Rose Dufresne

6th November 2017

I.D. #40030476

Technical pitch

Summary:

For my final project, I am going to make a competitive two player game called (for now) *Chicken Feed*. Each player in this game plays as a very confused chicken desperately trying to survive by trying to find the most food as possible before the other. Why is the chicken confused you ask? Well, finding food isn't as easy as it seems in this game. The reason being because these chickens have absolutely no sense of orientation and constantly turn in circles until a player tells them to go in a given direction. However, the player needs to time the orientation of the chicken at the right moment to be able to get the correct angle towards a grain. The game will look relatively simple:

-Each player has a different looking chicken to be able to differentiate between them (one white chicken and a brown one for example).

-The grain they need to eat will simply look like an appropriately sized seed for the players to see and distinguish from the grassy background.

-The size of the game will be 640x480 pixels.

What will happen and how do you interact with the game?

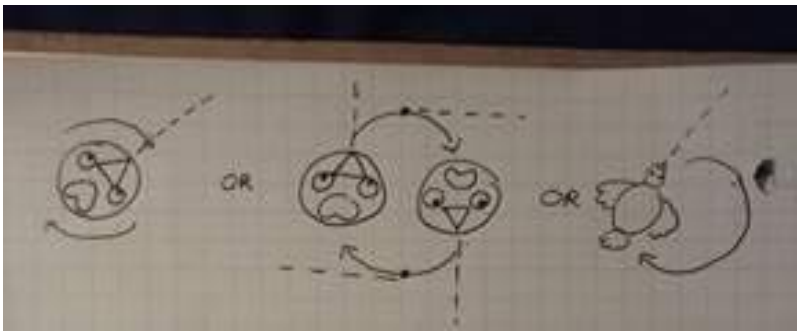
-When a player manages to eat a seed, he/she will gain a point (which will be shown with a score counter).

-When player 1 presses the space bar, the chicken will move in the direction where it stopped turning in circles. If the space bar is pressed again, the chicken will go back to going in circles. For player 2, the same mechanics are applied, but the button pressed will be the shift button.

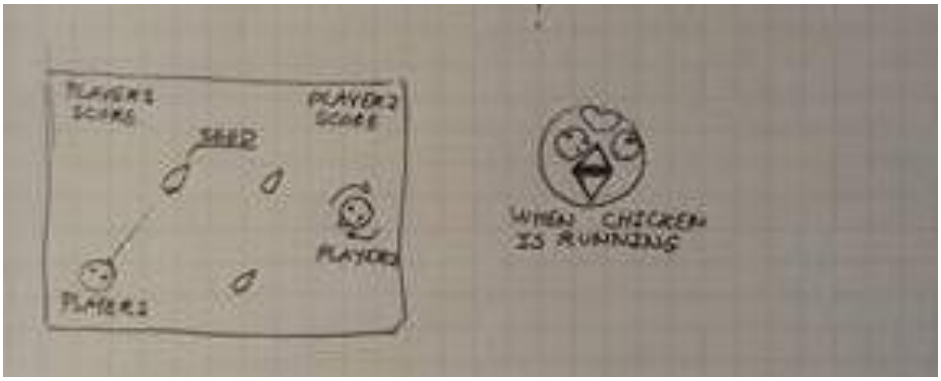
-There will be special seeds (ants) that give more points and that will disappear if not eaten on time.

-Seeds will generate more and more as the game progresses until someone gets 20 points.

Media:



These are my ideas for how the chickens will be represented. I want to make the game simple enough for the user to automatically know exactly how to play and so that the design of the chicken is straight forward enough to enable easy gameplay. Having a defined and big beak will show the exact direction the chicken is looking at. I haven't quite decided if I would prefer the chicken to turn on itself or turn around a point. I believe if it turns on itself it would be easier to avoid any boundary issues; however, if the chicken turned around a point, it would amplify the realism of the game because the chicken's movement would be more organic and closer to real life (and thus more funny to look at).



As seen in the left image, the game will look relatively simple. The cores are displayed at each corner of the window, the seeds are easy to see and so are the chickens. Maybe I will implement obstacles. Of course, I have to make the game funnier by changing the chicken's face when its starts running to get its food!

Inspiration:

This game idea simply came from a video I saw of two AI objects learning to fight each other with trial and error by shooting bullets. I liked the movement of these two AIs because it required good timing of orientation to shoot in the right direction. I liked the challenge and thought it would be fun to use this as the base mechanic for a two player game. (see video)
https://www.youtube.com/watch?v=u2t77mQmJiY

Technical approach:

Classes:

There will be a "Chicken" class that will control how a chicken moves around using typical Bouncer mechanics we have seen in class. The Bouncer will act as normal when moving, that is, moving at constant velocity in a uniform direction, while bouncing on the edges of the window. However, other methods with me implemented, such as rotate() that will cause the circle representing the chicken to stop moving in x and y and to rotate on itself. There will also be a keyPressed() method to allow the chicken to move again.

A second class will be a "Seed" class that will generate the seeds at random locations in the window. Will somehow try to find a way to cause the generation of seeds to increase over time.

There will also be special class called “Ant”, which will generate ants at random locations (just like seeds) and will give 2 points instead of only 1 if eaten. These ants will move to make them harder to eat. They will utilize the noise() function to make their movement unpredictable and organic.

Finally, there will be the “Score” class. This class will count the score or each player by adding one point if a player eats a seed and two if he/she eats an ant. If a player reaches 20 points, he/she will win the game.

Of course, this will all compile together in the main class, where all objects will be drawn and will act respectively to their class’ functions.

Controls:

Space bar for player 1 (player starts on the left and has white chicken) will stop the chicken from turning on itself and will direct it to move in the same direction/orientation it was pointing at the moment the key was pressed. Chicken will continue to move until the space bar is pressed again.

Same mechanics for player 2, but with the shift key. The chicken starts at the right side of the window and is brown.

Ants:

Utilize noise() to go across the screen. Once the ant reaches the edge of the window, it will disappear.

Technical research:

Might make use of vectors to move the chickens instead of our previously used methods for our bouncer classes that only updated the position of the ball at ever frame. Vectors seem to be easier to control if any specific movements need to be done.

https://processing.org:8443/tutorials/pvector/

Of course, using rotating mechanics are at use in this project.

https://processing.org/reference/rotate_.html