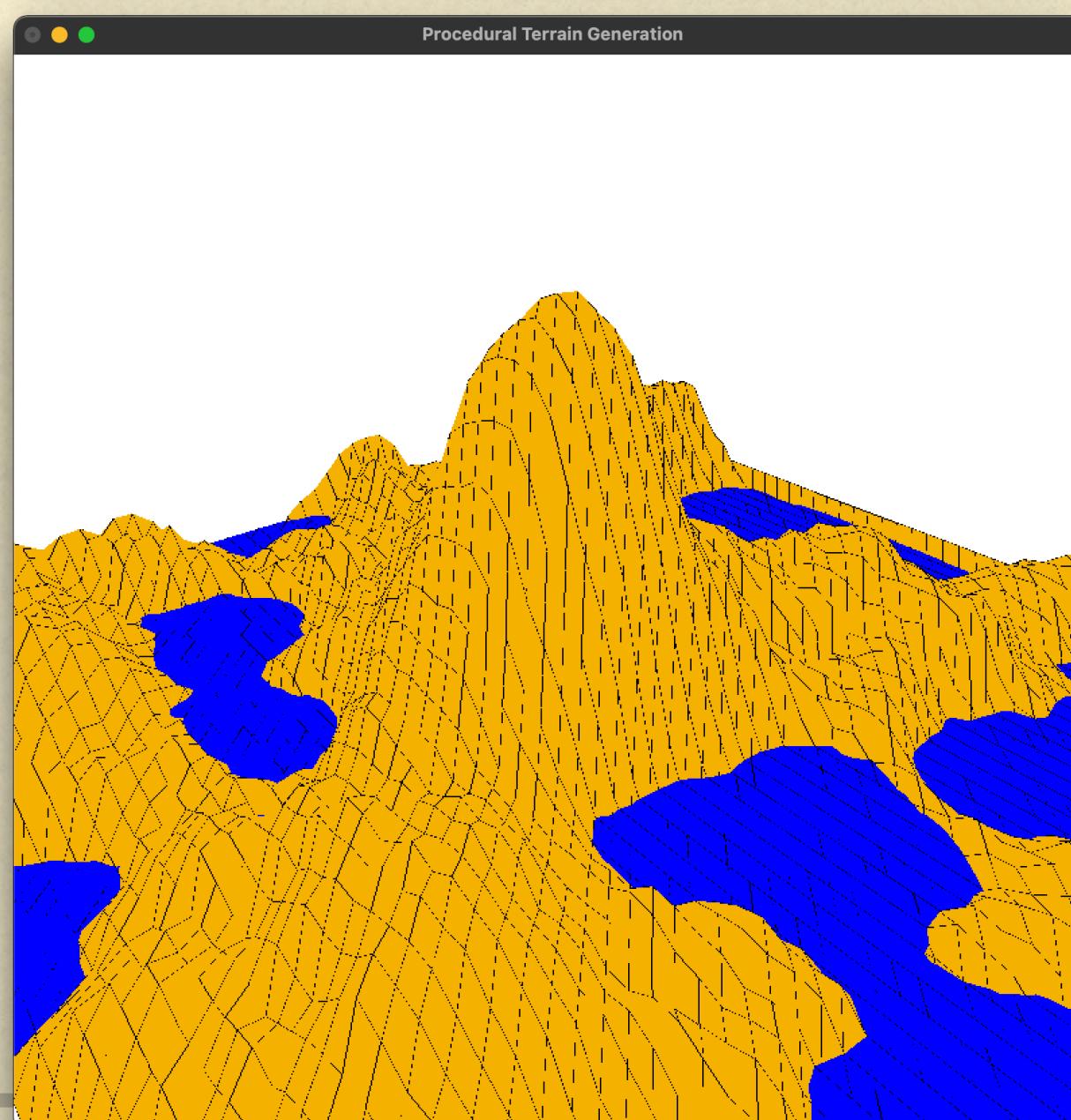
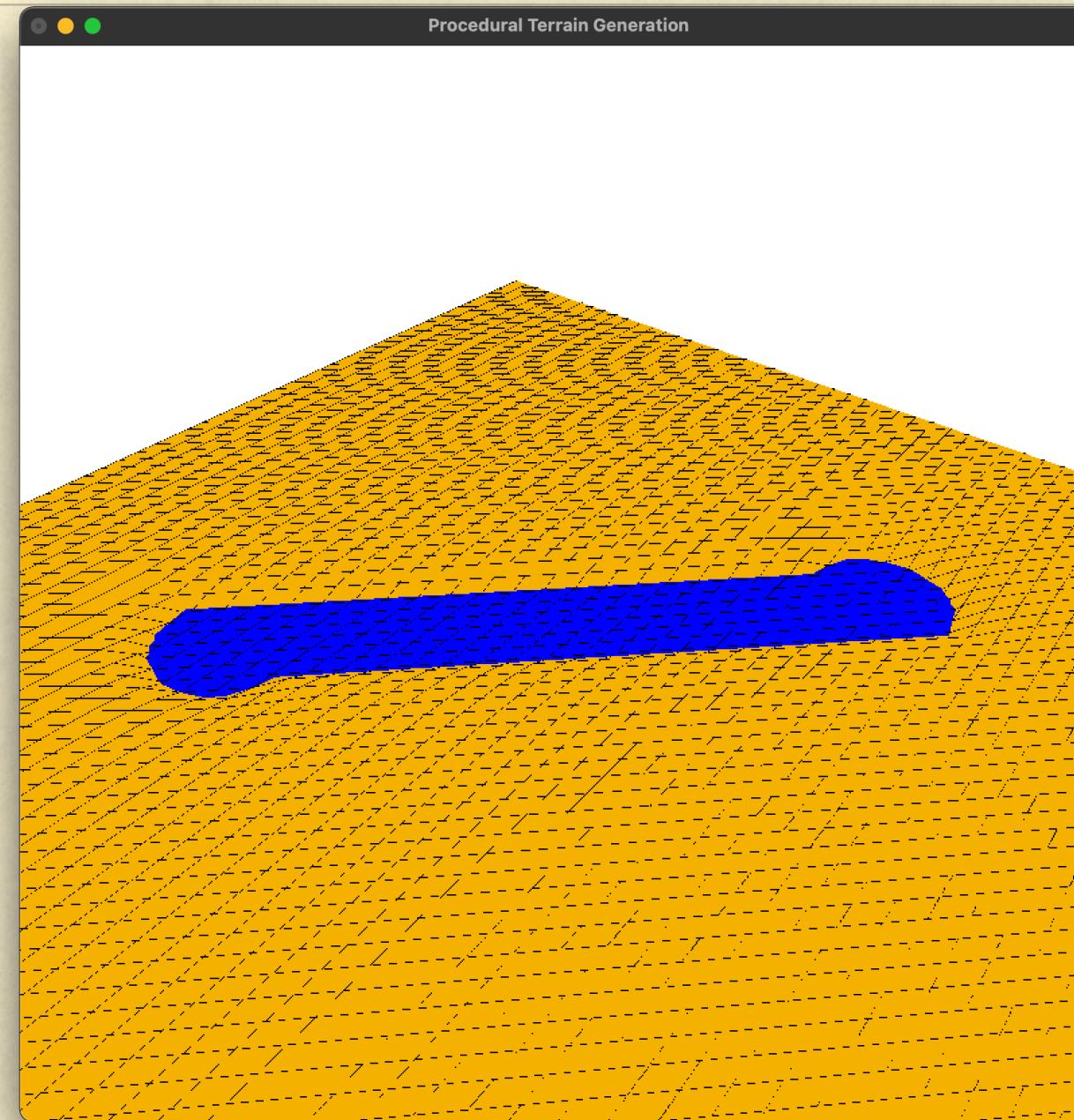
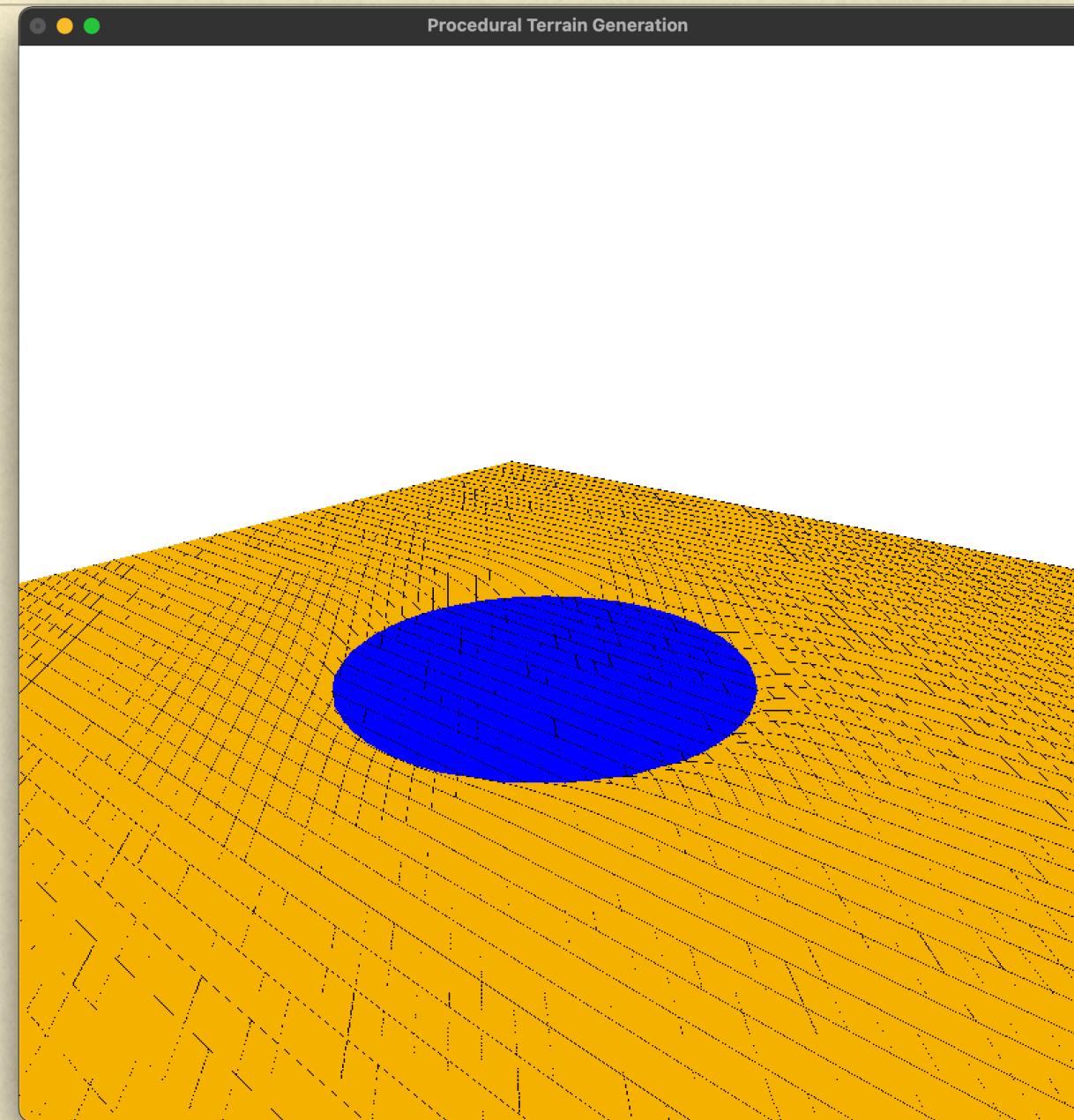
