Notes for Lab 10

Problem 1: `MyLogisticModel.m`

- Step 1: Set parameters.
  - The time step h is 0.01, and the initial and final time stamps are $t_0 = 0$ and $t_1 = 10$. To create equi-spaced grid points from t_0 to t_1, you can use:

    ```
    t=t0:h:t1
    ```
  - For the variable `t`, its i-th element `t(i)` means $t_0 + (i-1)h$.
  - Note: The function `linspace` also creates equi-spaced grid points; it is preferred when the number of points is given.
- Step 2: Initialize the solution.
  - Initialize a variable `y` that has the same size `t`.
  - For the variable `y`, its i-th element `y(i)` means $y(t_0 + (i-1)h)$.
  - Set `y(1)` to be the initial value.
- Step 3: Define the model.
  - Define a function to compute y'. There are two (out of many possible) ways:
    * Use function handle: `f=@(y) [BODY OF FUNCTION]`
    * Define an auxiliary function: `function yprime=f(y) [BODY OF FUNCTION] end`
- Step 4: Solve the ODE.
  - The forward Euler method applied to Problem 1 is

    $$y_i = y_{i-1} + hf(y_{i-1}), \tag{10.1}$$

    where y_i is an approximation of $y(t_i)$. Write a loop to compute Equation (10.1) iteratively. Figure out where the index starts and ends in the loop.
- Step 5: Plot the solution.
  - Plot y over t, i.e. x-axis is `t` and y-axis is `y`.

Problem 2: `MyLorenzAttractor.m`

- Step 1: Set parameters.
  - Similar to Problem 1.
- Step 2: Initialize the solution.
  - Initialize a variable `x` of size $3 \times n$, where n is the length of `t`.
  - For the variable `x`, its (i, j)-th element `x(i,j)` means $x_i(t_0+(j-1)h)$; its i-th row means x_i, and its j-th column `x(:,j)` means $\mathbf{x}(t_0 + (j-1)h)$.
  - Set `x(:,1)` to be the initial value.

- Step 3: Define the model.
  - Define a function to compute $\mathbf{x}'$. Let your function (note: function, singular form) takes a 3×1 vector and outputs a 3×1 vector. To define a 3×1 column vector, use the syntax:

    ```
    [ ELEMENT1; ELEMENT2; ELEMENT3 ]
    ```

    Without semi-colons, it's a 1×3 row vector.

- Step 4: Solve the ODE.
  - The forward Euler method applied to Problem 2 is

    $$\mathbf{x}_j = \mathbf{x}_{j-1} + h\mathbf{f}(\mathbf{x}_{j-1}), \tag{10.2}$$

    where $\mathbf{x}_j$ is an approximation of $\mathbf{x}(t_j)$. Write a loop to compute Equation (10.2) iteratively. Figure out where the index starts and ends in the loop.
  - Note that you are required to use the vectorized form.

- Step 5: Plot the solution.
  - Plot the trajectory in the space, i.e. x-axis is `x(1,:)`, y-axis is `x(2,:)`, and z-axis is `x(3,:)`. Use the function `plot3`; documentation is here: `https://www.mathworks.com/help/matlab/ref/plot3.html`
  - To set the view angle, use the function `view`; documentation is here: `https://www.mathworks.com/help/matlab/ref/view.html`.