ENGR 1181 | MATLAB 10: For Loops 2

In-Class Guide

Learning Objectives
1. Use more complex ways of setting the loop index.
2. Construct nested for-end loops in the following situations:
   a. For use with two dimensional arrays
   b. For repeating a calculation numerous times
3. Use for-end loops to repeat a code with conditional statements.

Textbook Reading
Chapter 6.5 – 6.6

Topics
This lecture contains the following topics:
1. For-End and step size.
2. Running Summations
3. Loop Counter for Indexing
4. Conditionals with For-End Loops
5. Nested For-End Loops
6. Break Command
7. Continue Command

Outline
Below is an outline of the topics and the order in which they should be covered:

1. For-End and Step Size (Instructor’s In-class Activity)
   a. For-end loops can use a step size other than 1.
   b. The default step size is 1 if it’s not specified.
   c. For-end loops can also have a negative step size.

2. Running Summations (Instructor’s In-class Activity)
   a. A for-end loop can be used to calculate a summation of values.
   b. The values can be stored in variables that continue to add to themselves.
   c. The sum variable must be initialized to zero before the loop.
   d. Remember to avoid using MATLAB function names as variable names.

3. Loop Counter for Indexing (Instructor’s In-class Activity)
   a. A loop index can be used for obtaining values from a vector.
   b. Variables calculated within a for-end loop that don’t reference the loop index will get overwritten with each run of the loop. Use array assessing concepts and a loop index to save all the calculated values to a vector.
4. **Conditionals with For-End Loops (Instructor’s In-class Activity)**
   a. Conditional statements can be included (nested) inside of for-end loops.
   b. This is useful in many applications such as sorting data or displaying statements to the screen.
   c. There is no limit to the number of conditional statements that can be nested.

5. **Nested For-End Loops (Instructor’s In-class Activity)**
   a. For-End loops can be nested within other for-end loops.
   b. Nested for-end loops are useful when creating and/or analyzing matrices. The loop index variable for the first loop addresses the row, and the loop index variable of the inner loop addresses each column.
   c. There is no limit to the number of loops that can be nested.

6. **Break Command (Instructor’s In-class Activity)**
   a. The `break` command will terminate a current loop.
   b. MATLAB will jump to the end and carry on with the program.
   c. It is often used with conditional statements (e.g. if a certain condition is satisfied, the loop can be terminated).

7. **Continue Command (Instructor’s In-class Activity)**
   a. The `continue` command will stop the remainder of the current loop pass.
   b. The program continues with the next iteration of the loop.
   c. It is often used with conditional statements (e.g. if a certain condition is satisfied, the pass can be stopped).

8. **Students will complete Part 1 of the assignment to turn in by the end of class. Once finished, students may start on Part 2 due by the next class. Students will submit Part 1 of the assignment to the Carmen dropbox as a PDF.**