Week 2 Project Reflection

| Anar Nyambayar

Repo: https://github.com/pomegranar/compsci208

This week, I created two works using contrasting rule-sets: 'Fields', which defines line-tracing entities of RGB traversing a grid of randomly generated vector fields that manipulate their trajectory in unexpected ways. And 'Tessellation', which features a tessellation of hexagons with squiggly topsides that overlap and add color to form an illusion of the Duke Kunshan University logo.

For both works, my method was to start with something simple, and while in my coding if an error occurs that resulted in something that I could understand and build off of, I kept it.

For Fields, this was when the actual lines stacking more and more right where the grid was divided allowed me to showcase the grid without explicitly drawing black lines. I see it as something that emerged from my rule-set rather than something I intended myself.

For Tessellation, it was the overlapping triangle shapes that appeared when I miscalculated a for loop that inspired me to reshape it. I intentionally calculated the exact shape that needed to be drawn to create a form of tessellation—that only when overlapped in a specific way—created the tessellation.

So although the tessellation appears as a result of the rule and not the rule itself, I believe it is not an emergent element, since it is something that I expected and intended. This sits in contrast with Fields, in which the fields can only be obviously seen by the viewer as a consequence of an unexpected, accidental result of the rule-set playing out.

As for the colors in Tessellation, I think the fact that they hover around a green-blue-purple hue (analogous color harmony) **is** an emergent feature simply by virtue of it being an accident. (I thought it would randomly choose any color from the color wheel). I can try to come up with a reason as to why it tends to choose these colors, much in the same way that an evolutionary biologist comes up with reasons for life on Earth. But that would be doing a disservice to my initial rule-set that unexpectedly created something that's more complex and structured, even if it's simply Perlin noise mapped to a value of 0 to 360. It wouldn't be useful to call every result of an accident 'emergent'.

As stochastic it may be, the grid that appears in Fields shows itself consistently in every run of the program, which I think should be criterion for emergence. If the phenomena does not consistently occur given the rule-set, then it is merely coincidence.

In creating art from rule-sets realized through code, I find the question of authorship more and more complex. Any thought or concept I think of will be shaped and reformed by not just the medium I transfer it to, but also by the eyes and minds of those that perceive it, including myself. Parameter tuning therefore largely became pointless unless it affected the concept or rule in a direct way.

I author the concept and pass it onto my hands to code, which passes it onto the Java interpreter, passing it onto my eyes to perceive, interpret, and judge. The artwork becomes something entirely new on every pass, adding another chance for emergence each step. The same rules must apply to any medium, therefore any medium is capable of emergence.