Full Name: ______________________________ **Lab Section:** ____________

EEL 4750 / EEE 5502 (Fall 2018) – Code #02 **Due Date:** **Sept. 20, 2018**

Question #1: *(1 pts)* How many hours did you spend on this homework?

Question #2: *(4 pts)* *Correlation*
Consider the expression for correlation (also known as cross-correlation) between $x[n]$ and $y[n]$,

$$c[n] = x[-n] * y[n] = \sum_{m=-\infty}^{\infty} x[m]y[n+m] \tag{1}$$

Let us discuss some properties of the correlation function.

(a) Show that if $x[n] = 0$ for $n < 0$ and $n > N_x - 1$ (i.e., it is casual and has a length of N_x) as well as $y[n] = 0$ for $n < 0$ and $n > N_y - 1$ (i.e., it is causal and has a length of N_y), then $x[n] * y[n]$ must satisfy

$$x[n] * y[n] = 0 \quad \text{for} \quad n < 0 \quad \text{and} \quad n > N_x + N_y - 2\ .$$

That is, it is causal and has a length of $N_x + N_y - 1$. Illustrations and correct intuition will be acceptable.

(b) Show that if $x[n] = 0$ for $n < 0$ and $n > N_x - 1$ (i.e., it is casual and has a length of N_x) as well as $y[n] = 0$ for $n < 0$ and $n > N_y - 1$ (i.e., it is causal and has a length of N_y), then $c[n]$ must satisfy

$$c[n] = 0 \quad \text{for} \quad n < -(N_x - 1) \quad \text{and} \quad n > N_y - 1\ .$$

That is, it is non-causal and has a length of $N_x + N_y - 1$. Illustrations and correct intuition will be acceptable.

(c) Let $y[n] = x[n]$. Under this condition, the correlation is known as the auto-correlation,

$$c[n] = \sum_{m=-\infty}^{\infty} x[m]x[n+m] \tag{2}$$

For this condition, show that $c[n]$ is largest when $n = 0$.

Hint: Use the Cauchy-Schwarz inequality, defined by

$$\left| \sum_{m=-\infty}^{\infty} x[m]y[m] \right|^2 \leq \sum_{m=-\infty}^{\infty} |x[m]|^2 \sum_{k=-\infty}^{\infty} |y[k]|^2 \tag{3}$$

(d) Let $y[n] = x[n - n_0]$. For this condition, show that $c[n]$ is largest when $n = n_0$.

Question #3: *(3 pts)* *Convolution*

Create the following signals in MATLAB:

```
>> x1 = sin(pi/10*(0:19));
>> x2 = sin(pi/5*(0:19));
>> x3 = sin(pi/2*(0:19));
>> x4 = (-1).^(0:19);
```

And concatenate each of these four signals:

```
>> z = [x1 x2 x3 x4];
```

(a) Plot `z` using `stem(z)`

(b) Create three systems with impulse responses $h_1[n]$, $h_2[n]$, and $h_3[n]$ (or `h1` and `h2` and `h3` in MATLAB) that are also 20 samples in length. Define these impulse responses as

$$\begin{aligned} h_1[n] &= \delta[n] \\ h_2[n] &= \delta[n-5] \\ h_3[n] &= \delta[n-19] \end{aligned}$$

Perform three convolutions in MATLAB between $h_1[n]$, $h_2[n]$, and $h_3[n]$ and $x_1[n]$ by running

```
>> y1 = conv(x1, h1)
>> y2 = conv(x1, h2)
>> y3 = conv(x1, h3)
```

Plot `y1`,`y2`,`y3` using `stem`.

(c) The output signals now have a length of $N = 39$ instead of $N = 20$. Based on the previous results, why is this necessary?

Question #4: *(4 pts)* *The Auto-Correlation*

(a) Perform a correlation of `x1` with itself, i.e., an auto-correlation. Perform the auto-correlation with the following commands:

```
>> a1 = conv(fliplr(x1), x1);
>> lag = -(20-1):(20-1);
>> stem(lag, a1)
>> xlabel('Lag [samples]')
>> ylabel('Amplitude')
```

As shown in Question #1, the auto-correlation is always maximum at $n = 0$. The x-axis for a correlation is referred to as the “lag” or ”delay.”

(b) Repeat this process for $x_2[n]$, $x_3[n]$, $x_3[n]$, and $x_4[n]$. Plot the auto-correlations (like above) for all four signals.

(c) Perform four correlations between $z[n]$ and $x_1[n]$; $z[n]$ and $x_2[n]$; $z[n]$ and $x_3[n]$; and $z[n]$ and $x_4[n]$. Hence you can plot the result `z1` with

```
>> z1 = % the correlation, determine yourself based prior results
>> n = % the x-axis, determine yourself based on Question #2
>> stem(n, z1)
>> xlabel('Lag [samples]')
>> ylabel('Amplitude')
```

Plot the correlations for all four signals.

(d) Based on your results, how can we use this result to determine the location of our four signals $x_1[n]$, $x_2[n]$, $x_3[n]$, $x_4[n]$ buried in a large signal $z[n]$?

Question #5: *(5 pts)* *Finding a Hidden Message*

Included with the zip file is a p-file (an obfuscated m-file) function

```
>> [message, code] = get_message(ufid);
```

This function provides a message with a hidden code. Your objective is to locate the location of the *first sample* of the code with correlation. Submit your .m file used to achieve this, plot your correlation, and display your resulting index.

Note: This works well for small data sets (like the example here), but this is extremely slow for larger datasets (like audio recordings). In this next coding assignment, we will see how to speed this up with the discrete Fourier transform.