

Procedural generation

Jaan Janno

17th of October 2016

Setup to get ready for later

- Download/install eclipse if necessary
 - <https://eclipse.org/downloads/>
- Download exercise repository
 - <https://github.com/JaanJanno/SimpleTerrain>

Procedural generation

- Generating lots of assets
 - Small amount of stored data
- Textures, animations, terrain, etc.
 - Even non visuals

Pseudorandom functions

- Same seed, same result
- Example of terrain in a game
 - Same terrain after each save/load.

Random number generators

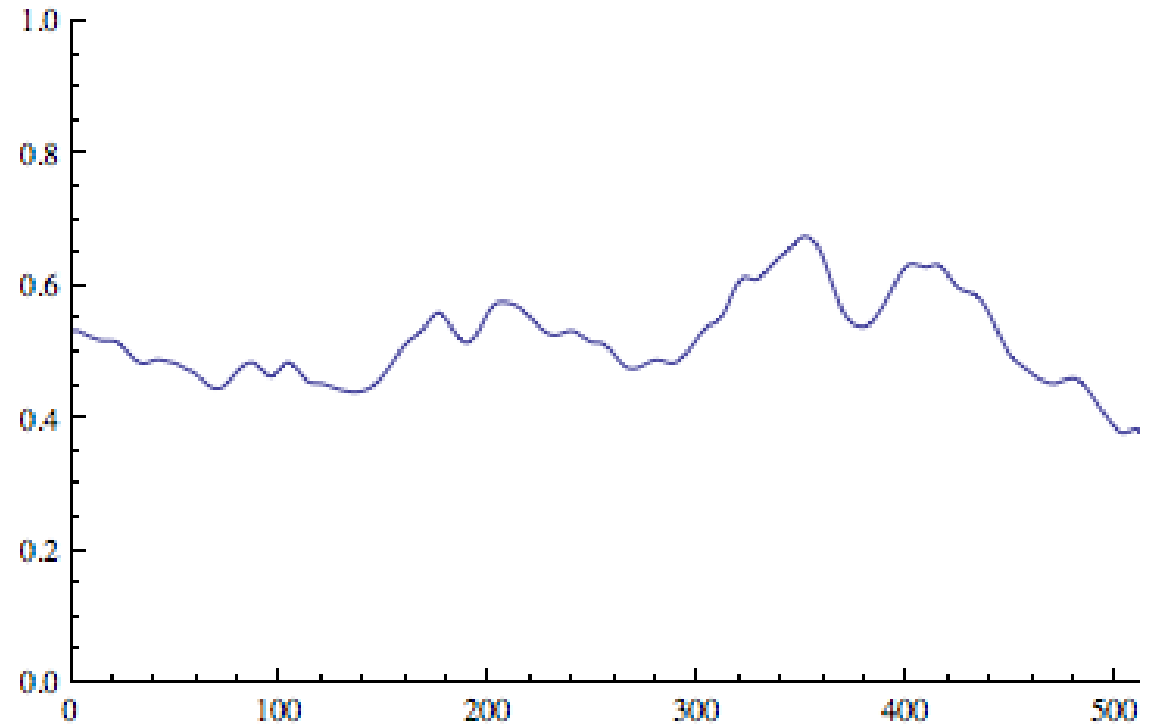
- Seed
- Sequence of pseudo-random numbers

Noise functions

- A continuous function.
 - n real-number arguments.
 - 1 real-number value.
- Pseudo-random

