Simple Algorithmic Design: Random, easing, noise

DECO1012 - Design Programming

Week 3

THE UNIVERSITY OF SYDNEY

Recap: Generativity

THE UNIVERSITY OF SYDNEY

Recap: designing with generative processes

Generative processes are techniques which can be used in a design to achieve variation in the resulting design (what the user experiences). This variation may be deterministic in relation to a given input or, it may instead, be indeterminate---meaning that the output of the generative process may be different, even for the same input.

THE UNIVERSITY OF SYDNEY

Random

THE UNIVERSITY OF SYDNEY

Random values

Random is an example of an indeterminate process, which we can use in our own generative designs.
Random can generate random numbers for us. We can use these numbers to represent all kinds of things we want to randomise.

You must be careful with random because you get **different results every time** which can lead to flashy and rapidly changing values. This chaos from random can also make designs messy and cluttered.

However, letting the computer generate values itself can be key to introduce an element of surprise into a generative design.

THE UNIVERSITY OF SYDNEY

The random() function

In p5, the random function lets us generate random decimal values. By calling the random function with different inputs you get different results:

random() — returns a random number from 0 up to 1 (but not including 1)

random(upper) — random number from 0 up to upper (not including upper)

random(lower, upper) — random number from lower to upper (not including upper)

THE UNIVERSITY OF SYDNEY

Challenge 1:
Introduction to Random

Challenge 1: Introduction to Random

You now have everything you need to attempt the first challenge on canvas.

Remember to consult the p5 reference pages.

Random: This challenge requires you to generate all kinds of random numbers. Remember that when you call random() you get a different result each time. You may need multiple random variables to store all the information you need.

Make sure to ask for help if you need it.

THE UNIVERSITY OF SYDNEY

Easing

THE UNIVERSITY OF SYDNEY

Easing values

The use of easing presented in the following section is an example of a deterministic process. This process can be used to generate movement in a design.

Easing lets us smooth animations to look more natural. There are lots of types of easing, and we will only cover an example today.

We can apply easing to all kinds of different values that change over time, including position, size, and color.

For more types of easing, check out easings.net which also has some handy javascript like implementations. These more advanced easing functions might be a bit hard at this stage, but you should revisit this slide in the future.

THE UNIVERSITY OF SYDNEY

Linear movement: each frame the circle moves the same amount

Ease-out: movement starts faster and ends slower

Easing: using lerp

The lerp command lets us linearly interpolate between two values. We provide a start value, a stop value, and the amount to interpolate between.

We can use lerp to smooth motion, in this example by setting xPos to 10% of the way to xTarget.

```
xPos = lerp(xPos, xTarget, 0.1)
```

p5 also provides us with a convenient lerpColor() function which works the same way but with colors. Experiment with how lerpColor() and colorMode() interact.

THE UNIVERSITY OF SYDNEY

Challenge 2:
Introduction to Easing

Challenge 2: Introduction to Easing

You now have everything you need to attempt the second challenge on canvas.

Remember to consult the p5 reference pages.

Animation: Step two asks you to trigger the easing animation automatically using the frameCount variable, another special variable provided to us by p5 which tells you how many frames have been drawn so far.

You can use random(height) to get a new square size that fits most screens.

Make sure to ask for help if you need it.

THE UNIVERSITY OF SYDNEY

Noise

Noise values

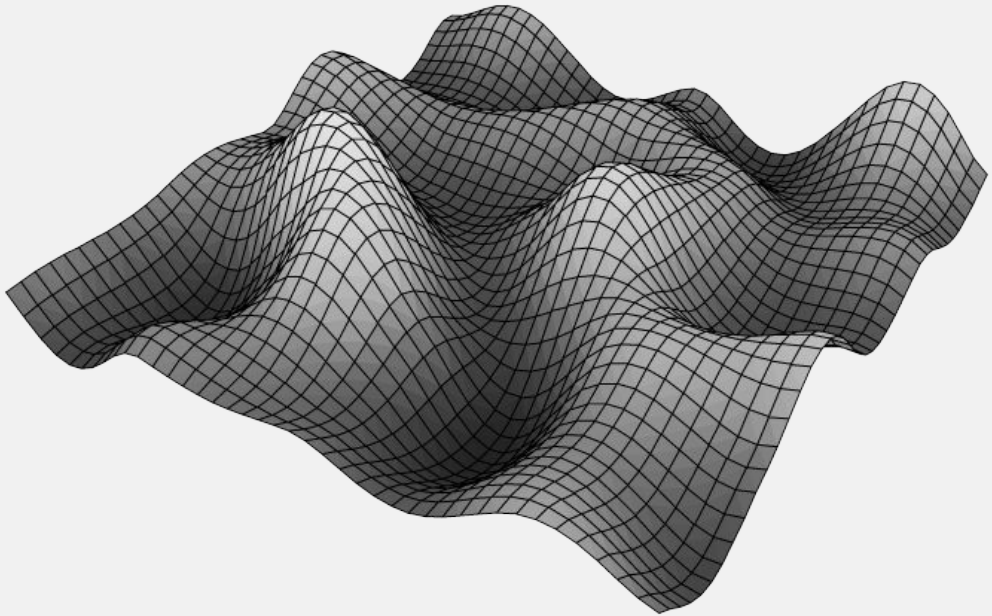
Noise lets the computer generate numbers for us that are more smooth than random. Rather than each number being completely unique, the values of noise change smoothly overtime.

