

ENGG1003 - PASS Session 9

Mitchell Deltoer

Monday	14-15	ES238
Wednesday	12-13	ES238
Thursday	10-11	MCLG42

MATLAB Basics

Command/Operation	Description
help <function>	displays the help for the function or keyword specified
doc <function>	opens the reference page for the function or keyword specified
clear	removes all variables from the <i>workspace</i>
clc	clears the <i>command window</i>

1. Perform the following operations in Matlab and interpret the results/error message.

- | | | |
|---------------------------------|-------------------------------------|--------------------------------------|
| (a) <code>1:1:10</code> | (b) <code>4:0.5:9</code> | (c) <code>1:10</code> |
| (d) <code>[1,2,3; 1,2]</code> | (e) <code>[3:-1:1,12:16]</code> | (f) <code>[4:7; 9:12]</code> |
| (g) <code>[2:0.5:4; 1:4]</code> | (h) <code>linspace(1,20,500)</code> | (i) <code>linspace(15,10,1e3)</code> |

2. Create the following vectors/matrices in Matlab.

- | | |
|---|--|
| (a) $a = \begin{bmatrix} 1 & 2 \end{bmatrix}$ | (b) $b = \begin{bmatrix} 3 & 4 \end{bmatrix}$ |
| (c) $c = \begin{bmatrix} 2 \\ 5 \end{bmatrix}$ | (d) $d = \begin{bmatrix} 1 \\ -6 \end{bmatrix}$ |
| (e) $e = \begin{bmatrix} 3/5 & 4 \\ 6 & -2 \end{bmatrix}$ | (f) $f = \begin{bmatrix} -5 & 0.4 \\ 7 & -2.6 \end{bmatrix}$ |

3. Using the array definitions from question 2, perform the following operations in Matlab and interpret the results/error message.

- | | | |
|-----------------------|----------------|---------------------|
| (a) $A = a+b$ | (b) $B = c-d$ | (c) $C = e.*f$ |
| (d) $D = d./c$ | (e) $E = 6*a$ | (f) $F = a+[1,2,3]$ |
| (g) $G = 2*b+3$ | (h) $H = e/2$ | (i) $I = [a;b]+e$ |
| (j) $J = 2*[c \ d]-f$ | (k) $K = a*b$ | (l) $L = a.*b$ |
| (m) $M = b^2$ | (n) $N = b.^2$ | |

Array Indexing

4. Create the following vector in Matlab.

$$Y = [11 \ 12 \ 13 \ 14 \ 15 \ 16 \ 17 \ 18 \ 19 \ 20]$$

5. Perform the following commands in Matlab and interpret the results.

- | | | |
|--------------------|------------------------|--------------------------|
| (a) $A = Y(1)$ | (b) $B = Y(6)$ | (c) $C = Y([2 \ 4 \ 9])$ |
| (d) $D = Y(5:end)$ | (e) $E = X([3:7 \ 9])$ | (f) $F = X([1:3 \ 5:9])$ |

6. Create the following matrix in Matlab.

$$X = \begin{bmatrix} 1 & 2 & 3 & 4 \\ 5 & 6 & 7 & 8 \\ 9 & 10 & 11 & 12 \\ 13 & 14 & 15 & 16 \end{bmatrix}$$

7. Perform the following commands in Matlab and interpret the results.

- | | | |
|----------------------------------|----------------------------------|--------------------------------|
| (a) <code>A = X(:, 1)</code> | (b) <code>B = X([1 3], :)</code> | (c) <code>C = X(1:3, 4)</code> |
| (d) <code>D = X(3, 3:4)</code> | (e) <code>E = X(2:4, 2:4)</code> | (f) <code>F = X(1)</code> |
| (g) <code>G = X(3)</code> | (h) <code>G = X(10)</code> | (i) <code>H = X(1, 4)</code> |
| (j) <code>I = X(4, [2 4])</code> | (k) <code>K = X(:, :)</code> | (l) <code>L = X(:)</code> |
| (m) <code>M = X(1, end)</code> | (n) <code>N = X(end, end)</code> | |

Plotting

Hint: The help and doc commands are your friends when encountering unknown functions.

8. (a) Create a Matlab *script* file called 'passWk10-plot.m'.
- (b) Using the `plot` command, write a Matlab script that:
- Plots a single vector `Y = [2, 5, 3, 2, 1]`.
 - Plots the vector `Y` on y-axis against the vector `X = 0:0.5:2` on the x-axis.
- (c) Using the relevant commands, add a **title** (`title`), axis **labels** (`xlabel`, `ylabel`) and a **grid** (`grid`) to the previous figure.
- (d) Using the `axis` command, write Matlab code that will change the axis scale to $-5 \leq x \leq 5$ and $0 \leq y \leq 10$.
- (e) Using the `hold` command, write Matlab code that will plot the vectors `X` vs `Y` and the vectors `A` vs `B` on the same plot. Additionally, make one of the plots a **dashed red line** and the other a **solid green line** to distinguish them.
9. In a **new figure** using the `subplot` command, write Matlab code that will create a 2x1 matrix of plots, with `X` vs `Y` on the first plot and `A` vs `B` on the second plot.

for loops

10. (a) Create a Matlab *script* file called 'passWk10-loop.m'.
- (b) Using a `for` loop, write Matlab script that outputs each element of a 1D vector to the command window. You may use the following template.

```

1 x = 1:10; % defining the vector, you can change this
2
3 for % TODO: initialise the loop
4     % TODO: complete the task
5 end

```