

MATLAB Marina: Arrays 1d

1. From the MATLAB Command Window enter and execute the MATLAB statements in Figure 1 paying attention to the variables and their values in the MATLAB Workspace window.

```
clear; clc;
x1 = [13, -2, 0, 42, 87, 10];
x2 = 1 : 1 : 10;
x3 = 10.5 : 0.5 : 20;
x4 = [x2, x3];
lx2 = length(x2);
lx3 = length(x3);
[nrx4, ncx4] = size(x4);
```

Figure 1. MATLAB Statements for Exercise 1

2. From the MATLAB Command Window, execute the MATLAB statements in Figure 2 paying attention to the variables and their values in the MATLAB Workspace window. What does the logic operation $x < 0$ return as a result? What does the function call `find(x < 0)` return as a result?

```
clear; clc;
x = [-5, 13, 8, 72, -4, 0, -1, 16, 23, 7, -8];
loc = x < 0;
ind = find(x < 0);
x(ind) = 0;
```

Figure 2. MATLAB Statements for Exercise 2

3. From the MATLAB Command Window, create the following 1D arrays and assign them the specified initial values:
 - a) A row vector named `v1` containing the values 0.0, 2.0, 5.0, 7.5, -5.0, and 20.
 - b) A column vector named `v2` containing the values 1, -7, 3, 8, and 6.
 - c) A row vector named `v3` containing the values from 0 to 99 with an increment of one.
 - d) A row vector named `v4` containing the values from 0 to 100 with an increment of two.
 - e) A row vector named `v5` containing 200 values linearly spaced from 0.0 to 5.0. Hint: use the built in MATLAB function `linspace`.
 - f) A row vector named `z25` containing 25 zeros. Use the MATLAB function `zeros`.
 - g) A column vector named `o40` containing 40 ones. Use the MATLAB function `ones`.
 - h) A row vector named `fives` containing 40 fives. This can be done by creating an array of ones and then scaling the array.
4. Enter MATLAB statements in the Command Window to perform the following operations:
 - a) Create an array named `w` with the values 0.0, 2.0, 5.0, 7.5, -5.0, and 20.
 - b) Index the fourth element of `w` and save the result in the variable `w4`.
 - c) Index elements two through five of `w` and save the result in the variable `w2to5`.

- d) Scale all the elements of the array `w` by five and save the result in the variable `wTimes5`.
 - e) Add one to all the elements of the array `w` and save the result in the variable `wPlus1`.
 - f) Square all the elements in the array `w` and save the result back in the variable `w`. You will need to either use the element power operation (`.^`) or element by element multiplication (`.*`) to multiply each element in the vector by itself.
 - g) Take the square root of all the elements in the array `w` and save the result back in the variable `w`. MATLAB has a function `sqrt` for the square root.
5. Write a MATLAB program that:
- a) Creates an array named `ww` with the values 78, 65, -5, 53, 88, -1, 90, and 0.
 - b) Determines the indices of the elements in `ww` that have values less than zero.
 - c) Replaces the values of the elements that are less than zero in `ww` with value zero. The other elements of `ww` should be left unchanged.
 - d) Determine the indices of the elements in `ww` that have values greater than or equal to 70 and less than 90. You will need to use a compound comparison operation, i.e. `(test1 & (test2))`.
6. Write a MATLAB program that will evaluate and plot the function $f(x) = x^3 - 5x^2 + 3x + 2$ over the range $x = -5$ to $x = 5$. Make sure to use an appropriate interval between points and to use array (element by element) arithmetic operations where appropriate. Use the comment skeleton of Figure 3 as your starting point and remember to start programs with the `clear`, `clc`, and `close` all commands. To plot a vector `f` versus a vector `x` (`x` on x-axis and `f` on y-axis) use the MATLAB command `plot(x, f)`. The MATLAB function `figure` creates a new figure window for a plot and the MATLAB functions `xlabel`, `ylabel`, and `title` can be used to annotate plots.

```
% create the vector x of independent variable values
% evaluate f(x) for all the values in x
% plot f(x) versus x
```

Figure 3. Comment Skeleton for Plotting Program

- 7. Modify the program written for problem 6 so that the program also generates a second plot of the function over the subrange $x = -2$ to $x = 2$. Do not evaluate the formula twice for two different time ranges. Instead, evaluate the formula once for the longer time range and then use an array comparison and indexing to extract the time and formula values needed for the second plot.
- 8. Write MATLAB program to create an array of all the integer powers of 2 starting with 2^0 and ending with 2^{15} , i.e. create the array `[1, 2, 4, 8, 16, ..., 32768]`. Use MATLAB element by

element operations to create the array; do not just type in the values. Use the comment skeleton of Figure 4 as your starting point.

```
% create an array of the powers from 0 to 15  
  
% raise 2 to the power of each element in the array
```

Figure 4. Comment Skeleton for Creating Array of Powers of 2

9. Write a MATLAB program to determine an N term approximation of the infinite series $g(x) = 1 - \frac{1}{x} + \frac{1}{x^2} - \frac{1}{x^3} + \dots$. The program should read in the value of x and N from the user and display the approximation of the infinite series. The general term in the series (mth term) is $\frac{(-1)^{m-1}}{x^{m-1}}$ assuming the terms start with term 1. The MATLAB functions `input` and `fprintf` can be used for user input and command window display. Use the comment skeleton of Figure 5 as a starting point.

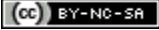
```
% read in x and N  
  
% create array of term numbers from 1 to N  
  
% evaluate the general term function for each term number  
  
% sum all of the terms
```

Figure 5. Comment Skeleton for Program Approximating a Series

10. Write a MATLAB program that will determine the minimum, maximum, and mean (average) score of a set of exam scores. Use the array `scores = [78, 57, 82, 91, 81, 84, 73, 68, 24, 88]`, to test your program. MATLAB has built in `sum`, `mean`, `min`, and `max` functions for arrays.
11. Modify your program from problem 10, so that the average is computed ignoring the lowest and highest scores, i.e. throw out the low and high score and then compute the average of the remaining scores. Test the modified program using the same `scores` array of problem 10.
12. Write a MATLAB program that will analyze a set of measurement data. Some of the measurements may be faulty and should not be included in the analysis. Faulty measurements are those less than 0.0 or greater than 5.0. The program should:
- Determine which measurements are valid (greater than or equal to 0.0 and less than or equal to 5.0). Having this result be an array of indices will make part b) easier.
 - Determine how many of the measurements are faulty.
 - Determine the percentage of measurements that are faulty.
 - Determine the minimum and maximum valid measurement.

Test the program using the measurement array `meas = [1.2, 2.6, -0.6, 0.0, 4.2, 3.2, 3.0, 2.0, 5.4, 0.8];`

Last modified Friday, September 18, 2020

 [MATLAB Marina](#) is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License.