Summary:

- I want to make a story told with purely-code generated elements like lines, text, shapes and color. I've done a fair amount of processing/p5.js narrative work with images, and I want to try my hand at a story using only abstract representations as characters
- it will be interactive, of course! I want to add interactivity to a lot of the elements presented, and depending on how you interact with these, you'll get a different ending to the story. I won't be telling a story in the traditional sense, more like an in abstract/philosophical-ish sense. it will be a sort of conversation between the program and the person interacting with it (sometimes with words, sometimes purely visually, or a combination of the two) and the program will have emotional responses to what the person does to it (example: if the program is displaying text slowly and the interactor decides to click to see if it'll load faster, the program will stop loading the text and display other text that says, "hey that was rude. I wasn't done talking")
- I will start with a simple amount of story branches (let's say 3) and work my way up from there
- I'll try and use a variety of elements from the get-go so it's not too boring to look at or too simplistic to code
- to not overdo it, I could try to aim for one complicated animated interactive element and a handful of more simple elements to go with it (like for example, narrative text gets shown as the viewer plays around with the more complicated animation)
- I do not have a really detailed plan, because this is the type of thing that I feel will emerge once I play around with code a bit

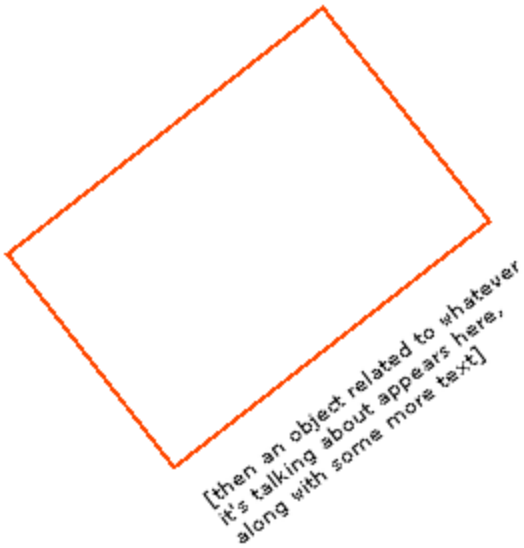
Media:

I don't have photoshop so shoddy drawings in Aseprite will have to do:

