Name: Frederik la Cour
Stud. Nr.: 201707873
Date: 03-04-2018
Characters: 7052/ 7200

DRAFT Synopsis

During this Easter holiday, I have been thinking a lot about what to write about for my final exam in software studies. It has been quite hard to figure out what part of our curriculum to choose a subject from, that said I did have some idea of what I wanted to work with. Some of the topics that I found to be more interesting than others are generativity and temporalities, so I know that I want to work with these subjects and I feel confident in that there is enough material in these two subjects to write those 12-15 pages for the exam. However, for quite a long time I didn't really know exactly what these subjects should be centred around. Just before the Easter break me and my study-group had a meetup where we discussed what each of us was going to write about or thought about for our finals. It was very inspiring to listen to what the others was planning to write about, and it was here I decided to try to look more into Live coding, with are more or less sonic angle to it, however this was still a relatively un-conclusion idea, in that I wanted to research more about the subject of performing code, and code in relation to musical composition and notation.

During the Easter break I have had some time to read and so I set out to do some inquiry work where I researched everything I could find in the software studies lexicon and other materials from our curriculum having to do with audio, music, composition, notation etc. A member of my study group recommended me part from "The philosophy of Software" called "Running Code" in which there is chapter called "Reverse remediation" which really intrigued me. In this chapter, we are introduced to a Japanese composer called Masahiro Miwa, who experiments with music that is composed through programming metaphors and frameworks. He has created a work called "Reverse-Simulation Music" which is explained throughout the text. In this work, he is experimenting with a certain form of algorithmic composition in which he sets up some relatively basic rules which are carried out by a group of musicians. Each musician has to follow the rules and doing so will result in a generative piece of music that over time will change and become more and more complex. In really think it could be very interesting to research more about Masahiro Miwa's compositions and explore them from a software studies perspective where I could draw upon certain topics in generativity and temporality. In fact temporality plays a huge part in Miwa's works since he through his experiments are able to show how running code works because normally the speed of running code is determined by the clock speed of the computer which often is insanely fast compared to what we humans are able to follow, thus making it very hard for us to comprehend how running code actually works, however through Miwa's reversed simulation we are then able to set the clock cycle to a more human perceivable framerate because the processes which are performed in his works are carried out by humans. This does however raise a lot of question in regard to the temporality of Miwa's work. How do we determine clock speed in a system which is not actually a system but a group of people?

I am very happy that my study group help me find this text, because this has inspired me to find more texts like it, and so I also found these two-other texts which are from the Software studies lexicon, "Sonic Algorithm – Steve Goodman", and "System event sounds – Morten Breinbjerg".

Name: Frederik la Cour
Stud. Nr.: 201707873
Date: 03-04-2018
Characters: 7052/ 7200

Which both seem to be like very relevant texts that uncovers more on the subject of algorithmic composition or notation. Steve Goodmans text is even more about this generativity in composition, than the reverse-remediation one. Here they also touch upon biological cellular data and this datafication of everything and how this is transposed into music by assigning pitch to values derived from biological data. So, they also touch briefly on the game of life program which is a highly mathematical generative program, which they then explain has been turned into a compository software called CAMUS. In general, I find that in a lot of these texts about audio and the contemporary music pieces and experiments a lot of the artists use software that I am quite familiar with or have briefly used to some extent, software like Max/msp, Pure Data and Supercollider are different forms of software used to create generative music combined with other multimedia functionality.

I have used Max/msp to create a live audio visualizer and performed with it on stage. I think that having the background that I have, and reading these texts which is something that I can relate to is also a big factor in writing an exam about this topic. I am very passionate about electronic music in general and I spend most of my time experimenting with compositions and audio designing, using everything from Max/msp and Ableton live to modular hardware synthesisers and MIDI controllers. Having this knowledge and passion for electronic music and the urge to learn more about it in a software studies context is something that will benefit the depth of the final exam paper.

In regard to what angle I will try to discuss in my paper I am of course still researching but so far, I think that the authorship debate could be interesting to bring up in the context of the generativity in the algorithmic composition, and maybe also question this whole concept of algorithmic music as music. Another thing that might be interesting would be to investigate further into the datafication and how it influences compositions, this would be in relation to using biological data and to transpose it to correspond to certain values. I haven't quite found a good angle from a temporalities point of view, but I am certainly working on it.

This brings me to the final question that I will answer in this synopsis, which is what will I do next? My plans are currently to read Morten Breinbjergs "system event sounds" and make as good annotations as I have done with the other texts I have mentioned in this synopsis, and furthermore read through my annotations of the Alex McClean text concerning live coding. I think that in general I need to go through my annotations on subjects like generativity and temporality and found interesting question that may lead to more interesting angles which I can project upon the subject that I am trying to define. I also think that maybe looking more into more different projects like Masahiro Miwa's reversed-simulation music, which are also created with Max/msp could be interesting and my also pose new question of give some interesting angles who knows? All things considered I am really not too worried about not exactly knowing what angles that I want to go at it from yet, however I think it is very important to keep finding new material and also try to use some of the material that I have already read previously. Also, I want to make this topic a lot more confined so that I don't disperse out in every direction when I start writing this paper.