My GitHub Link: https://github.com/yrrrng/Coding2-Final-Work_Shaohua-Yang

Coding2 Final Work:

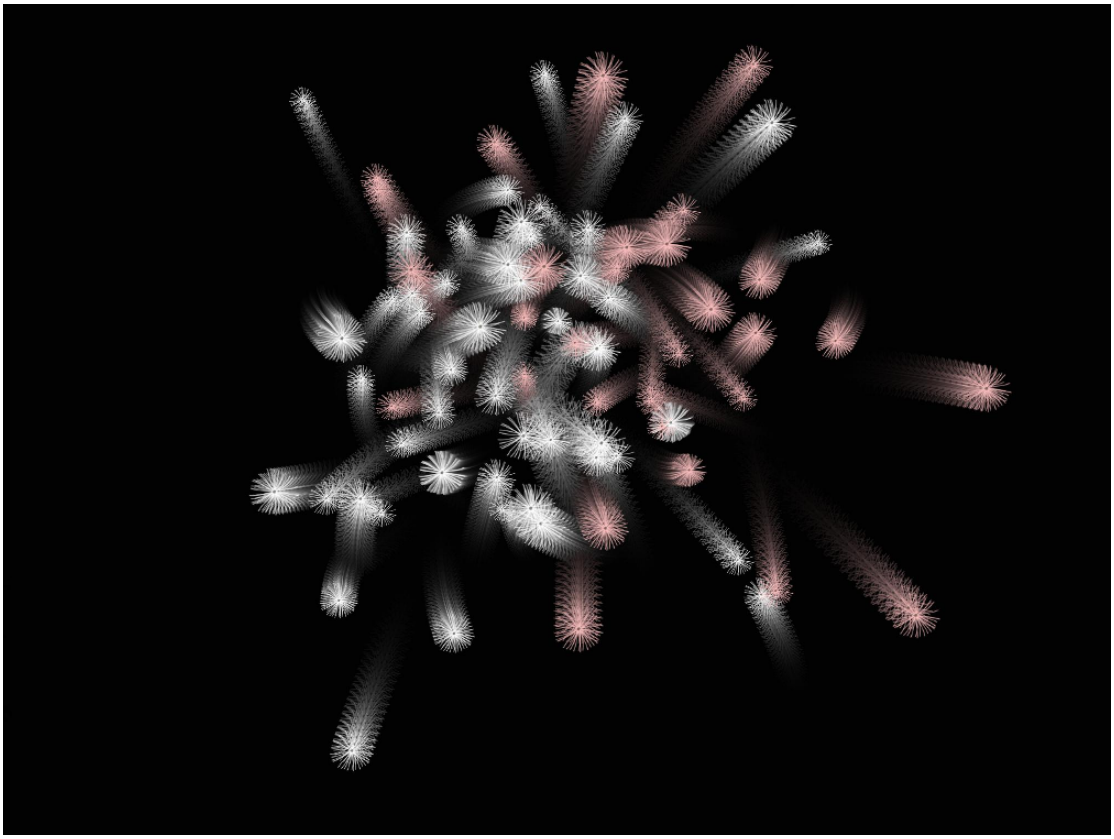
Video Link:https://www.youtube.com/watch?v=SL-H8N4FXus

Panopto Video Link:

https://ual.cloud.panopto.eu/Panopto/Pages/Viewer.aspx?id=c662d6f8-8f43-44a1-a091-afc801050e50Overview

"I tried to find something interesting to relieve the anxiety inside me". Anxiety is often hidden in the noisy darkness and once you find it and drag it into the spotlight and blow on it hard, it will drift into the air like the seeds of a dandelion. I use the machine as a medium to establish a binary opposition between me and my troubles, and use dandelions as clues to connect what I call anxiety. When we drive the machine to operate, these troubles will also float into the air.