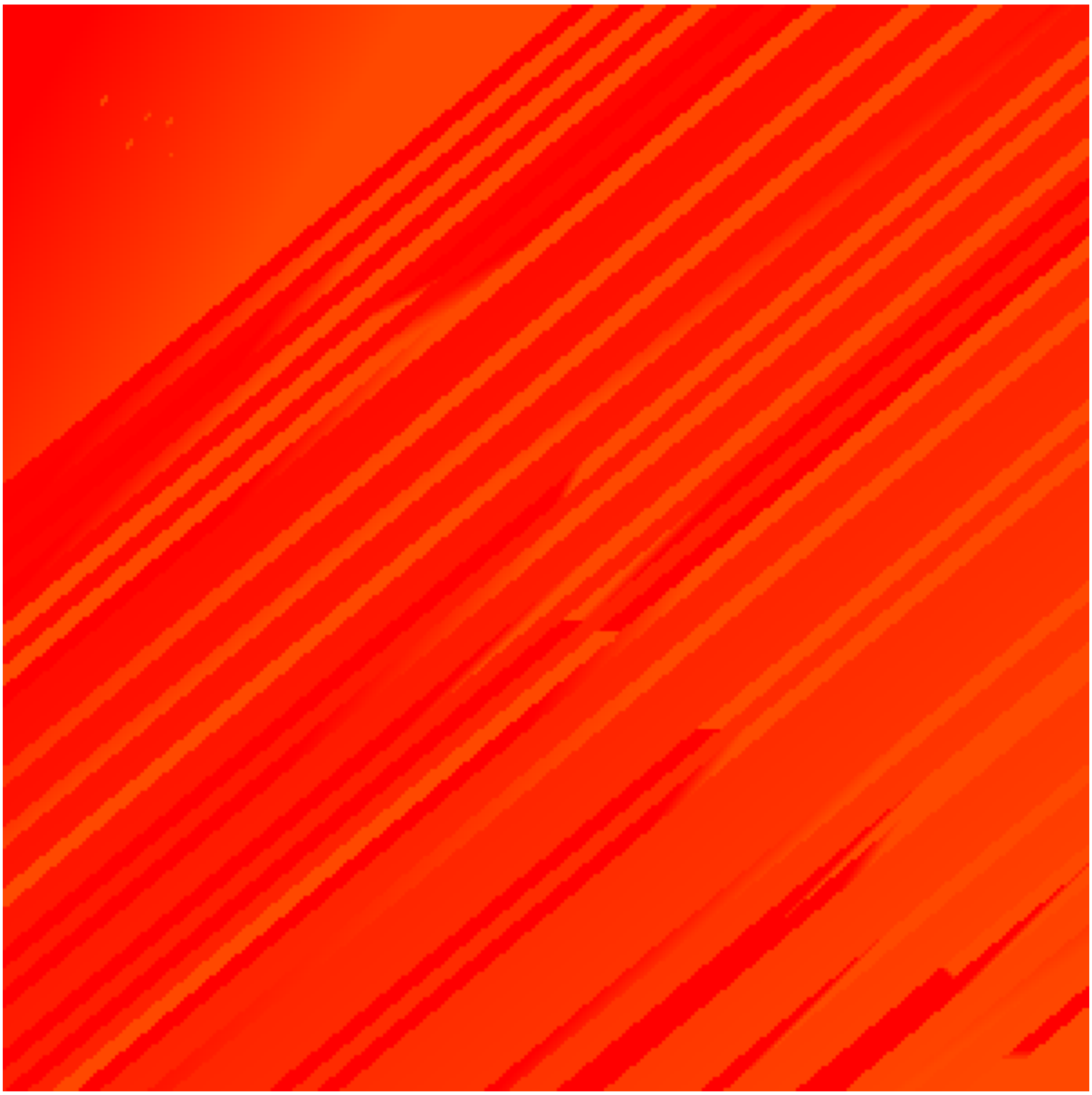
Example layout of a scene:

[the program says some things]

[then some more text starts to appear down here]



Example of a decorative pattern that could be generated with code to convey a feeling:



Inspirations:

http://www.creativecoding.org/examples

https://www.openprocessing.org/sketch/169537

https://www.openprocessing.org/sketch/153760

http://rhizome.org/art/artbase/artwork/form-art/

https://www.openprocessing.org/sketch/467484

https://www.openprocessing.org/sketch/377730

https://www.openprocessing.org/sketch/376645

Technical approach:

- to create the different story branches, I can probably separate them into different objects. Or maybe it would be better if the story branches were separated by methods instead?
- for the elements that aren't so complicated that they need their own class, I can organize them under a tab called "other elements"
- no images will be used at all, only code will be used for the visuals
- I plan to use several words and phrases, so maybe I can expect to use a lot of Strings. After all, I may want to have some text parts of a phrase act differently from the rest of it, so to do that I need different instances
- I expect I'll need to create arrays of object instances for the parts of the program that use many similar elements at once
- The types of input I was thinking of using was, primarily, mouse input. So I'll most likely need mouseClicked() or mousePressed() functions, and use mouseX and mouseY for the attributes of many elements
- To add more feeling to the story, I can import sound. Background music can be easily looped. I'll use an ambient free-to-use track from the yume nikki freebie pack. The pack also has some sound effects I can use.
  http://ask.plasterbrain.com/post/112074935411

Technical Research:

- I was thinking of using a trail effect for elements when but didn't know how to implement it. This sketch seems like a good reference (the comments may be in german, but the code is not very complicated so it shouldn't be a problem)
  http://www.creativecoding.org/example/processing:mausverfolger