Assignment #4

Getting Started with MATLAB/Octave

Try programming in the MATLAB language. You can access MATLAB environments in any of the following ways:

- In a workspace, type **`matlab`**
- In a workspace, type **`octave`**
- Launch the OCTAViEw tool: https://nanohub.org/tools/octaview

Once you've got any of these environments up and running, experiment with the "Spirograph" equation:

$$z(t) = e^{i 2\pi n_1 t} + e^{i 2\pi n_2 t} + e^{i 2\pi n_3 t}$$

where the variable *t* runs along the domain [0,1]. The variable *z(t)* is a complex number with some interesting characteristics. If you plot the real part of *z(t)* versus the imaginary part, you get different curves that look like Spirograph drawings, depending on the values that you pick for n_1, n_2, and n_3. If you want to learn more about Spirograph and the underlying mathematics, read this article: http://linuxgazette.net/133/luana.html

For our purposes, this simple formula can be expressed in the MATLAB language as follows:

```
n1 = 13;
n2 = -7;
n3 = -3;

z = exp(i*2*pi*n1*t) + exp(i*2*pi*n2*t) + exp(i*2*pi*n3*t);
plot(real(z),imag(z));
```

Add a line of code to define the variable t, which should have 1000 points equally spaced from 0 to 1. Then, try running this code in your MATLAB environment. Experiment with different values for n1, n2, and n3, and see the various plots that you get.

What happens if you don't include the semicolon at the end of each line?

Lessons Learned:

- MATLAB is a good language for expressing equations and making scientific calculations.
- MATLAB has built-in symbols like `i` and `pi` for mathematical concepts.
- Built-in plotting makes it easy to visualize what you compute.