MUS 4711 –
Interactive Computer Music

Course Lecture Notes

Intro to MaxMSP, Math, and Computer Logic

PART 1b: Math and Logic

Objects - [+], [-], [*], [/], [pow], [sqrt], [maximum], [minimum], [abs], [%], [<], [>], [<=], [>=], [==]. [!=], [||], [&&], [expr], [decide], [random], [urn], [print]

In analysis – [del], [uzi], [counter], [join], [route], [i]

Review Order of Operations

- Right to Left, top to bottom
- Right most sets arguments within an object
- Review HOT and COLD inlets
- Output is also right to left, just on the bottom of an object
- Demo with Print
  - Three separate messages (one) (two) (three)
  - One CS message (one, two, three) – THIS WILL BE REALLY IMPORTANT WITH OSC
  - One NON-CS message (one two three)
- Flow is top to bottom – unlike a lot of text languages like SonicPi or SC3 that are inside out

Hello World, Math, Logic, Keyboard

Let's start by creating the most basic patch of all time: "Hello World."

-Make sure the console is up, create [print] attach a (Hello World) to it.
-Hit the message. Wow…

Ok – pretty boring – let's get MaxMSP to do some math for us.

-Basic math functions are the [+] [-] [*] [/] [pow] [sqrt] objects
  -Note that [pow] takes two inlets – cold inlet on right sets the exponent
-Also, [maximum] and [minimum] – both of which have to be primed on the right
-[abs] for only positive, absolute values
-[%] modulo – outputs the remainders only. VERY useful with [/] for 12-tone pieces…

Play with math objects for a bit in conjunction with messages and floats/ints – Make sure to do a [/] and [%] examples

Let's look at Logic

-[<] [>] [<=] [>=] [==] [!=]

-All of these output a 1 if the result is true, or a 0 if it's false. These will come in handy for triggering specific things to happen only when you want it to…

-[||] or, and [&&] and

-[||] Logical Or compares two numbers and outputs 1 if *either* number is non-zero, and a 0 if *both* numbers are zero.

-[&&] Logical And compares two numbers and outputs a 1 if the two numbers are *both* non-zero, and a 0 if *either* number is zero

Play with logic objects in conjunction with messages and floats/ints

There's got to be an easier way, let's look at the [expr] to create expressions.

-Expressions are algebraic equations. The variables are indicated by $i1

-$-type-number

-$ specifies that it's a variable

-The type tells [expr] what type of variable to check for (i = int, f = float)

-The number specifies what inlet it's coming in from – inlet 1 always triggers the calculation

-This is one of the most complicated objects, so you'll be looking at the help patch frequently.

Play with expr

Random numbers are easily generated with the following objects

-[decide] gives you a 1 or a 0 – it's a coin flip

-[random X] gives you a random value from 0 to X-1. The X here is an argument, an initial value you specify at its creation.

-[urn X] is like random, but it will cycle through all the values between 0 and X-1 without any repeats. VERY useful.

Data Types!

Data (Max) – grey cables

Objects

Messages

Lists

MIDI

Audio (MSP) – grey/yellow cables

ALWAYS designated with a tilde ~

Tells Max to operate at the sample rate of your audio interface, MUCH faster than normal control rate

Can only connect to objects with a tilde~

-LOTS of data objects have an equivalent – e.g. [+] and [+~]

BEAP – MaxMSP's modular synth environment uses both types

Video

Jitter – green/black cables

-The OG video system for Max. Lets you rapidly create video synths, but processes the visuals on the CPU, which makes the framerate less than ideal for complex patches

jit.gl – blue/black cables

-OpenGL based extension of Jitter that uses meshes, shaders, textures, and the super powerful [jit.gl.pix] object. Processes video by offloading it directly to the graphics card, so your framerate is noticeably faster, but requires a lot more to learn

VIZZIE – MaxMSP's modular video synth environment was Jitter based in Max 7, but has been updated to jit.gl in Max 8

Analyze 12 Tone Generator – why doesn't it work?

Max Puzzle 2 – Download the Puzzle 2 patch. Write a short description for why each of the numbered patches inside of it does not work. Upload the Word Doc with your answers.