PHY 250L – Spring 2018
Computing tutorial 7

Welcome back to the PHY 250L computing tutorial, PHASE 2 MATLAB woooooooo! This week, you'll further hone your MATLAB syntactical skills with a few longer problems. You know how to develop algorithms for all of the tasks below – you simply need to figure out how to do it in MATLAB.

Please consult the MATLAB notes from class (see Sakai).

Problems for 4.12.2017 The following problems should be completed (in MATLAB, duh) and uploaded to Sakai by 09:45 on 4.12.2017. Each problem should correspond to its own MATLAB program (*i.e.*, each problem will correspond to a single file). The preferred names for the files are indicated in each problem.

1. `iris_plots.m`, 50 points
   Acquire **the** iris dataset from the internet. WHICH iris dataset, you ask? Well, there are probably many, but there is only one that is used as a benchmark for data science tasks. It gives characteristics for 150 samples of three species of iris (*setosa*, *versicolor*, *virginica*). Read in the data, and make three scatter plots:

   - petal length vs petal width
   - sepal length vs petal length
   - sepal length vs petal length with marker size dictated by sepal width and color by species

   When you read in the data, you will probably want to dump it to new arrays that will be easier to manipulate. Include all three plots on the same figure, and enlarge for easy readability.

   Lots of good stuff here `https://vincentarelbundock.github.io/Rdatasets/datasets.html`, but check Wikipedia for a description of the iris dataset.

2. `twod_walk1.m`, 10 points
   Last week you wrote code that generated a random walk in one dimension. In the following problems you'll generate many random walks in two dimensions. The step length for these random walks will be equal to 1, and the walker will have an even probability for stepping in any direction in the 2-d plane. Thus, for step number i, the change in the walker's position will be dictated by some angle $\phi \in [0, 2\pi)$:

   $$\Delta x_i = \cos\phi, \quad \Delta y_i = \sin\phi \tag{1}$$

   We want to know how far from the origin the walker is after 1000 steps. Because the random walk is *random*, we'll need to investigate this problem using the Monte Carlo technique. This means that you'll generate *many* trajectories, and average the results.

   For the first step, generate a single trajectory (*i.e.*, 1000 x, y points) and plot it as a scatter with the points connected. This will give you an idea of what a 2-d random walk looks like. (This is close to how Brownian motion works, too!) Do not proceed until you are confident that this bit of your code works properly.

3. `twod_walk2.m`, 20 points
   Now generate 1000 random walks and make a histogram of the final displacements (*i.e.*, $\sqrt{x^2 + y^2}$). It would be useful to write the trajectories to file in some easy-to-reuse format (ahem).

4. `twod_walk3.m`, 20 points
   Now make a scatter plot of the average displacement after n steps (δ_n) *vs* n. There are several ways to do this. Think before you act. We don't yet know how to fit with MATLAB, but what does the relationship look like?