Noise is therefore used to create things that look more natural and real, or smoother.

You can think of the ‘noise-space’ as like a mountain range that you travel through. At any place, the noise at that coordinate is like the height where you are standing. As you traverse through the space, nearby values are similar.

THE UNIVERSITY OF SYDNEY

Noise space



THE UNIVERSITY OF SYDNEY

The noise() function

In p5, the noise function behaves differently to random. Rather than passing in a range for the output, noise always outputs a value between 0 and 1. The value we pass into the noise function is our coordinate in that 'noise-space' and the value we get returned is like the height at that coordinate.

You can call the noise function with just an x coordinate or can optionally specify a y and z coordinate too.

noise(x, [y], [z])

THE UNIVERSITY OF SYDNEY

Setting a seed value

One interesting feature of noise is that, unlike random, calling noise multiple times with the same input will give the same output.

To reset the noise space and get new values we need to restart our program.

If you want to generate the same noise space every time, you can use the noiseSeed() function.

You can set a randomSeed() also so that the random function generates the same random numbers in the same order.

THE UNIVERSITY OF SYDNEY

Challenge 3:
Introduction to Noise

Required Challenge - Introduction to Noise

You now have everything you need to attempt this recommended challenge on canvas.

Remember to consult the p5 reference pages.

Noise scale: Try dividing your inputs by different results to get different scales of noise.

Remember that noise always returns values from 0 to 1, so often mapping is required.

Make sure to ask for help if you need it.

THE UNIVERSITY OF SYDNEY

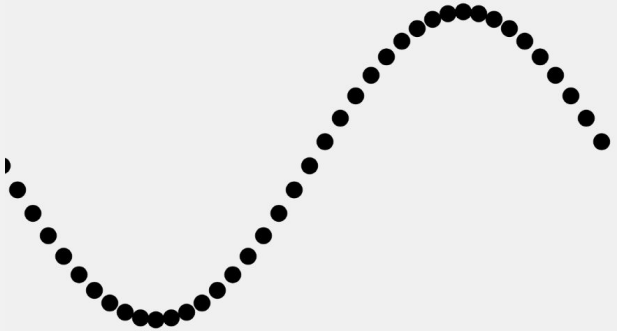
Sin and Cos

Sin and Cos

They can also be used to create smooth waves and animations. Both sin and cos take in an angle that, as it continues to go up, will return values that oscillate between -1 and 1. (the difference is that sin starts at 0 and cos starts at 1)

```
let waveSize = 100;
angleMode(DEGREES);

for(let theta = 0; theta <= 360; theta += 360/40){
     let yPos = sin(theta) * waveSize;
     circle(map(theta, 0, 360, 0, width),
     yPos + height/2, 10);
}
```



Challenge 4: Step 1
Sin and Cos

Challenge 4: Sin & Cos

You now have everything you need to attempt the first part of the fourth challenge on canvas.

Remember to consult the p5 reference pages.

THE UNIVERSITY OF SYDNEY

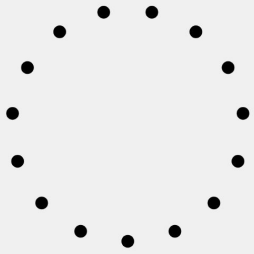
Sin and Cos

Math can be scary but in p5 it also lets us create many beautiful designs.

Sin and cos can be used to calculate trigonometry, such as the points that make up a circle:

```
let radius = 100;
angleMode(DEGREES);

for(let theta = 0; theta < 360; theta += 360/15){
     let xPos = sin(theta) * radius;
     let yPos = cos(theta) * radius;
     circle(xPos + width/2, yPos + height/2, 10)
}
```



THE UNIVERSITY OF SYDNEY

Challenge 4: Step 2
Sin and Cos

THE UNIVERSITY OF SYDNEY

Challenge 4: Sin & Cos

You now have everything you need to attempt the second part of the fourth challenge on canvas.

Remember to consult the p5 reference pages.

THE UNIVERSITY OF SYDNEY

Further recommended and extension challenges

There are a few more challenges you can complete this week outside of class. These further expand the ability for the computer to generate values for us.

- Random walker — Create a walker that moves randomly
- Complex patterns — Combine random and nested loops to make patterns

THE UNIVERSITY OF SYDNEY

Before we wrap up…

Try and re-try this week's challenges before the next class. Practice helps!

Assessment 1: Part B quiz will be released on **9th March at 17:00.** Complete this quiz by **15th March 23:59 (AEDT). Simple extensions will not be granted for this quiz.**

If you have any questions and would like to talk to your tutor, arrange for a meeting using Canvas (option to "setup a meeting" available in the Home page).

THE UNIVERSITY OF SYDNEY

See you next week!

THE UNIVERSITY OF SYDNEY