First demo video (Youtube):https://www.youtube.com/watch?v=7_ixJ00Of9c

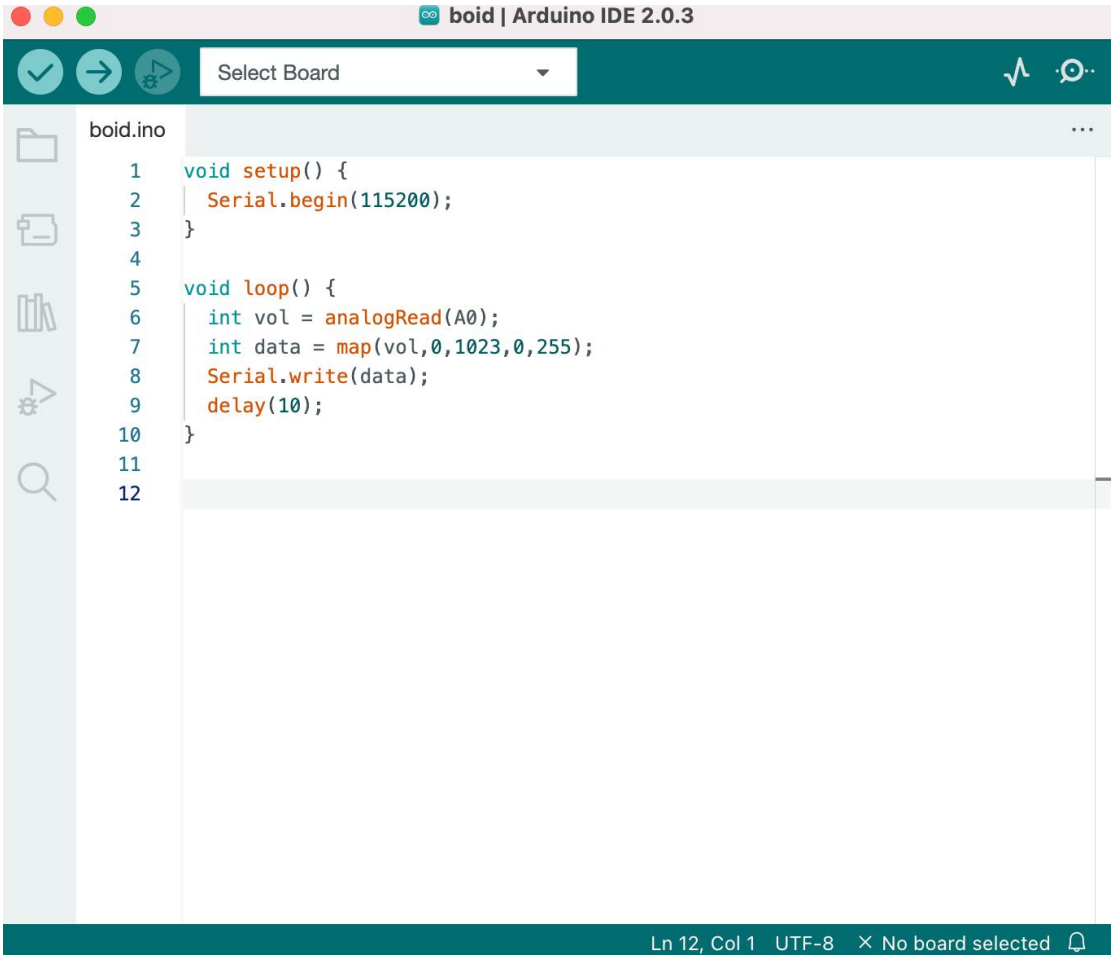


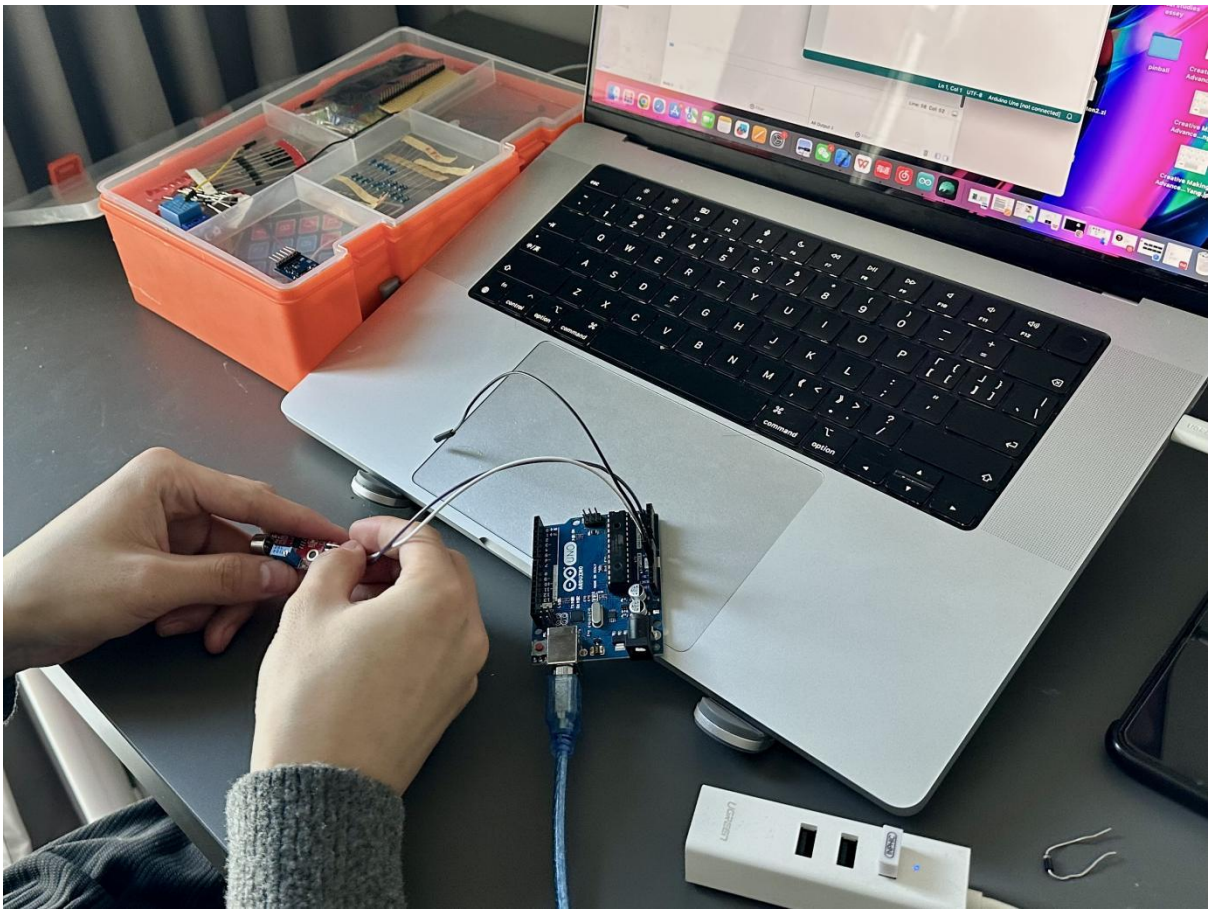
Outcome:

This project is an interactive work that the user can blow to the device to make the generated virtual flowers fly with the wind. The program mainly consists of three parts: (1) Arduino - OpenFrameworks Communication System. (2) Gerative System of Virtual Flowers. (3) Interactive Control Flower Behavior System.

1. Ardino-Openframeworks Communication System.

In this system, we utilize an Arduino-based microphone for detecting the user blowing. It can analog input the volume data to Arduino A0 with the random from 0-1023. We use the function "analog read" to read the data and map it to 0-255 because we want to transmit the data to OpenFrameworks through a serial port with the "Serial.write" function which can only transmit 1 byte. In the OpenFrameworks, I use the "serial.readByte" function to receive the data.





