Background:
We're developing a program that procedurally generates worlds through the use of noise based fractal algorithms (more about this in a bit) in addition to evolutionary, macro-scale algorithms such as the modeling of tectonic plates, large-scale erosion, wind/precipitation patterns and Holdridge life zones. Using these algorithms we will render a voxel-based world, with the hopes of eventually moving on to a gradient-colored mesh grid or something more advanced/pretty.

Current Status:
Sam can draw squares now (so many squares; not infinite squares)
Tectonic plates implemented
Graphics code working, mostly (as voxels)
Diamond-square coming along
Textures mapped

Problems:
We still have too much to do (chunking, erosion, life zones, diamond-square texturing, caves, correct oceans, volcanoes? (probably not))
Erosion is real hard, also processor intensive
How do we make diamond-square work on existing heightmap?
Chunking is now working? Is this a problem I don't know Sam help
Memory and processor constraints (we are representing/evolving an entire world with points)
We are probably not evolving this ^ honestly

Harder Problems for the Future:
How do you combine fractal algorithms with evolutionary algorithms
Caves?
Does anyone understand large-scale geological things cause we don't