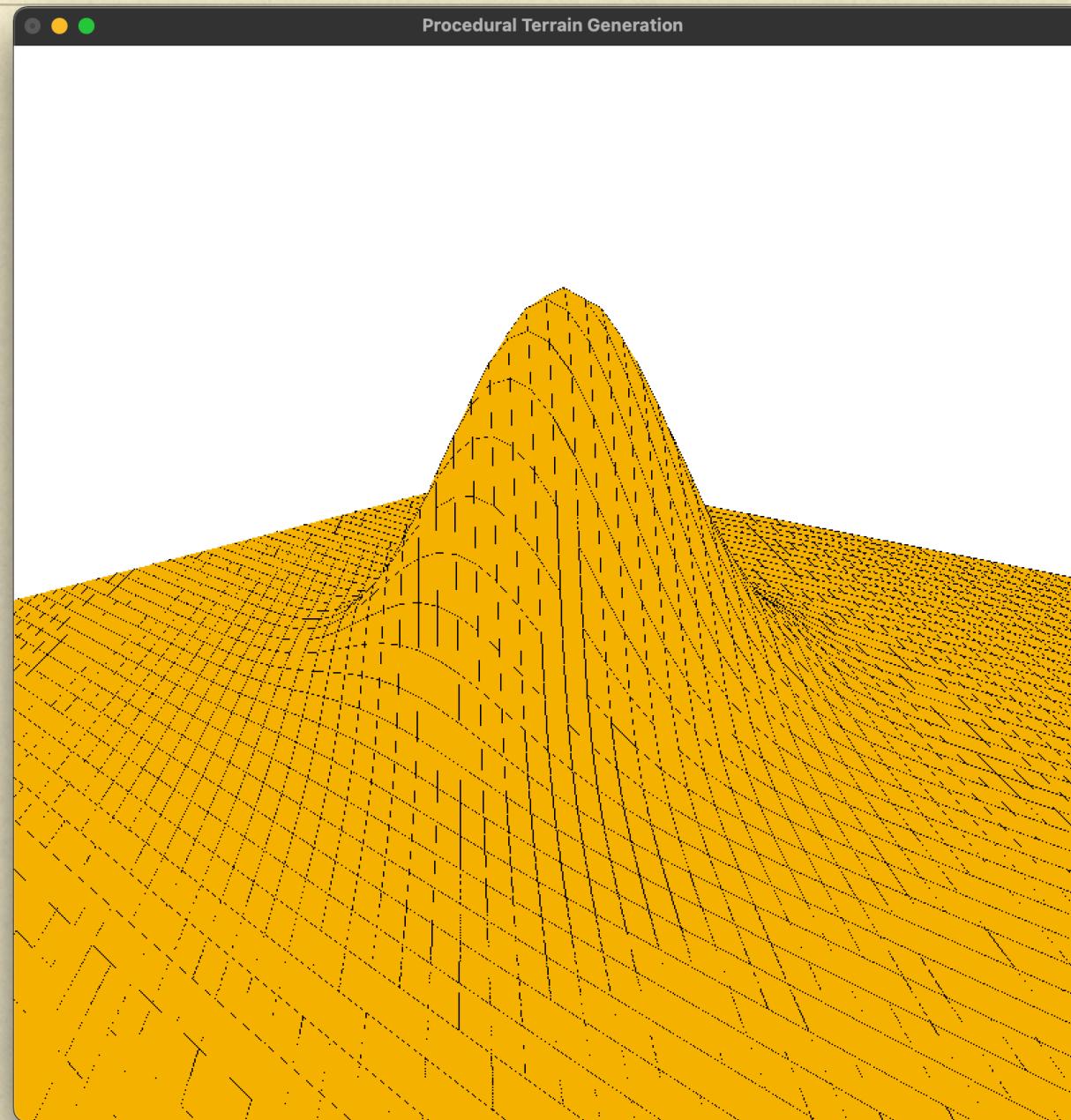
# Encourage mountain
# x z height width
M 34 34 15 40

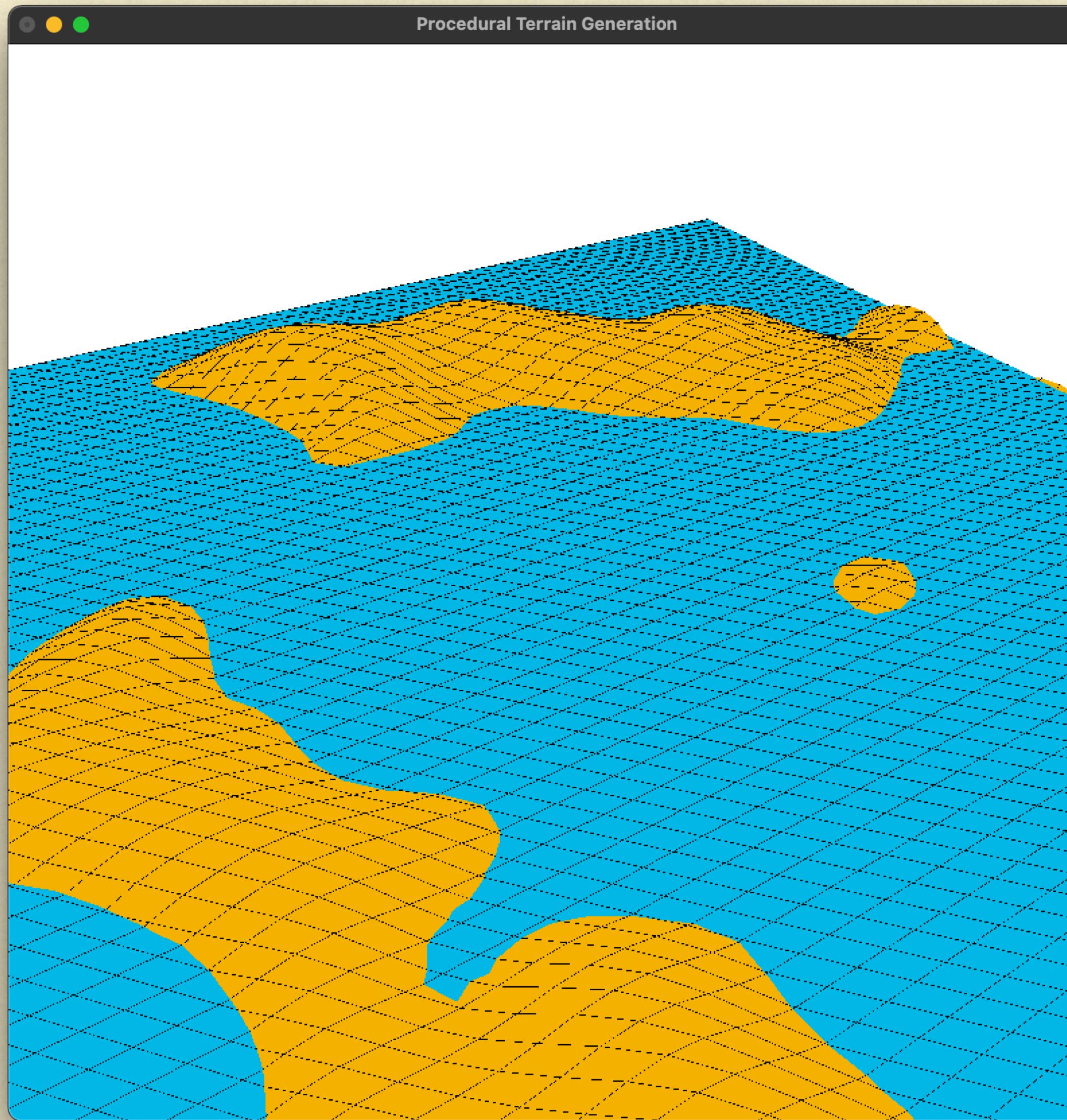
