CSC395 S23 Flocking Project

Due Dates March 7 End Day

Tuesday, March 7 before class

1. **Identify 3 resources related to computer simulations of flocking behavior (in addition to the 2 videos referenced below).**
    - Provided Links:
    https://www.youtube.com/watch?v=QbUPfMXXQIY&ab_channel=dante
    https://eater.net/boids
    - **Adnan's Link:** https://cs.stanford.edu/people/eroberts/courses/soco/projects/2008-09/modeling-natural-systems/boids.html
    - **Sam's Link:** https://www.youtube.com/watch?v=4LWmRuB-uNU (I really enjoy this youtuber)
    - **Ermais's Link** : https://codeheir.com/2021/03/27/the-flocking-algorithm/#:~:text=Flocking%20is%20a%20behaviour%20that,%3A%20 separation%2C%20 alignment%20and%20 cohesion.
2. **Repo is created. List your resources in the README.md.**
    - https://github.com/SammyWint/FlockingProject ✅
    - Create README.md ✅
3. **List of anticipated methods that will be needed (in the planning document/directory).**
    a. Public void averagePosition() -Public void averageDirection() -
    b. Public void findNeighbor() - Public void updateGUI() - Display all current positions of circles

4. **A documented process for commits (in the planning document/directory)**
    a. Let's talk as a team on 03/07
    b. Should we push when we have updates for a class or work on classes together

5. **New model created for flocking with a large population.**
    New model created for flocking with a large population.
    a. Increase circles to 40(Change the code below)

