CE640 / OC599 – MATLAB – Fall 2013
Class 3 Assignment
Functions and Program Control
(due beginning of class, October 22)

1. Most of you did HW1 using for loops (which we had not yet covered) or 'brute force' where you just built up the datasets one by one, computed stats, and plotted the result. Repeat the exercise in a cleaner fashion using a 3d matrix. Don't know how to do a 3d matrix? Easy. As an example, zeros(10,10,10) creates a 'cube' of 1000 zeros. You can access elements as before, for example A(3,5,1) or B(1:2,:,3).
2. Write a function, called multGauss, to compute the sum of multiple Gaussians. The function should itself call the function gaussmf (built-in matlab function) to do the basic calculation, returning only the sum. The easiest way to indicate the number of required Gaussians might be from the number of rows in a matrix, beta, each of which contains the parameters of each component Gaussian. Include error checking to ensure that the input parameter matrix has three columns, returning an error if not. To clarify, the input to multGauss should be (i) an input matrix with three columns (see help on gaussmf; the third parameter is simply the amplitude (default is 1) of the Gaussian) for the parameters and at least one row (each row being an individual Gaussian; (ii) a vector x specifying the domain for the calculations. The output from multGauss is simply the sum (over the ensemble of Gaussians, NOT over x).
3. Write an m-file that issues a call to multGauss and plots the result. Let the domain range from 0 to 20. Let there be four Gaussians. The amplitudes are to be 0.1, 0.5, 1, and 2. The standard deviations (sigma) are to be 0.5, 1, 2 and 3. The positions (C) are to be 2, 4, 6, and 8.
4. Write an m-file to calculate the mean, variance and standard deviation of a random number sequence generated with the command data = rand(10000,10). Your mfile should have two parts. In the first part, use for loops and the following equations

$$\bar{y} = \frac{1}{N}\sum_{i=1}^{N} y_i$$

$$\mathrm{var}(y) = \frac{1}{N-1}\sum_{i=1}^{N}\left(y_i - \bar{y}\right)^2$$

$$std(y) = \sqrt{\mathrm{var}(y)}$$

   In other words, old-school programming. No cheating! You should get 10 means, 10 variance, and 10 stds (one for each column). Use tic and toc to determine how long it takes to calculate using for loops. The second part of your m-file should do the same calculations using the built-in matlab functions mean, var, std. Compare the performance (tic toc).
5. Repeat 3 with a slight variation. You want to compute the statistics only on the numbers that are greater than 0.5. Again, do it old-school style, with for and if loops. Measure the performance. Then, do it with matlab niceties. Find command and so on. Compare performance.