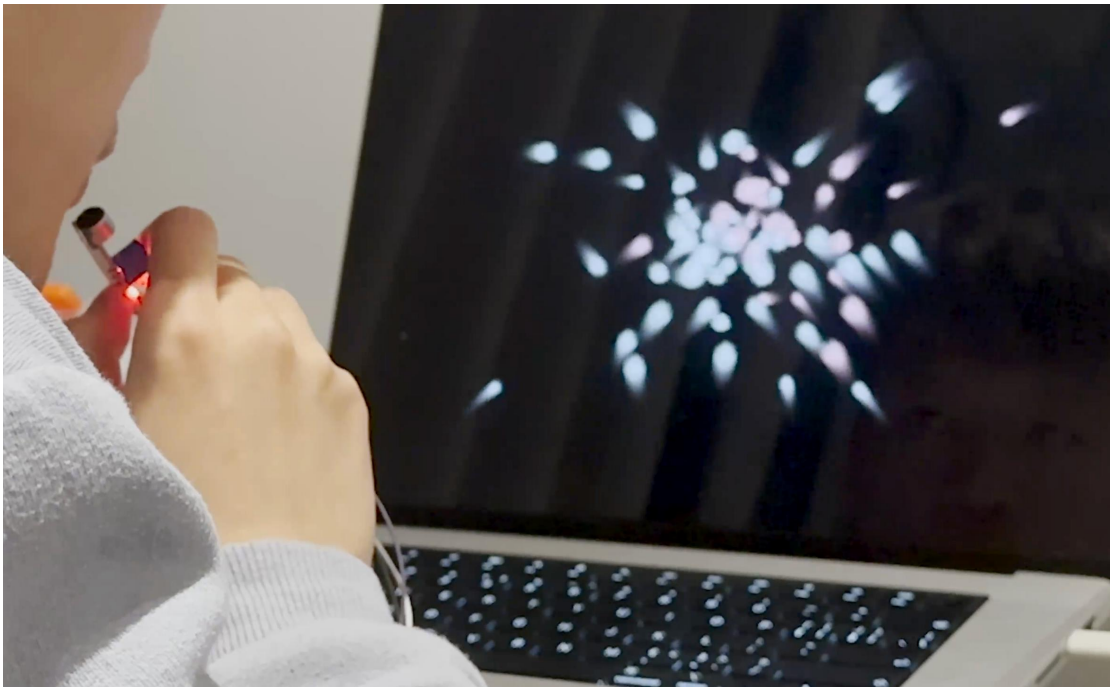
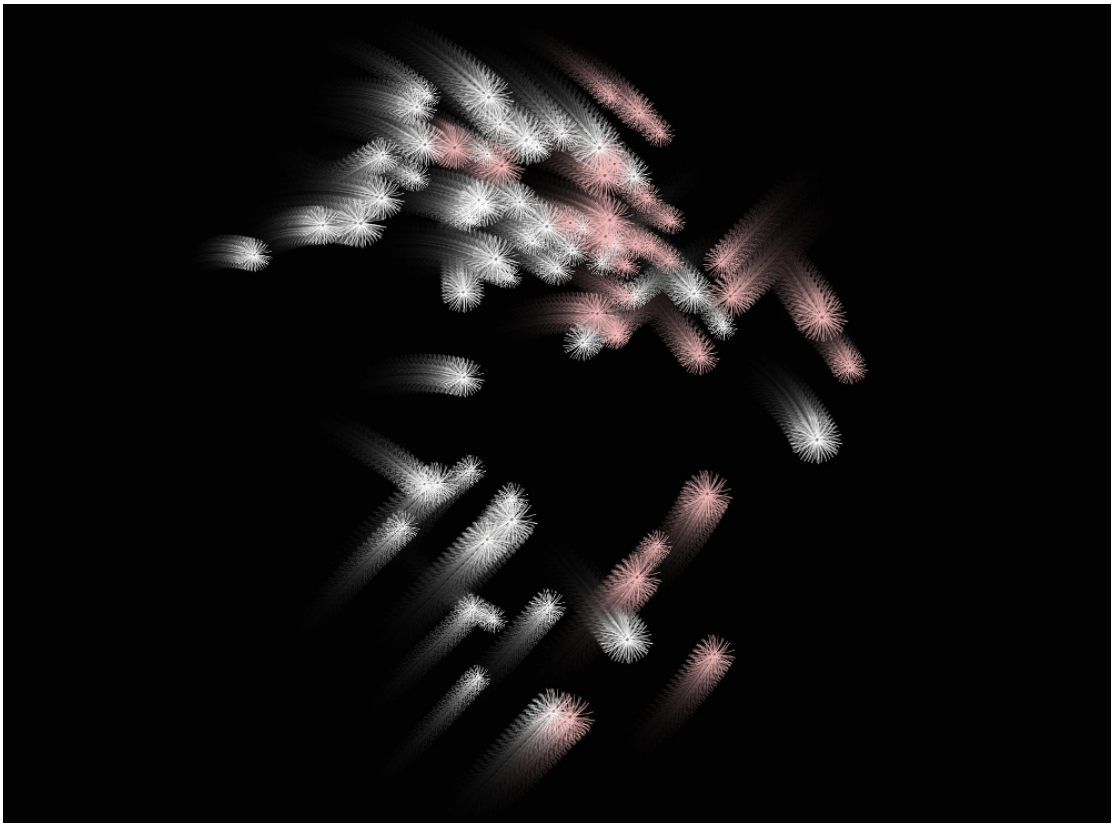
