

Easing

There is an attempt by some users of Processing to make movement more “natural” and/or more pleasing. The use of the `noise()` function can be used to simulate motions such as flocking behavior of birds as well as natural forms like clouds, land, and water.

Easing is another technique. It is not a function but the use of a variable – the easing coefficient – to modify the distance traveled by a figure on the screen.

In this example John used easing to move a smaller “eased circle” to the position of the mouse marked by a larger circle. The code associated with this set of notes (cc09_3) turns off the cursor with the function `noCursor()` so you only see a larger circle at the position of (mouseX, mouseY).

The strategy is to move the eased circle only a fraction of the way to the mouse location and slow the eased circle’s speed as it nears the mouse location. This is similar to the motion of a car stopping when it is approaching a stop sign (except in Pittsburgh where the letters **STOP** stand for “**Speedup To Obliterate Pedestrians**”).

Let’s look at John’s code in segments. His entire code is available on the class code web page (cc09_3). Please note that Jim could not leave well enough alone and he altered some of the names of variables and functions. John would probably never used these...

First the variables:

<pre>float easingCircleX, easingCircleY, distX, distY, easingCoefficient;</pre>	<pre>x coord of the smaller eased circle y coord of the smaller eased circle distance between eased circle and mouse location this controls how far the eased circle moves.</pre>
--	---

<pre>void setup() { . . . easingCircleX = width/2; easingCircleY = height/2; easingCoefficient = .02;</pre>	<pre>initial x coord of eased circle initial y coord of eased circle value of the easing coefficient</pre>
--	--

The initial coordinates of the eased circle are arbitrary. If you do not like them, you can alter them to suit your tastes.

The easing coefficient is set an arbitrary value that John liked. This can be altered when the program is running by pressing the [UP] and [DOWN] arrow keys.

```
void draw()
{
  backgrounder();
  drawMouseCenteredCircle(40);
  drawEasingCircle(20);
  showStats();
}
```

The only new stuff in the draw() function is in the function drawEasingCircle() so we will just look at that function next. If you have any questions about the other three functions, talk to John or Jim in class or office hours.

<pre>void drawEasingCircle(float dia) { fill(180); noStroke(); distX = mouseX - easingCircleX; distY = mouseY - easingCircleY; easingCircleX = easingCircleX + (distX * easingCoefficient); easingCircleY = easingCircleY + (distY * easingCoefficient); ellipse (easingCircleX,easingCircleY, dia, dia); stroke(255); line (easingCircleX,easingCircleY,mouseX,mouseY); }</pre>	<p>Note 1</p> <p>Note 2</p> <p>Note 3</p> <p>Note 4</p> <p>Note 5</p> <p>Note 6</p>
---	---

Note 1, Note 2: These two lines of code determine the distance between the eased circle and the mouse location.

Note 3, Note 4: If John just set the coordinates of the eased circle to mouseX and mouseY, there is no easing. The two circles never separate and look like a target:



Since the idea of easing is to ease the smaller circle into the position of the larger circle that is at (mouseX, mouseY), John wants to move the circle only a small fraction of the distance between the current location of the eased circle and the larger circle. To do that, he multiplies the distance variables (distX and distY) by the easing coefficient. If he is using the initial value of the variable easingCoefficient, he is multiplying the distance variables by the value of 0.02 which is a small fraction of the distance to the larger circle.

As the eased circle gets closer to the larger circle, the actual amount moved by the eased circle decreases resulting in the motion of the eased circle slowing down.

Note 5: John draws the eased circle at the newly computed position that is closer to the large circle.

Note 6: John draws a line connecting the centers of the two circles to give the effect of the larger circle drawing the eased circle closer.

Here is a shot of the working code taken while Jim was moving the mouse. Notice that the eased circle is blurred a bit as it is trying to move closer to the larger circle:

