Electrical and Electronics Engineering Institute
University of the Philippines, Diliman
Quezon City, 1101 Philippines

EE 214 Module 1
Lab Exercise 1: MATLAB Familiarization

General Instructions when submitting machine problems and lab exercises

1. Answers to the questions and additional discussion must be in the form of a written report and submitted in PDF format.
2. Your MATLAB scripts should be executable from the command line with no additional user inputs.
3. Include comments in your code to make your code more understandable
4. Use variable names that are meaningful to make your code more readable
5. Submit all the necessary files (pdf document, m-files, output images) as a zipped file using the following naming convention: **surname_nickname_EE214-LabXX_EE214.zip**
6. Submit the zip file through the UVLe submission bin

I. Analog to Digital Conversion

This part of the lab exercise is meant to familiarize students with the use of MATLAB. For first-time users of MATLAB, we suggest you browse through and try out some exercises in the Introduction to MATLAB by David Houcque of Northwestern University, which is also uploaded in UVLe.

Note that signals/variables manipulated in MATLAB (or in your PC) are already in its digital form. That is, they are sampled and quantized data. They are stored in memory as a series of 1's and 0's. Let us try to mimic an Analog-to-Digital conversion (ADC) process as an exercise in variable manipulation and plotting.

a.) Let us start by defining our "analog" signal. Note that the signal is still digital but with fine intervals so that it seems like an analog signal. To plot the signal, we use the **plot** command.

```
% clear all variables and close all windows
clear all;
close all;
% define "analog" time and signal
t = 0:0.001:1;    % time in sec
freq = 5;        % frequancy = 5 Hz
analog = 2*sin(2*pi*freq*t);
% plot the analog signal
plot(t,analog)
xlabel('time, t (sec)')
```

Discussion: Take a snapshot of the figure. How many sinusoidal cycles are plotted over the 1-sec timeframe? Is this consistent with what you expect? Explain.

b.) Analog-to-Digital Conversion follows 2 main steps: Sampling and Quantization. Sampling, as the name implies, takes samples of the analog signal at equal time intervals, typically defined by the sampling frequency, Fs. Quantization, on the other hand rounds the sampled value to the closest discrete quantization level. Some texts would indicate 3 steps for ADCs, the third step being Encoding. Encoding simply assigns values (possibly a string of binary values) for each quantization level. Let us now complete the ADC process in MATLAB.

```
% sampling
Fs = 25;       % sampling frequency = 25 Hz
n = 0:1/Fs:1;   % sampling intervals
sampled = 2*sin(2*pi*freq*n);
% plot the samples on the same figure
hold;
stem(n,sampled);
% quantization
% define 5 quantization levels as postiive and negative integers
quantized = round(sampled);
stairs(n,quantized)
```

Discussion: Take a snapshot of the figure. How many samples are there per sinusoidal cycle? Is this consistent with what you expect? Explain.

c.) Now we calculate for the quantization error. Error is simply the difference between the actual (in our case sampled) and the quantized.

$$Percentage\ Error = \frac{|sampled\ -\ quantized|}{|\text{sampled}|}x100\%$$

Exercise:

1. (Hand calculation) The *sampled* data is $2sin(2\pi n * freq)$, where freq = 5. What is the value of *sampled* at n = 1/Fs (where Fs = 25)? What would be its corresponding quantized value (round to the nearest integer)? What is the corresponding percentage error?
2. (MATLAB) Plot the percentage error of the ADC we just created. Notice that value n = 1/Fs is the second value of the sampling interval, *n*, in our code. So the corresponding percentage error we computed in (1) above should also be the second value in the percentage error vector we computed here. Did you get the same value?
3. (Bonus) Looking at the plot in (2), what random variable would you use to describe the percentage error? (you may choose a discrete or a continuous RV). Explain your answer.

II. Matrix Operations

The strength of MATLAB (whose name stands for MATrix LABoratory) is in matrix and vector manipulation. In the previous section, the variables are represented as vectors or 1D matrices. Vectors can also be defined by specifying the elements, for example, v = [1, 2, 3, 4]. Equivalently, we can define v similar to how we defined t and n in the previous section: v = 1:1:4. That is, a vector from 1 (first value) to 4 (third value) with an interval of 1 (second value).

The matrix $M = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix}$ may be defined in MATLAB as M = [1 2 3; 4 5 6]. To access row 1, column 2 of M, simply use M(1,2). Try to see what the following commands will give: (a) zeros(3,4); (b) ones(3,4); eye(3).

Exercise:

1. Define matrices $M = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ and $N = \begin{bmatrix} 1 & 1 & 1 \\ 1 & -8 & 1 \\ 1 & 1 & 1 \end{bmatrix}$. Let us then define matrix **A = M*N** and matrix **B = M .* N** . What is the difference between the two operations used: " ***** " and " **.*** " ?

2. The following system of equations may be solved using matrices:

$$x - 2y + 3z = 9 \qquad (1)$$
$$-x + 3y - z = -6 \qquad (2)$$
$$2x - 5y + 5z = 17 \qquad (3)$$

   We can re-write this system of equations as

$$\begin{bmatrix} 1 & -2 & 3 \\ -1 & 3 & -1 \\ 2 & -5 & 5 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} 9 \\ -6 \\ 17 \end{bmatrix}$$

   Write a MATLAB code to solve this system of equations. What is the solution to these equations?