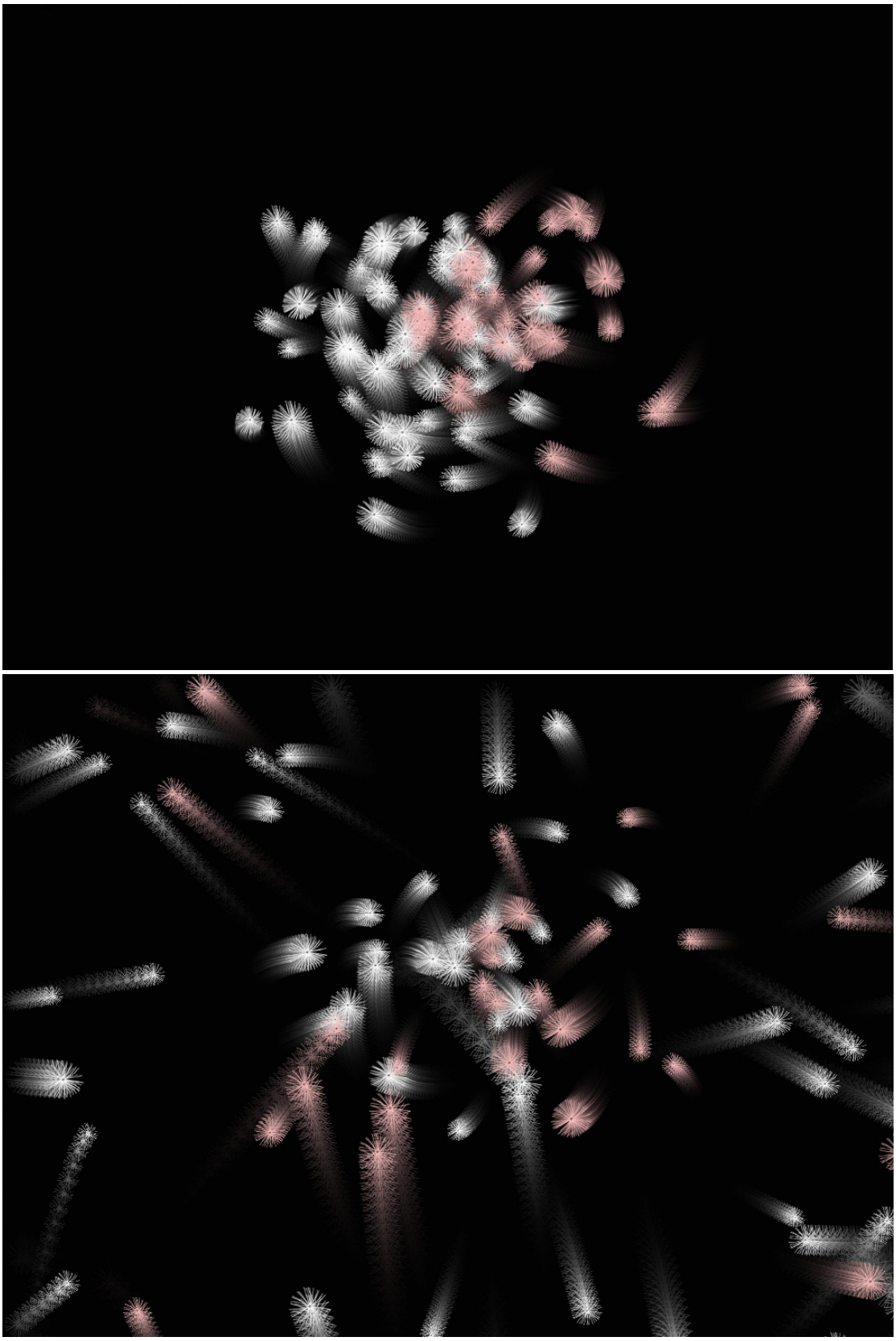
2. Gerative System of Virtual Flowers.