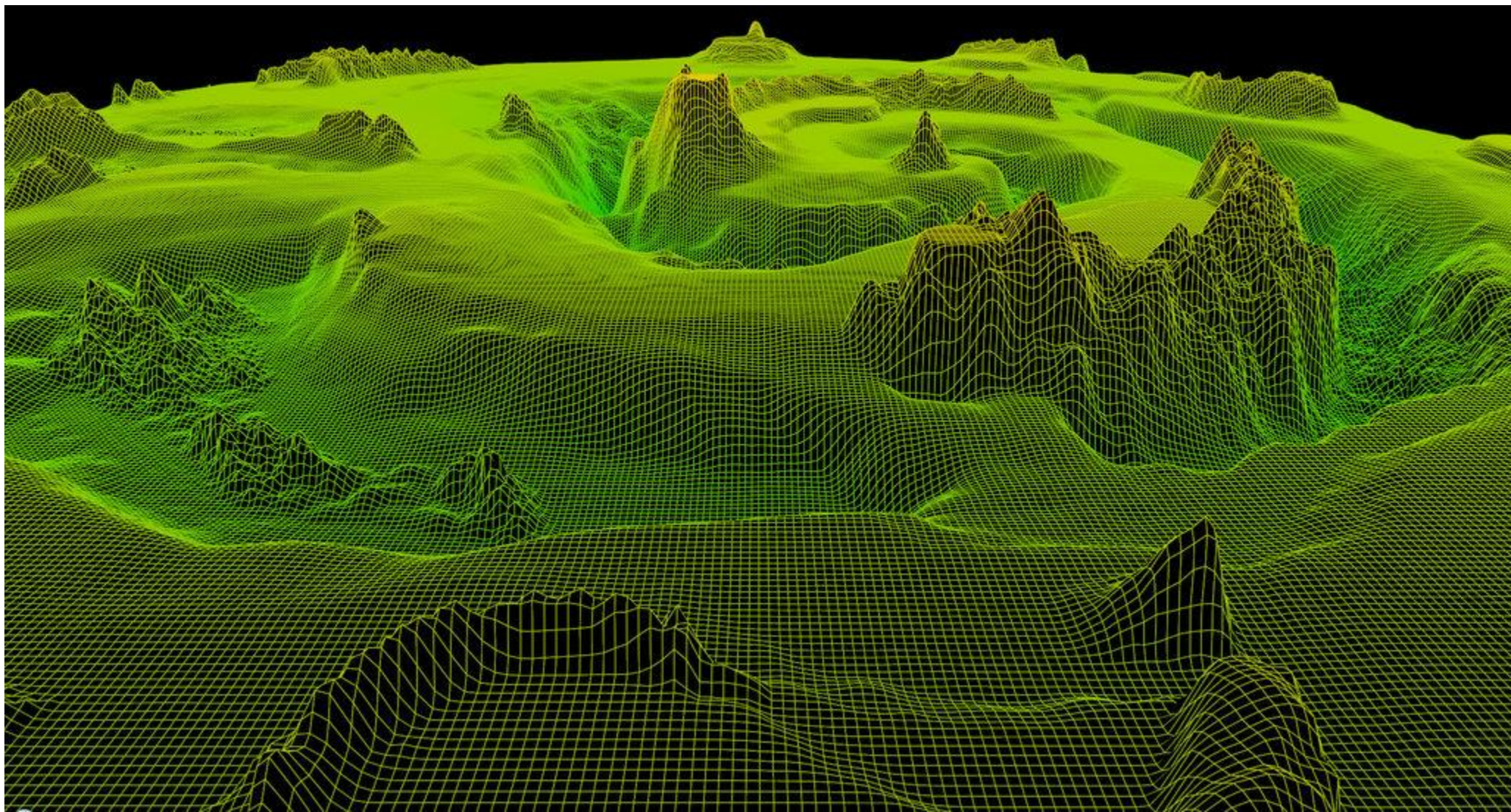
Noise functions

- 1-d



