Creative Coding Spring 2016

The “π-ano” (Pi-ano).

Team 7
Members:
Christel Tiu | Vinit Kumar | Alexander Marck

Inspiration

The group’s initial concept was to create music using math, specifically the fibonacci series. The Fibonacci series is a sequence of numbers: 0, 1, 1, 2, 3, 5, 8, 13 and so forth, where the next number is the sum of the previous two. This pattern is observed in a number of fields such as art but also in nature. The group wanted to see what the pattern would produce in terms of musical notes through our own Sonic Rube Goldberg Machine.

After the initial brainstorming meeting, the group not only continued researching the fibonacci series and past applications on music but also extended its scope to other mathematical concepts such as the golden ratio and infinite sequence of pi. The result: a Max patch draft that consisted of a “π-ano” (Pi-ano).

Concept

We then updated our patch to include a musical beat. We developed a patch that utilized PI music where there are 15 instruments that change randomly every 4 seconds. (i.e after every 4 seconds, random instruments play random notes of the PI digit). We are also using several sets of scaling techniques in order to utilize the entire fret of a guitar or the entire keyboard of the piano.

The machine runs in this particular pattern: 4 seconds playing the digits in an octave then scaling up the note and playing it on a different instrument, then scaling down for another 4 seconds and then playing the PI note on a different instrument. Through this we have achieved both the serial and parallel process requirement of the project.

For note conversion to ASCII, we took a base of 12 and the two remaining higher notes were given A and B value. We then took 17 digits of PI and mapped the keyboard ASCII character. To produce a melody, we took random notes (different from the ones in the PI array) and played them in parallel on the MIDI slider. We then turned on the “Polyphonic” option which made the patch sound good.

For the exact steps of conversion of the electronic notes into ASCII character and its implementation, we referred to the paper and video mentioned in our references.

Module BreakDown

We have also made the patch as modular as possible with comments on all important aspects.

ScreenShot of Methods

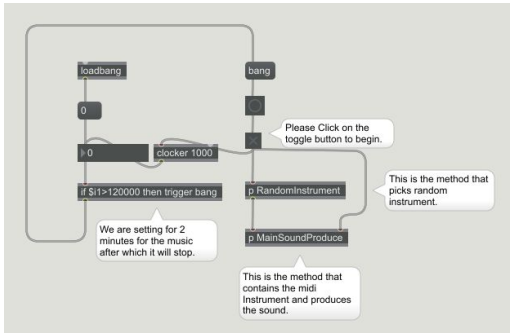

Figure 1 Main Method

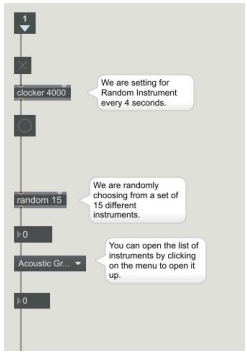

Figure 2 RandomInstrument Method

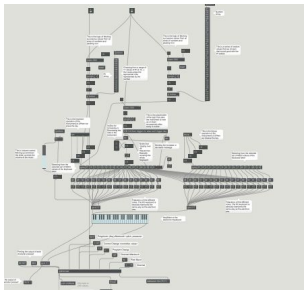

Figure 3 MainSoundProduce Method

Requirements

As for the minimum requirements, we made sure to meet them all.

1. *Run for at least two minutes:*

   Our machine runs for at-least 2 minutes before invoking a bang and getting out of the loop.

2. *Have both serial and parallel processes:*

   We actually have 3 major processes in the machine.

   First is the selection of the notes. The machine takes both the keyboard pressed and keyboard released button in parallel which causes the emanation of a polyphonic tone.

   Second is the scaling up and down of the notes across the keyboard. We have designed the MIDI keyboard the same way as the original electronic keyboard of a piano. We input a range of ASCII values into the keyboard which starts from A to H for the first octave.
   So basically, in an arrangement of 2 black upper keys and 3 white lower keys of an original piano, we have an “AWDRF” combination.

   Third is the random changing of different instruments. This changing of instruments also takes place in parallel with the two previous processes.

3. *Have both recorded and synthesized sounds:*

   As per discussion in the class, we need not have recorded sound in the machine and MIDI was enough.

4. *Rely on stochastic processes to a certain extent:*

   We have included 15 randomly chosen instruments to run our music, resulting in different melodies each time it is played.

Roles and Responsibility

Vinit was responsible for the inception of the idea, looking at different aspects of the music and its conversion into mathematical form and also building the project like the coding aspect and how to link each individual elements to each other, helping with the preparation of the initial and final draft, doing the patcher documentation and helping with the preparation of final video.

Christel was responsible for helping the team with her valuable inputs in the form of the music to be selected, helping with the documentation of the patcher, also preparing the initial draft and providing with the audio and video files and also proofreading the entire documentation.

Alexander was responsible for the preparation of initial draft, proofreading and helping with his valuable inputs in the music research.

References

For reference and inspiration we used the following paper:
http://www.nntdm.net/papers/nntdm-20/NNTDM-20-1-72-77.pdf
Additionally, we also referenced this video on the fibonacci series:
https://www.youtube.com/watch?v=2pbEarwdusc