    b. Increase GUI size

    c. Change the pane size

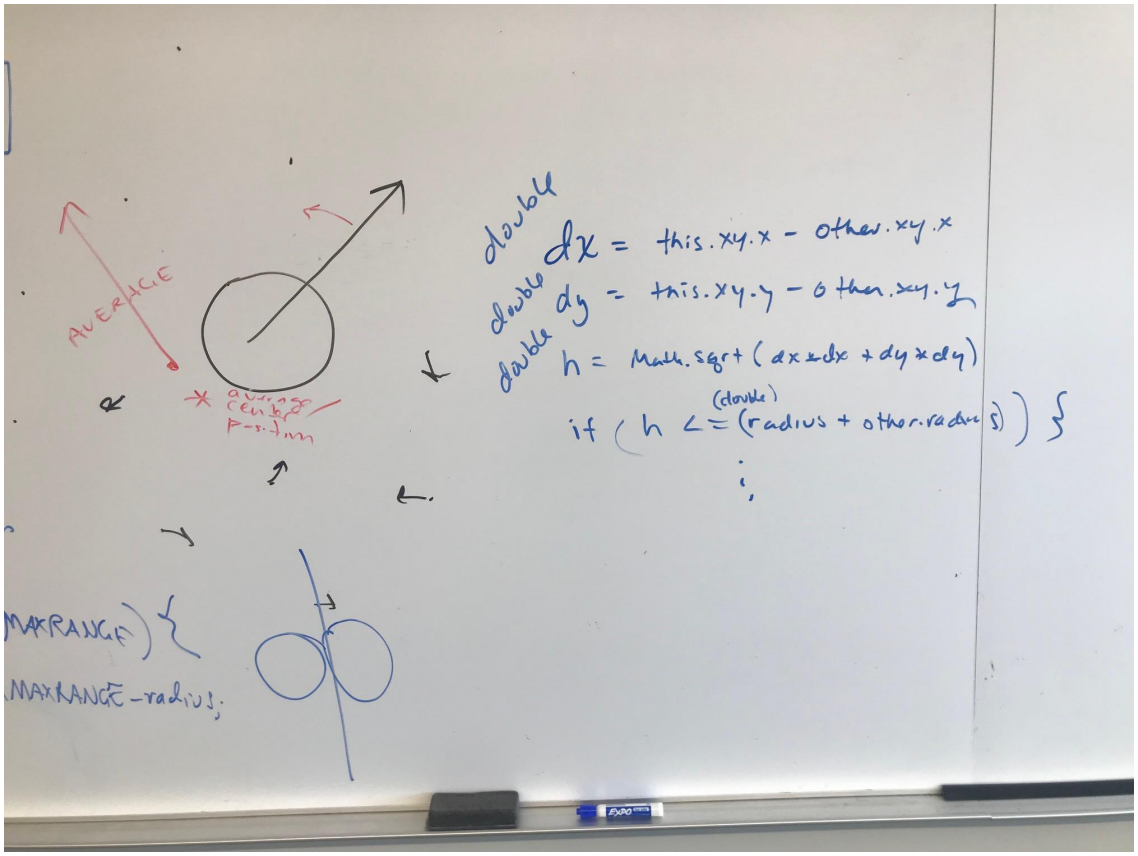
6. **List of all methods**

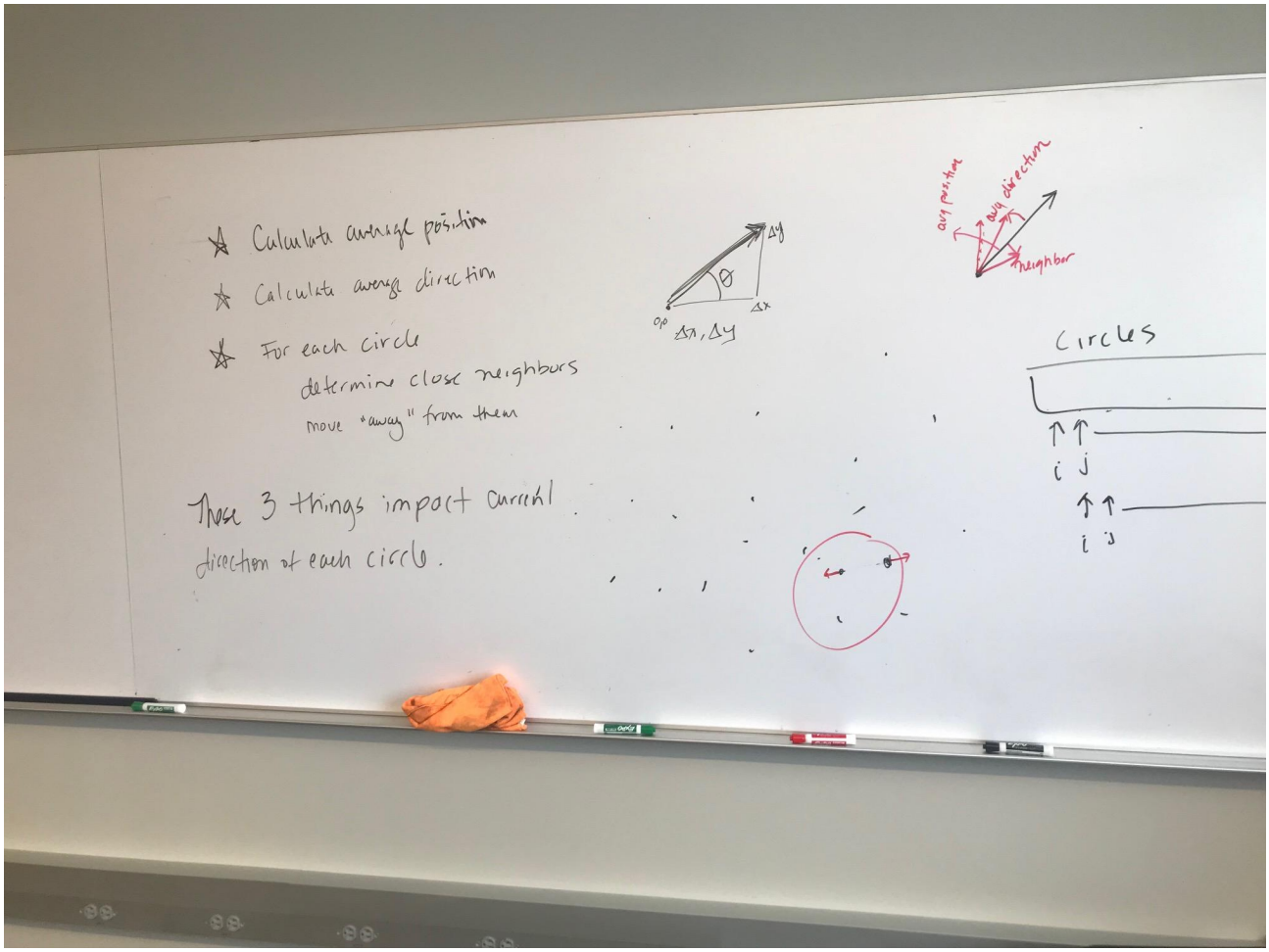
Class Notes: March 2, 2023

Additional In-Class Project Notes:

- ➔ Calculate average position
- ➔ Calculate average direction
- ➔ Vector or angle

- For each circle
  - ➔ Determine close neighbors
  - ➔ Move away from them
  - ➔ These 3 things impact current direction of each circle(either circle class or model class)


AVERAGE
double dx = this.xy.x - other.xy.x
double dy = this.xy.y - other.xy.y
h = Math.sqrt(dx*dx + dy*dy)
if (h <= (double)(radius + other.radius)) {
MAXRANGE) {
MAXRANGE-radius;


Calculate average position
Calculate average direction
For each circle
determine close neighbors
move "away" from them
These 3 things impact current direction of each circle.
Δy
θ
op
Δx, Δy
avg position
avg direction
neighbor
Circles
i j
i j