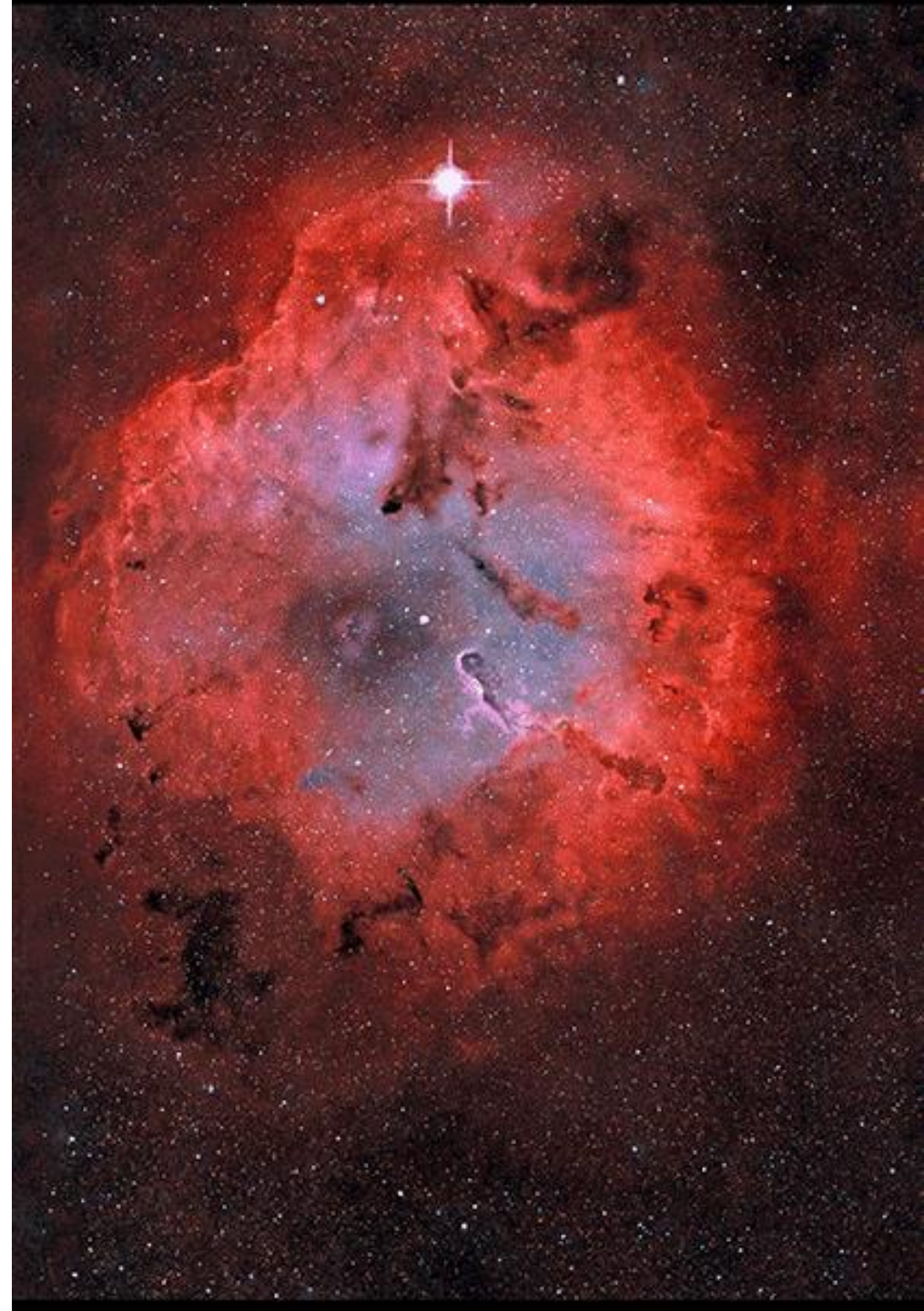
Noise functions

- 2-d



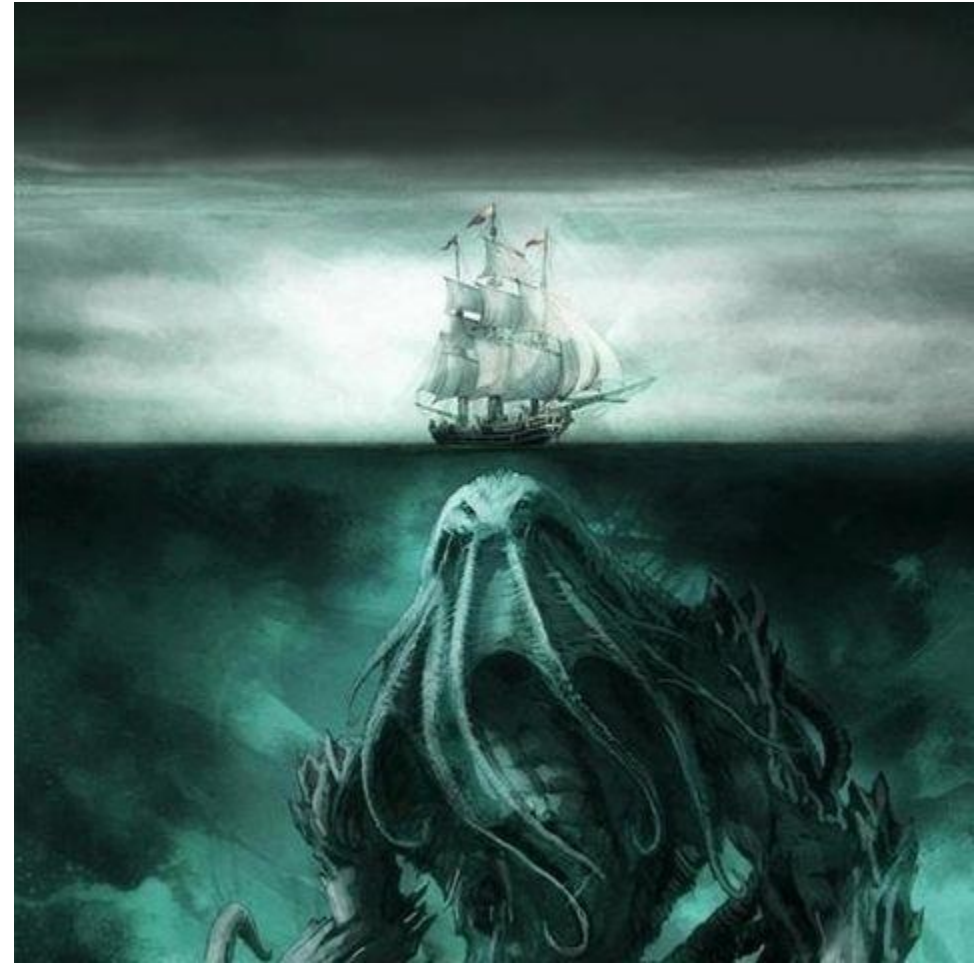
