IBEHS 3A03 – Biomedical Signals and Systems (2019)

Tutorial Assignment #2: LTI System Responses and Convolution

Groups: Please complete this assignment in groups of 2 within your tutorial section.

Submission deadline: Before your next tutorial, one week from when your tutorial section starts this Tutorial Assignment (except for section T01, which has a one-week extension due to Reading Week).

Submission requirements: Uploaded to Avenue: i) a PDF report with plots and typed answers to the questions, and ii) MATLAB .m files that generate the plots in your report.

Posted on Avenue in the content section **Tutorial Assignments** and subsection *Tutorial Assignment 2* are three MATLAB files named `ltisystemA.p`, `ltisystemB.p`, and `ltisystemC.p`.

These functions all operate in the same fashion as the .p files from the first tutorial assignment (i.e., `y = ltisystemN(n,x)` where `n` is a time-step index vector and `x` is an input signal vector), but they all implement *linear, time-invariant* systems. Note that all of these functions assume zero initial conditions, i.e., $x[n] = 0$ and $y[n] = 0$ for all values of n before the first time-step value in your supplied input array `n`, and that $x[n] = 0$ for all values of n after the last time-step value in your supplied input array `n`.

Download these files, and develop a MATLAB script (i.e., .m file) that for each of these systems A–C:

I. computes the system output when the input is the unit impulse function $\delta[n]$,

II. computes the system output when the input is the unit step function $u[n]$,

III. tests that the unit step function output is equal to the cumulative sum of the unit impulse response (hint: MATLAB has a built-in function `cumsum`),

IV. computes the system output when the input is the example ECG signal from the first assignment `ECGexample1.mat`,

V. computes the system output when the input is the example respiration signal from the first assignment `respiration_example.mat`,

VI. tests that the outputs for the ECG and respiration signals are equal to convolution of those input signals with the impulse response $h[n]$ of the system computed in part I above (hint: MATLAB has a built-in function `conv`).

Note that the output array `y` from each of these functions is only as long as your input array `x`, so you will need to make sure that your input array is long enough (i.e., has enough time steps) to capture the bulk of the impulse and step responses in parts I and II.

In your report, you should: i) show the plots from your code for parts I to VI, and ii) comment on where the tests that you performed in parts III and VI do or do not match the theory. In any case where your results do not (entirely) match the theory, give an explanation of why.

Again, it is okay for you to *discuss* your methodologies and your results with other groups, but *do not share code* or *reports between groups*. All work submitted on Avenue will be assessed for plagiarism via TurnItIn.