Assignment #5: B-mode Ultrasound Imaging

DUE: Sunday, 2016-11-20 at midnight.

- You will be working in your assigned group. [1] Please create a repository called `ultrasound_netid_netid` (fill in netid of each student member). Be sure to add Dr. Palmeri (`mlp6`) as a Master access-level member.
- Use all good git repository management practices that have been promoted all semester.
- Create Issues (that are associated with Milestones) for all development tasks on the project, and **assign a specific group member** to each task. While this is a group project, each group member will be graded individually based on their contributions to the project, so strive to have even effort distribution, as represented by these issues. **Be sure to associate commits with specific issues.**
- Use all good python coding practices that have been promoted all semester, including PEP8 style compliance.
- Your group will be developing code to process raw, radiofrequency (RF) ultrasound data from a linear array. The ultrasound physics and image formation procedures will be reviewed during lecture. Your code will need to do the following to generate a B-mode ultrasound image:
  - Data acquisition metadata will be stored in a JSON text file. The specific parameters that will be available include:
    * `fs`: sampling frequency (Hz)
    * `c`: sound speed (m/s)
    * `axial_samples`: number of samples in depth
    * `num_beams`: number of lateral beams
    * `beam_spacing`: spacing between lateral beams (m)

    Your code should accept an input argument for the JSON filename, with a default of `bmode.json`.
  - The RF data will be serially stored as `int16` binary data ordered as samples from shallow to deep for a single beam, followed by subsequent lateral beams from left-to-right in the image. Your code should accept an input argument for the RF binary filename, with a default of `rfdat.bin`.
  - B-mode image formation will involve the steps of envelope detection and logarithmic compression discussed in lecture.
  - Your code should provide an option to either:
    * Render a B-mode image using `matplotlib` with axial and lateral dimensions labeled in meters. This should be done with a `--display` Boolean input argument that is `False` by default.
    * Save a PNG file of the B-mode image, with the user able to input a desired filename, with a default of `bmode.png`. This should be done with a `--save` Boolean input argument that is `True` by default.
    * These options should not be exclusive of one another (i.e., the user can choose to do both).
- Create an annotated tag (`v1.0.0`) of your final version.

[1] `https://gitlab.oit.duke.edu/medical-device-software-design/rw_data`

- Grading criteria:
  - Git Repository
    - Issues/Milestones [10%]
    - Commits are discrete, logical changesets [10%]
    - Branching & Merging [5%]
  - Modular coding
    - Separate modules for JSON reading, binary reading, envelope detection, logarithmic compression, image display, image saving. [10%]
    - Avoidance of hard-coded variables; robust functional input for algorithmic control. [10%]
  - Full unit test coverage of all functions, with passing CI build[2] [20%]
  - Logging: INFO, DEBUG, ERROR [10%]
  - Sphinx documentation for each module/function [10%]
  - Handle and raise exceptions [5%]
  - Functionality [10%]
- Test JSON and binary image data are available in: `https://gitlab.oit.duke.edu/medical-device-software-design/bmode_ultrasound`.

[2] Dr. Palmeri will enable gitlab runner for your repository.