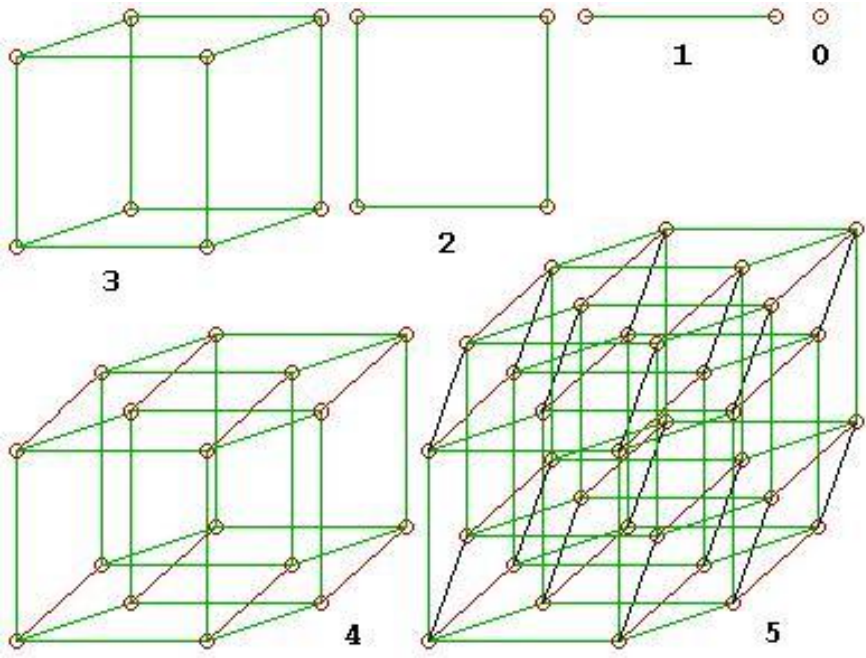
Noise functions

