Aesthetic Programming

Class 05- Code and Interactivity

*** Weekly mini ex3 – Design a throbber| discussion in class:**
https://github.com/AUAP/AP2017/blob/master/all_miniex/mini_ex3/mini_ex3.md

Discuss the way of learning:

- How you come up with the idea?
- How to test and experiment code?
- What frustrations you encounter and how you solve it?
- What are the tips for programming?

- My throbber is an animation based on a little fun
- When loading something on a computer, it often happens very quickly and the speed is what the fast-moving hands are supposed to symbolize.
- A throbber seems like a stupid little thing with no purpose from the outside but it serves an important role in the digital experience with computers and the internet on sites such as youtube, and it requires a lot of work to design and program. This knowledge opens up for thought about symbols that are used in digital artifacts and their significance.
- In the digital culture waiting is not appreciated and therefore the throbber should activate the user with something.
- We can use programming as a tool to understand how digital visual things can be created
- There will always be culture evident in programming. As such, the digital visual culture can ongoingly be understood, reflected on and evolved through understanding programming.
- the throbber represents a more significant shift in the way we consume digital content. […] We rely on more and more 'feeds' - streams of data, changing instantly and constantly.

Discussion / Sharing:

- Share your work with the one next to you
- How may we imagine the *now* that is entangled and loaded with computational logic?
- How would you modify your work such that it becomes a critical work in itself?

The mundane throbber calls for a critical attention towards mediated processes not only at a planetary scale, but also at the micro-temporal level of operations, including clock cycles, instructions execution, packet switching and data buffering, which exhibit micro-decisions and micro-interruptions.

Soon, 2017, p. 99

CODE AND INTERACTIVITY

Technique for reading code

→ Decoding the logics

- → Observe what you see from the interface
- → Try interact with the piece
- → What are the possible functions?

https://auap.github.io/AP2017/class05/sketch05/

Exercise:

1. Try to locate the corresponding functions
2. Can you spot on other additional functions in this program?

https://auap.github.io/AP2017/class05/sketch05/

- Loops
- Listening events
- Conditional statement: if/else

Loops

Discuss: Can you locate the for loop? What is a loop??

Loops

Discuss: Can you locate the for loop? What is a loop??

```
function footer(distance, footersize) {
  for (var i = 0; i < width; i+=distance) {
        fill (random(200));
        ellipse(i, height/1.05, footersize,footersize);
  }

}
```

Listening events

Discuss: where are the listening/interactive events? How do they operate?

Basic condition

If, Else, Else if:

In the physical world:

"If I am hungry then eat some food, otherwise if I am thirsty, drink some water, otherwise, take a nap"

→ Try to turn to programming:

```
If (I am hungry) {        //condition > boolean expression
        eat some food;  //instructional action
} else if (thirsty) {
        drink some water;
} else {
        take a nap;
}
```

Basic condition

Common language:

The class starts too early → true
The class starts too late → false

Formal logic of computer science → **relationships** between numbers

15 is greater than 20 → false
5 equals 5 → true

With variables (in programming practice)

X > 20 → depends on the current value of x
Y == 5 → depends on current value of y
Z <= → depends on current value of z

Conditionals

Relational Operators:

>	Greater than
<	less than
<=	less than or equal to
==	equality
>=	greater than or equal to
!=	inequality

Mini exercise4 walkthrough: feedback loop

Peer-tutoring: Group 3 / Respondents: Group 4
Topic: Conditions with sound and video

- How to incorporate sound and video in your sketch?
- Sample code
- Can you discuss your work with one or more aspects of Fazi and Fuller's understanding of computational aesthetics?