COM322 LAB 1: MATLAB INTRO

Perform the following exercises in the MATLAB command window. Refer to the MATLAB course notes and MATLAB help for syntax and conventions. Remember that you can use up/down arrow keys to cycle through the command history. Copy and save the commands that correspond to these exercises either in the MATLAB editor or a text editor – you will need to run them to get checks.

1. Assign the following values for matrix **A**:

$$A = \begin{bmatrix} 3 & 6 & 1 & 9 \\ 6 & 5 & 3 & 7 \\ 7 & 3 & 2 & 6 \end{bmatrix}$$

a) Display elements **A**(1,2), **A**(3,2) and **A**(2,4).
b) Display the first row of matrix **A**.
c) Display the second column of matrix **A**.
d) Display the submatrix consisting of rows 2 & 3 and columns 3 & 4

2. Find the results of the following operations:
a) Multiplication of a matrix with a scalar

$$\begin{bmatrix} 5 & 7 \\ 2 & 9 \end{bmatrix} \times 4.5$$

b) Matrix addition

$$\begin{bmatrix} 2 & 6 & 4 \\ 9 & 7 & 2 \\ 3 & 4 & 8 \end{bmatrix} + \begin{bmatrix} 3 & 7 & 4 \\ 12 & 3 & 5 \\ 1 & 11 & 4 \end{bmatrix}$$

c) Elementwise vector multiplication

$$\begin{bmatrix} 4 \\ 6 \\ 2 \\ 1 \\ 8 \end{bmatrix} \circ \begin{bmatrix} 3 \\ 7 \\ 4 \\ 5 \\ 9 \end{bmatrix}$$

d) Matrix multiplication (note that **A**' is the transpose of matrix **A**)

$$\begin{bmatrix} 2 & 6 & 1 & 3 \end{bmatrix} \begin{bmatrix} 3 \\ 8 \\ 2 \\ 5 \end{bmatrix}$$

e) Matrix multiplication

$$\begin{bmatrix} 3 \\ 8 \\ 2 \\ 5 \end{bmatrix} \begin{bmatrix} 2 & 6 & 1 & 3 \end{bmatrix}$$

3. Construct a 5 x 10 array that contains zeros. Assign elements at (row, column) (3,5), (2,4) and (1, 9) values of 44. Display the matrix.

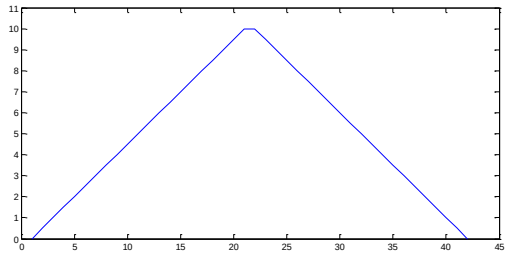
4. a) Assign variable **c** a value of 200. Cast to type **uint8**, display the result.
b) Assign variable **c** a value of 200. Cast to type **int8**, display the result. Explain the difference and repeat both parts for **c**= -150.

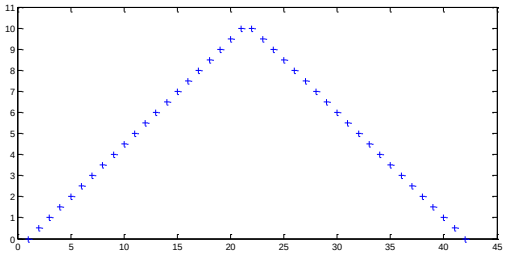
5. a) Create a column vector called **vect1** that contains even numbers in the range 100 to 120.
b) Create a row vector called **vect2** that contains numbers in the range 0 to 1 in increments of 0.1.
c) Multiply **vect2** by 100, assign the result back to **vect2** and display the result.
d) Display the lengths of both vectors and make sure they are equal (note their orientations). Then display the elementwise sum of the two vectors (first element in **vect1** is added to first element of **vect2,** second element to the second etc.) in the form of a row vector.

6. Without using a for loop, find the sum of integers divisible by 7 between 161 and 224. Use MATLAB Help to get information on the summation function called **sum**. Before using the function, test it with a few numbers to understand how it works.

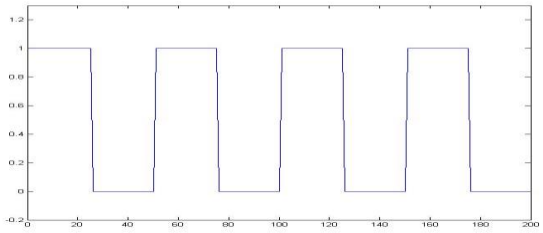
7. a) Assign variable **d** a value of 5. Write a logical expression (single line of code) to check if the value of **d** is in the range '1 to 3' or '6 to 9' inclusive. Assign different values to **d** and test your code.
b) Assign variable **d** a vector that contains 20 random integers in the range 1 to 10 (either use 'rand' and 'floor' or look up 'randi' from Help). Create another vector **e** of the same size with a different set of random integers in the same range. Check to see the matching elements using the command d = = e and display how many corresponding elements are equal using sum(d = = e).

8. a) Create a sequence of 21 numbers increasing linearly from 0 to 10 followed by a sequence of 21 numbers decreasing linearly from 10 to 0 (hint: use concatenation of two sequences). Assign this sequence to a variable and plot it with lines joining the points (left). Repeat with points and no lines (right).



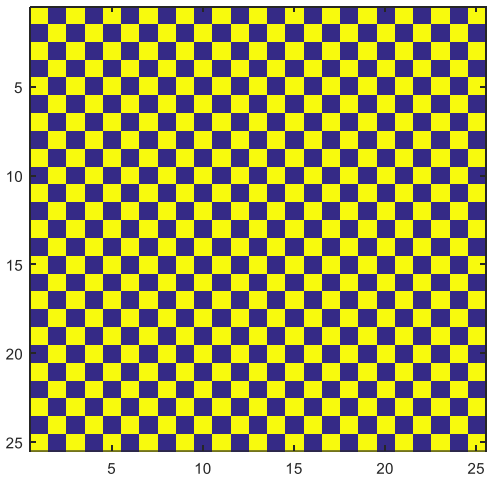


b) Create a square wave (value alternating between 0 and 1) with 4 periods over 200 points and assign it to a variable (you may concatenate ones and zeros to form the square wave). Plot the content of the variable. You may use this command to scale y axis of the plot as seen below axis([0, Inf, -.3, 1.3]);



9. Assign **A**=[1 0; 0 1]. Display it using imagesc(**A**). Repeat for **A**=[3 5 2; 8 6 1]. Repeat for **A**=[0:0.01:1; 1:-0.01:0]. Note the auto-scaling and the meaning of colors.

10. a) Create a vector **B** that contains 15 zeros. Assign a few selected elements the value of 1
e.g. **B**([1 3 6 7]) = 1 OR **B**(1:5) = 1.
b) Create a 25x25 matrix and fill it with a checkerboard pattern using 0s and 1s (without using MATLAB's checkerboard function). Display the matrix using 'imagesc'. Note that it is possible to do this with assignments similar to part (a) (without using loops) – you will need multiple lines of code. Repeat for 40x40 and 50x50.


5
10
15
20
25
5
10
15
20
25