- 3-d
- Spatial of progression of time



Noise functions

- N-dimensional



Example of texture

Animations

- Time as a dimension

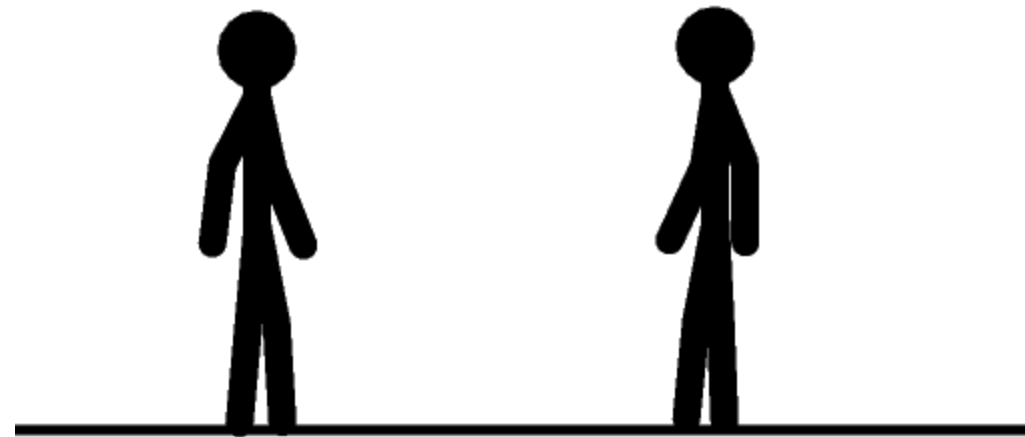


Looping animations

- What we need



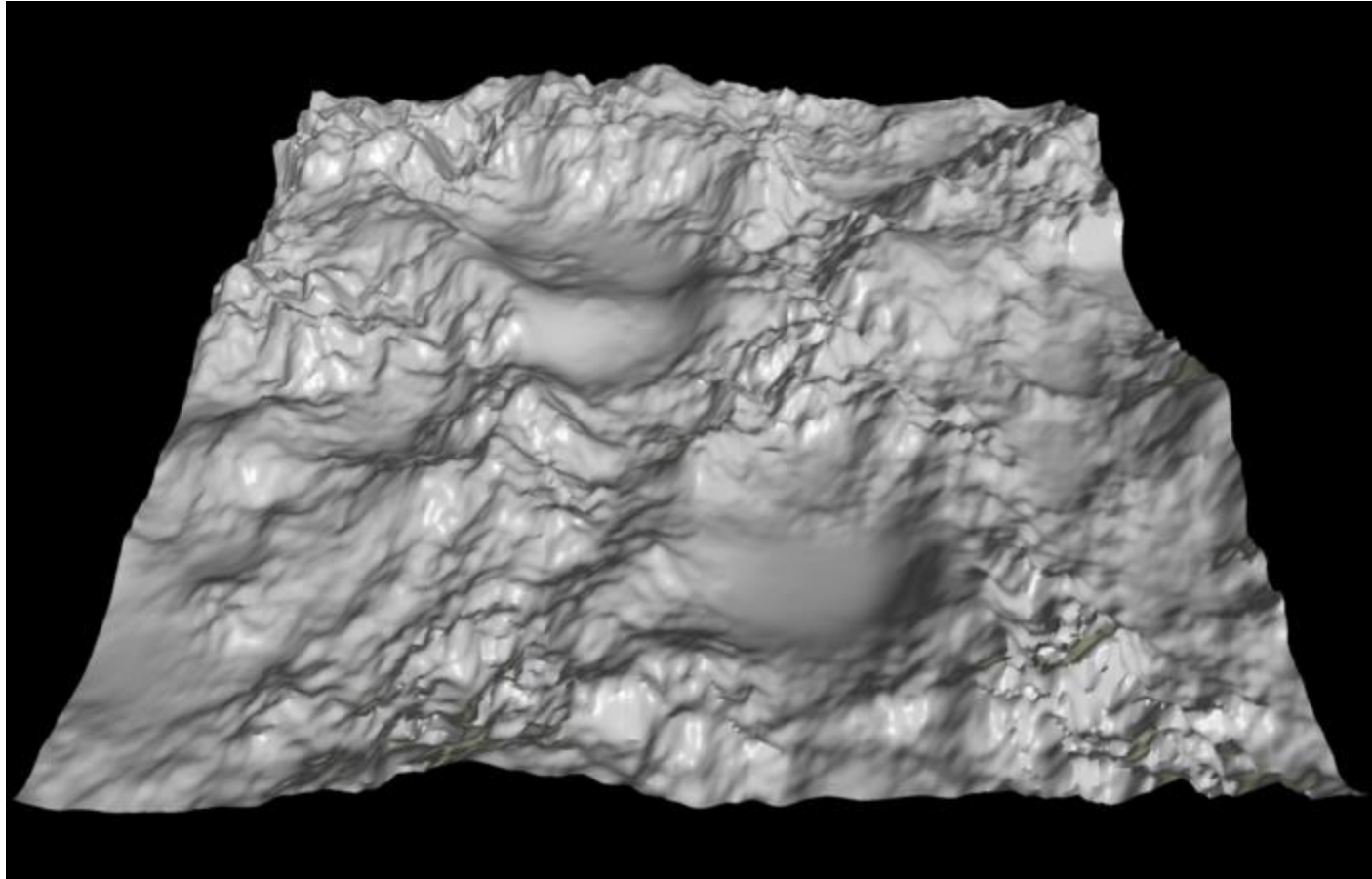
- What usually don't want



Example of an animation

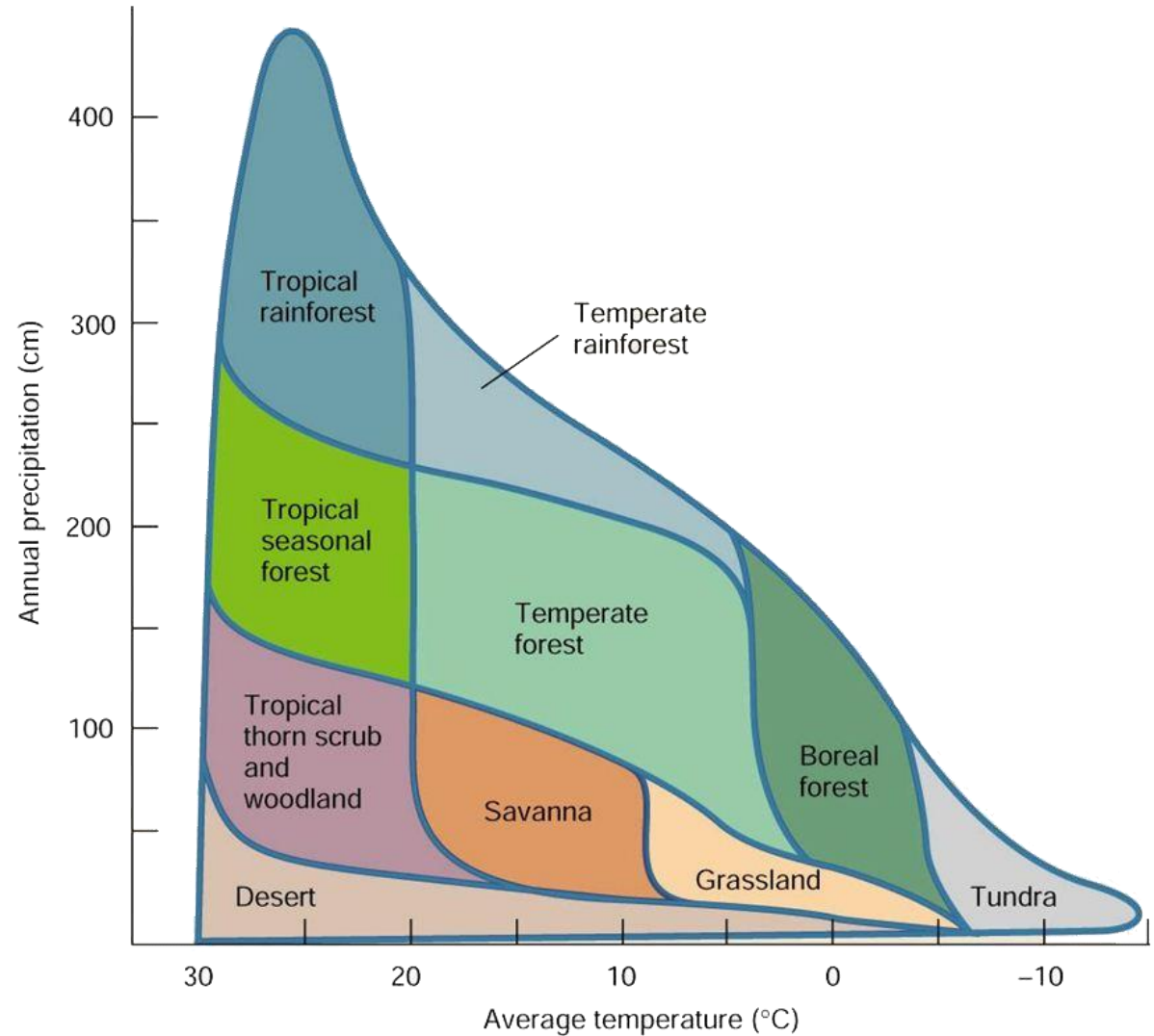
Terrain generation

- Heightmap



Terrain generation

- Varying features



Lets try it ourselves!

- Download/install eclipse
 - <https://eclipse.org/downloads/>
- Download exercise repository
 - <https://github.com/JaanJanno/SimpleTerrain>

Expectation vs reality

- <https://www.youtube.com/watch?v=2Pl4NzN6q4M>

The end 😊