ENGR 130 Final Summary Session

Review of Material:

1. In groups, create an outline of the topics that we have covered so far this semester.

   For each topic, list at least two bullet points on the key things to remember, syntax, equations, etc. for that topic.

   a. Example: polyfit
   b. Inputs: x vector, y vector, degree of polynomial fit
   c. Outputs: Coefficients for a polynomial
   d. p = polyfit(x,y,n)
   e. Necessary for creating best fit lines

		Notes: Inputs? Outputs? Purpose?
Basics	sum	vector input → sum of vector matrix input → Sums of columns in vector form
	sqrt	takes square root of all inputs
	length	returns length of longest side of a vector or matrix input
	linspace	linspace(min val, max val, number of vals) creates vectors of specified length
	\n	New line command in input(' ') and fprintf statements
	.	Specifies element by element operations in matrix opprts. Needed for *, /, and ^
	'	Dot isn't needed, but it doesn't hurt to include it in this case Transpose function. [1 2 3].' = $\begin{bmatrix}1\\2\\3\end{bmatrix}$ row ↔ column
	:	This can go backwards if step is neg Second Bound could be excluded sometimes Creates vectors with step size btwn. min and max. min:step:max
	;	For vectors/matrices, they must be the same size Suppress output to command window Move next entry to the next row of vector
	%, %%	Comment and section breaks, respectively
	' '	Character specifier
	" " {curly} to index strings	String specifier, creates a variable of characters or values strung together
If Statements	if	Start conditional statement
	elseif	Add under 'if' statement to create second condition
	else	'Catch all' statement following conditions. Will run if no other conditions were met
Input / Output	input	Prompt user for input: integer or floating point by default, follow with 's' for strings
	disp	Display one variable or character/string input to command window. Variables not in quotes.
	fprintf	Print text to command window. Include variables by using '%s', '%f', and '%i' for strings, float, and int respectively. \n = new line, \t = tab
Loops	for	Loop that runs a specified number of times. count variable used to dictate this For loop running 4 times: for count = 1:4

Not needed for scalar operations or addition/subtraction

If step is not specified, MATLAB assumes step = 1.

(x, y, z)

Variables specified after quoted message, in specified inside.

	while	Loop that runs while condition is not false (=0). Condition can be single variable or logical comparison with and/or statement
Plotting	plot	plot(x,y,'format') '--g' dashed yellow 'sc' square cyan Format specifiers: 'b' dotted blue line 'ok' black '-r' normal red '.m' dotted purple points
	xlabel, ylabel	Adds label to x and y axis, respectively. Must be right under plot command. Must be in quotes for string labels. Include on any plot.
	xlim, ylim	Sets hard limits on the plot. Turns off MATLAB's auto scaling. Must use vector in the format [min, max] to work properly.
	legend	Adds legend to graph, labeling lines in the order they were plotted in. Must be in quotes for string or character labeling.
	text	text(x,y,'label') adds text to a specified point on a graph. 'label' is a character vector
	title	Adds title to most recent plot. Include on any plot. Do not just put 'y vs. x', make it specific. Must be in quotes for strings or characters
	figure	Specifies figure to plot on. useful for creating seperate plots. figure() makes MATLAB just create a new figure with next unused integer.
	hold on, hold off	Causes all subsequent plot commands to be applied to the most recent plot. Can be placed anywhere, just be sure to not plot on the wrong figure.
Logical Operations Returns 1's and 0's only	==	Compares to values, returns logical 1 if equal and logical 0 if not.
	~=	Opposite comparison to ==, not equal returns 1 if comparison shows unequal sides.
	<, >	less than and greater than, respectively
	<=, >=	less than or equal to, greater than or equal to respectively. Place = on right side always.
	& , &&	'And' comparison. Returns 1 if both sides are true, and 0 if either side is false.
	| , ||	'Or' comparison. Returns 1 if either side is true, and 0 if neither side is true.
Files: .csv, .txt, .mat, .m	load	Loads data file from folder. Must all be the same data type. Works best with .mat files as they can contain any variable/workspace. load("file.mat","var1")
	save	Saves specified variables or the entire workspace to a file, typically .mat. To use non-.mat files must be same variable type. save("file.mat","var1","var2")
	writematrix	Create file with given specifier. Used often with .csv files writematrix(var,"filename"). Only saves one variable, notice quotes to file, in quotes.
	readmatrix	Opens file, typically .csv. Must be read to a variable Sometimes loads unneeded data by default. Must use MATLAB to edit out a=readmatrix("b.csv")
	fprintf	Writes data to an open file. Suppress output. Looks similar to printing to command window. Use \t to create columns in file, \n for rows. Use same specifiers for variables i.e. '%f','%s', and '%i' fprintf(file_ID,'I made $%f today', 1.74);
	fopen	Open file to a file_ID. Use 'r' for reading and 'w' for writing. file_ID is used to reference data in script file_ID = fopen('engine.txt','r')
	fclose	Closes an opened file. Always put in your code when done working with a file. fclose(file_ID)
	fscanf	Reads specified data from a file to variable. Store strings or chars with {curly braces}, and values with (parentheses). Must specify string, float, or int
Statistics	mean	if v is a vector, mean(v) returns scalar average. if m is a matrix, mean(m) returns vector containing mean values of m's columns.
	min, max	[M,I] = min/max(x) returns M value and I index of that value. If multiple occurrences exist, returns the first occurrence.
Matrix Operations	length	Returns length of the largest side of the matrix use size for dimensions [rows,cols] = size(x) Use numel for number of elements
Random Numbers	rand	Returns floating point values from 0 to 1. rand([m,n]) returns an m by n matrix For a range of float randoms: Var = min + rand([m,n])*(max-min)