In this system, I write a main class and a subclass to manage the flower object. In the main class, I control the movement of the flower by using three main functions. The first one is "separation()" which calculates the distance between each flower object inside the array, if two flowers are too close, it will calculate a force based on the vector of two flowers to separate them. The second function is "cohesion()" which targets the mouse position and tracks all the flower moves toward the mouse. It also calculates the force based on the vector between the mouse and the flower position. The last one is the "update()" function which updates the velocity of the flower based on the "separation()" and "cohesion()" function which returns the force vectors, then modifies the position of the flower. It also implemented "wall()" to detect whether the flowers have exceeded the boundaries of the canvas and used "constrainSpeed()" to limit the maximum speed of the flower. The class uses a "draw()" function to visualize the flower based on a texture image that is a png file I created in advance. In the

sub-class, most of the parameters and functions are inherited from the main class, except for three functions. The first "new draw()" allows us to draw flowers of different colors by setting the color. The other two functions had to be modified because the input parameters needed to be changed to the subclass pointer object.

3. Interactive Control Flower Behavior System

In order to make an interactive system in which the user can control the movement and the sound of the program, I map the received data (which has been mentioned in part 1) to 3 parameters. The first one is "blow_force" which can adjust the separate force of the flower inside the "separation()" function. The second one is "blow_maxForce" which adjusts the range of the max speed of the flower. If the user blows the device, it will increase the force of separation and the boundary of maximum speed, allowing the flowers to separate from each other and move violently. I also map the data to "whiteNoiseVolume" which will affect the sound output volume. This program achieves the sound output by using the default function "audioOut()", which generates the sound based on the output array. The system generates random numbers and inputs them into the output buffer to produce white noise. The range of the "random" function affects the volume of the output, which corresponds to the "whiteNoiseVolume" parameter I have set.



Photography