# Encourage lake
# x z width
L 0 0 20

# Encourage river/stream
# p0, p1
R 0 0 34 34

# Custom Color
# terrain rgb, water rgb
C 0.92 0.71 0.20 0.26 0.51 0.64
```

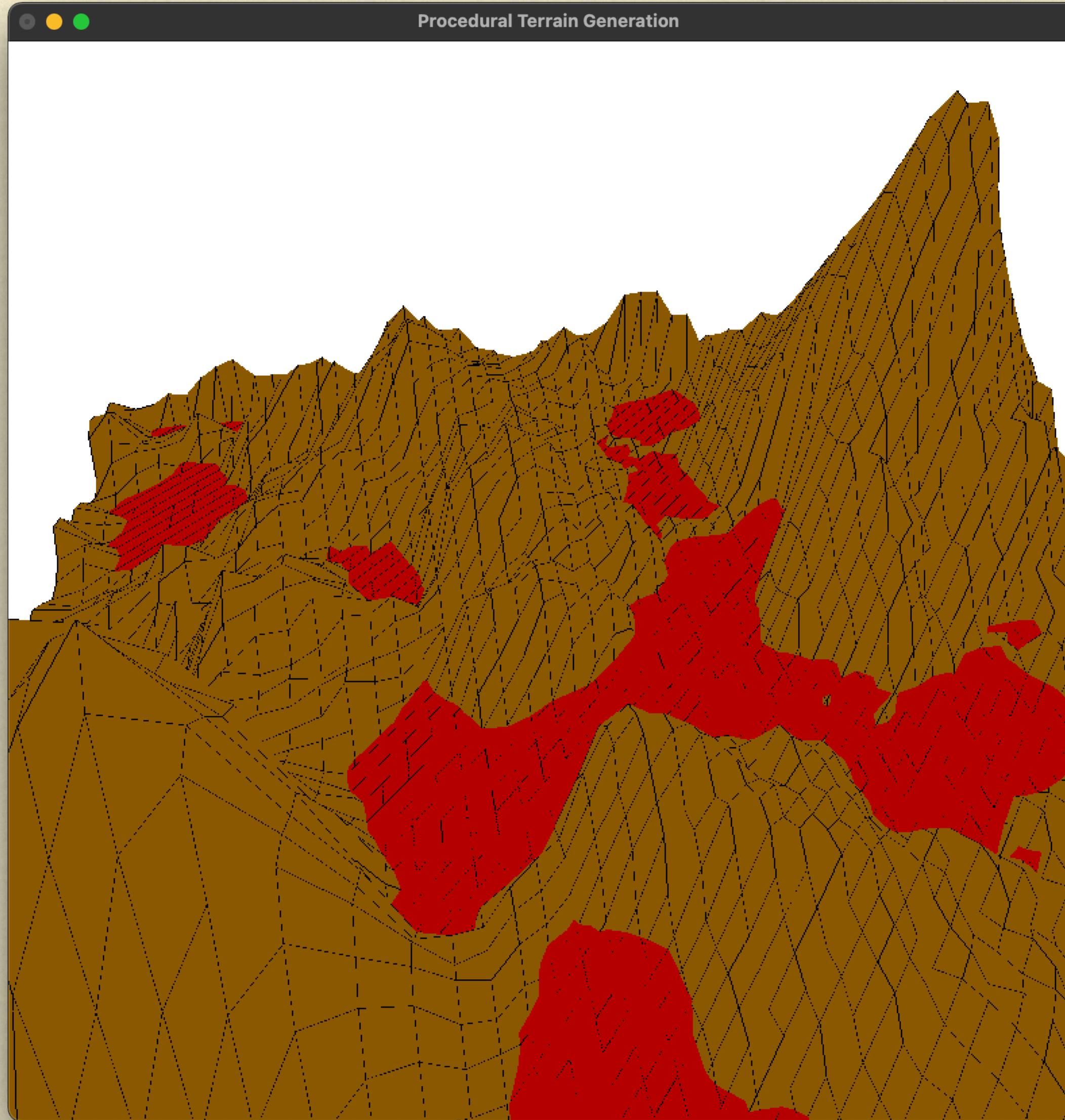
- Dimensions, terrain characteristics
- ‘Encouragers’
