Introduction to MATLAB
Fundamental Engineering Skills Workshop

General Tips and Reminders

- The most important command in MATLAB: `help function`
- To terminate a program that is currently running: `Ctrl-C`
- The command `whos` will display all variables that have been defined and their respective dimensions
- MATLAB indexing begins with 1, as opposed to 0 in C/C++
- In order to suppress printing to the Command Window, end the line with a semicolon
- The equal sign is an assignment operator, whereas the double equal sign is a relational operator. For example, `a = 3` will substitute all future instances of `a` with the value 3. In the conditional statement `if (a == 3)`, the value of `a` is compared to 3.
- The command `clc` clears the Command Window
- The command `clear all` removes all variables from memory
- The command `hold on` will retain plots in the open figure window
- Remember to comment your code for ease of understanding, both for yourself in the future and for other users

`for` Loop

- Use when you know the number of times a given calculation should be repeated
- Syntax:
  ```
  for i = 1:2:9
      …
  end
  ```
- In the above case, the first iteration of the `for` loop is for `i=1`, and the second iteration is for `i=3` because the step size is specified to be 2. The variable `i` continues to be incremented by 2 until `i=9`, at which point MATLAB completes one last iteration and then exits the `for` loop.

`while` Loop

- Use when you have a known constraint and require a calculation to be performed an unknown number of times
- Syntax:
  ```
  x = 1;
  while (x <= 5)
      …
      x = x + 1;
  end
  ```
- In the above case, `x` is initialized to a value of 1 so that the program will actually enter the `while` loop. Remember that the calculations performed in the `while` loop must alter the constrained variable, otherwise this will result in an infinite loop.

`if/elseif/else` Statement

- Use when you have a set of calculations to be performed only when a specific condition is met; these conditional statements allow the program to make decisions during its execution
- Syntax:

```
if (j == 1)
      …
elseif (j == 2)
      …
else
      …
end
```

- Note that specific relational comparisons must be stated for `if` and `elseif`, whereas `else` is an option for any remaining, yet not explicitly specified, condition.
- Only one of these options will be performed each time the set of conditional statements is encountered.