Use lines for lines or contain data, scatter plot format for experim. or tests

will apply to most recent plot

figure(n), where n is a positive integer

hold off turns off the plot addition

Returns logical array when comparing vectors (must be equal size).

Useful for logical indexing.
a = 1:6;
b = 1:2:12;
c = b(b > a)
-> 3 5 7 9 11

Only use double signs. || and && No reason to use single symbol!

Must be in quotes vars come after the name. No vars specified will load or save entire workspace

not on test

Use specifiers mentioned previously (i.e. '%s'%f and %i).
fscanf(fileID,'%s', 1)

If x is a matrix, returns value for each column in a vector
ms = min(x)

	randi	Generates random integers in specified range. randi([min,max],[m,n]) generates a m by n matrix with random integers min ≤ n ≤ max.
	rng	Seeds random number generation. Often we use rng('shuffle') to seed generation based on current time. Prevents predictable randomness.
Linear Regression	polyfit	Generates polynomial coef based on data. Needed for best fit lines. Specify degree of poly needed poly_coef = polyfit(x_vec, y_vec, degree)
	polyval	evaluate a data used fit coef. returns fitted y_data that corresponds to inputted x vector fitted_data = polyval(fit_coef x_vec)
Strings white space = space, tab, or new line	strlength	Returns the number of characters in a string. Characters can be defined as any keyboard press, such as letters, numbers, spaces, or exclamations.
	strcat	Concatenate. Joins strings or characters. Spaces included with strings, but dropped when using characters
	strtrim	Removes leading or trailing blank characters from a string or character array.
	strcmp	Compares two strings and returns logical operator if identical or not. The entire array must be completely identical to be true. Can compare character and strings
	split	Divides a string at any white space characters. Result is string array with each word belonging to a new row. spaces, words removed, leading/trailing zeros included
Other	function Functions can be their own file, or can be in a script. If the latter, include AFTER all other code in script!	Creates a user made function. Start each function with block comments (i.e. purpose, all, inputs, outputs). function [out1, out2...] = my_func(in1, in2...)
	end	Denotes the end of a loop, conditional, or function.

MATLAB's random is pseudo random, use rng to help true

Degree of n yields coef vector of length n+1

Coef vector of n corresponds to degree of n-1

Avoid using identical variable names in function to those in script.

Suggestions for Further Study

Complete/redo the exam practice packets.

Use Matlab to check your practice responses.

Practice writing out code on paper (keep an eye consistency and casing)

Reminders

- Go to Open Lab to get your last minute questions answered:

Remaining Open Labs	**Monday, Dec. 18**	**2:00 - 4:00 pm**	**Rucha**
	Tuesday, Dec. 19	**1:00 - 2:00 pm**	**Michelle**

- Sleep
- Fill out the SI survey