- Terrain/water color





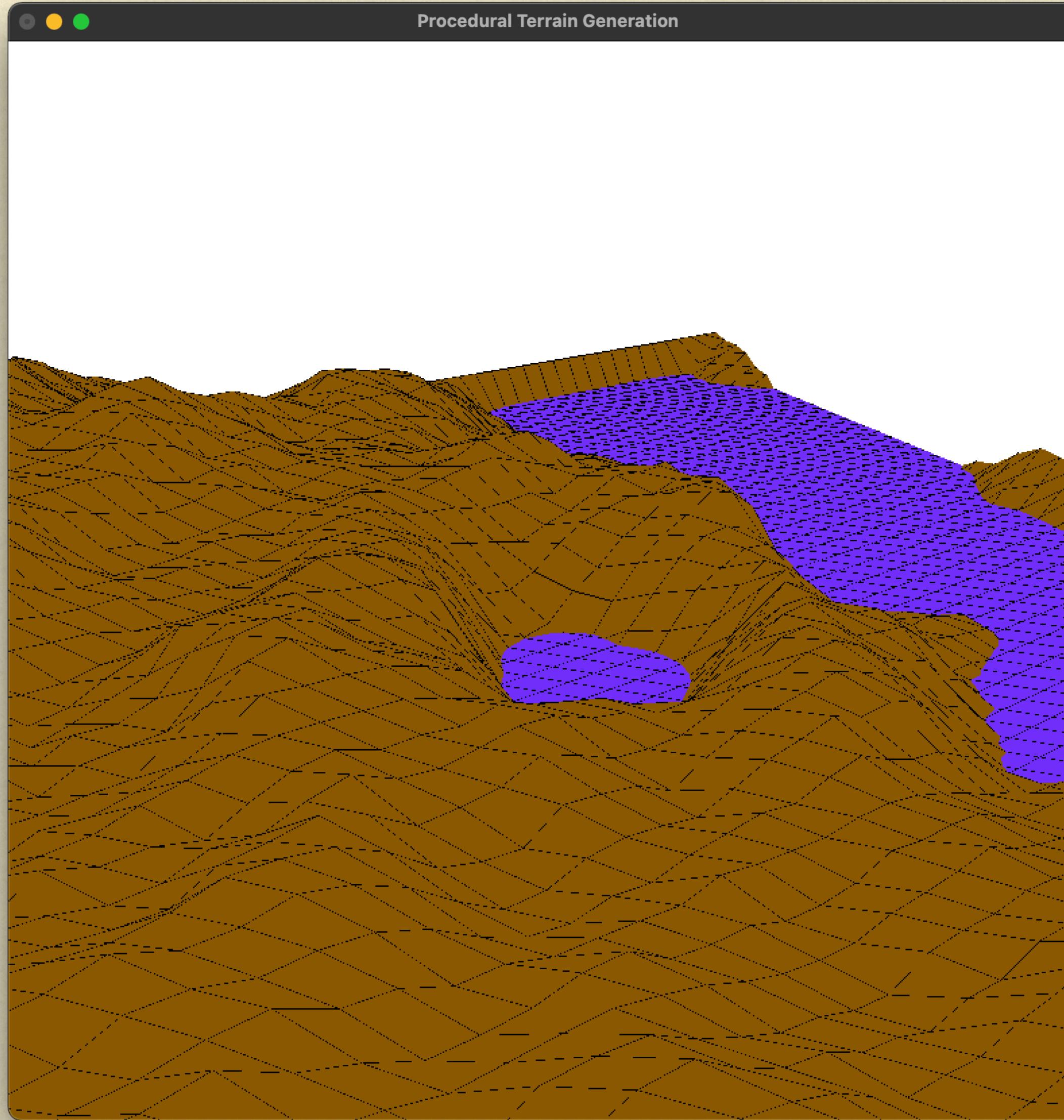
islands.txt ▾

```
50 50
12 3 2.0 1 1.3
C 0.92 0.71 0.20 0.30 0.72 0.9
```



lavalake.txt

```
50 50
15 8 4.0 1 -2.0
M 10 10 15 40
M 40 40 15 40
R 15 15 35 35
C 0.51 0.36 0.02 0.65 0.0 0.0
```



crater.txt

```
50 50
15 8 2.0 1 -2.0
M 25 25 15 50
L 25 25 30
C 0.51 0.36 0.02 0.41 0.20 0.94
```

LIVE DEMO
