Documentation of my final project “somewhere”:

Components:

5 classes: (* means from Daniel Shieffman, ** means from ComputationalGeometry library)

- flowfield*
- tentacles
- vehicle*
- IsoContour**
- IsoSkeleton**

2 functions:

- generate_arc
- limb

Elements:

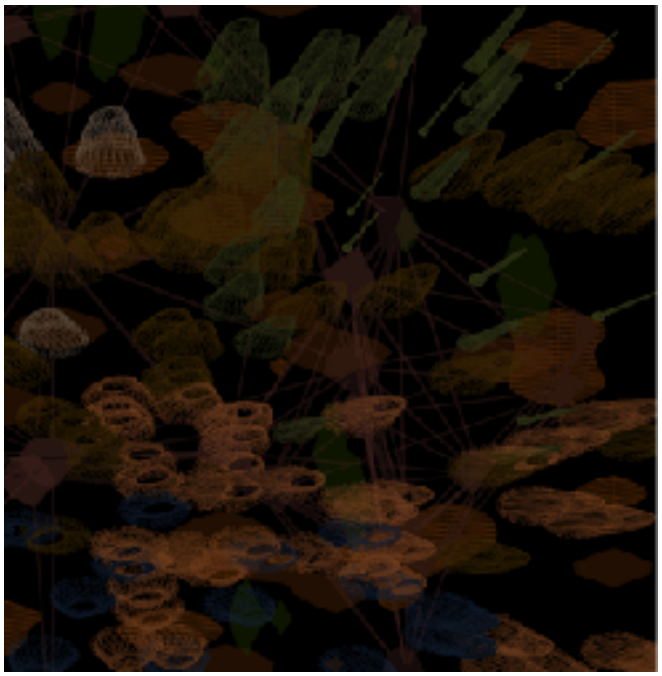
- tentacles of different colors grouped together, randomly distributed positions
- hidden web with mouse-dependent thickness
- flowfield with directions determined by perlin noise, each mouse press updates the field
- sea sperms driven by flowfield
- pond-like traces that expand and contract periodically (using isocontour)
- sea flowers with crazy limbs

Sounds:

- breath in: more reverberation when mouse lands at righter boundary
- breath out: more reverberation when mouse lands at lefter boundary
- processed noise: come and go periodically
- whispers: comes when sperms come, re-triggered when flow field is changed
- swallow: mouse-dependent swallow, get more reverb on the side.

Proud moments:

1. subtleties in layers and hidden moving parts.
2. intimate sounds, which can’t be shown through images.



Self-assessment:

On time management:

I didn't spend much time on learning syntax. Most of the code structures, including the ones on Generative Design come quite natural to me. However i don't think I've spent enough time coding either. I found myself staring at the available visuals I have for usually, very long periods of time, not knowing what to do next. These pauses are usually resolved by moments of revelation or looking at others' code. I don't like to randomly poke around unless I'm absolutely certain about what I'm trying to achieve. There are exceptions when I accidentally created a bug that incurs unusual effect (like in my final project each time i update the flow field all my tentacles start to go nuts, I have no idea how that happened but i kept it). Throughout my work process there will be a few "checkpoints" of big advancements, but along the lines connecting these checkpoints are usually unsatisfying twists. Although I am gradually learning to articulate on a given concept, there's still room. You can probably see from my code that each element is not that much different from its first version, except I do have a lot of elements.

On processing:

I think to some extent I've discovered my style on creating digital visual items. This would not have happened without processing. I also developed a working method - write a poem. create new features, articulate with variations, and arrange features onto the canvas.
There are still frustrations of course, I don't think I will be able to know how some effects are coded by looking at them. I want to create freer visual objects, but my means are contradicting that. Perhaps I need to learn shaders.

On struggles

Most of my time are spent on developing important features which I can build upon later. From the very first project I spent three days (probably) to make a simple generate_arc function work. Then i used that as the foundation of everything i created later on, including my monster, my tentacles, my crazy limbs, even tails of my sperms. It's been a right decision because there's so much you can do with arcs. However there are always that very few revelations where I'd be able to come up with some structure and surprise myself, then the rest of the time are spent blanking out. Thinking in terms of this one function is constraining somehow, as there are so many other tricks out there creating crazy effects (the flow field for example). And I can never think of them unless looking at their code. I have also spent some time looking at others' work on instagram, lots of which I still don't know how to achieve. I often view visual elements as individual structures built from scratch, but a lot of the times they aren't. They can be members of groups, determined by other forces, counterpart of something else, their curvy contours can be results of shaders rather than manually adjusted collection of coordinates, their colors can be pixelated etc. I need to understand more of how this canvas is rendered internally so that I can manipulate things easier.

On improvements

Look at others' code. Look at other languages (openframeworks for example) so that i can understand better what a certain language is good at doing, how the visuals are rendered, discover multiple pathways to achieve the same visual effects.

On class

I wish there will be myth-breaking sessions where students can present visual effects they love made by processing and together we come up with how it might be achieved through code, in the end the teacher exposes the real code and talk about it.
I like emotional visual objects, and creative use of sounds. Some interesting activities may include exploring different ways to express “absurd”,”awkward”,”ecstacy” through visual objects without being too literal about it. For sound it will be fun to create sound that represents random nouns like “elephant”, “body lotion”, ”a stop sign”, without using any sound made by that certain object.
I needed these occasions to break through the net of my usual imaginations.