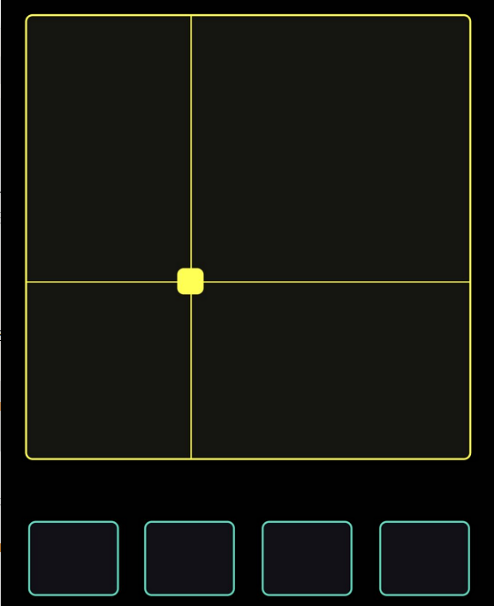
Assignment 4

The purpose of this project is to further explore the Langton’s ant simulation and add cool effect on the existing project.

Development:

On top of assignment 3, this project can be roughly described into 3 phases.

1. Tinkering and deciding what can be done to further develop the project.

I spend some time to fix the toroidal issue. Now once the ant step on the edge, it will show up on the other side of the canvas.

Afterward, I could decide what can be done next. I think of doing a two-person game utilizing OSC. So, one person can use the phone and the other person uses the mouse to play with each other. I successfully implemented the mouse drag as well as receiving OSC signal from this panel. In this case, dragging both the mouse and the panel will paint a line to the canvas.

I then planned to do a game where one person drag to draws a line and the other person drag to erase pixels. However, I cannot erase the painted pixels no matter what I do. During the class presentation, I was hit that maybe I can fine a way to implement age with each cell. So after a cell is paint it dies in X amount of time.

1. Implement the age idea.

The idea is simple, for each frame, once the pixel is turn on, I add time to the pixel. Once the time is more than a threshold, the pixel dies.

In order to do so, since it is hard to store local variable for each pixel between frames, I decided to build another texture to record age info. But I then realize if I can compress my directional system to use only 1 channel, I can use another channel to store the age info. So I squeeze the direction into 0 for up, 0.1 for right, 0.2 for down and 0.3 for left and do marginal check to avoid precision problems.

After I change my directional system I set an uniform float, growthRate. For any “on” pixel, I add growthRate to alpha channel and number reaches 1, reset to zero.

On draw shader, for each pixel, I check its r channel to see if it is turn on, if so, I use its alpha channel as HUE value and set saturation and brightness to 1. Hoping to see the color changes as the time goes and eventually going back to red and died.

1. Aesthetic polish.

To achieve smoother transaction, the growthRate should be small. I started with the lifespan of 3 seconds (grothRate = 1/(60\*3)) and find the life span too short. I then envisioned a canvas of pixel slowly changes color and died in 10 seconds. Therefore, the growthRate should be 1/ (60\*10). However, this number does not work (output is all red). I even change precision to high but that does not help. I eventually landed on a magic number 0.0022. Any number smaller than it will not work. And for this magical number, all pixel will freeze at cyan. I really like this effect and decided to leave as is.

Even though the simulation is toroidal now, it is not as exciting to look at as the buggy version where multiple ants were spawn at the edge. So I decided to disable the toroidal mechanics.

Feedback:

People generally like the color schemes comparing to the last project. Threads created by edge case, especially coupled with colors Ping-Ponged in two ends, was mentioned by different people. They also like the effect of how ants were frozen and died after iced.