Tutorial problems for Lectures 4/5. Due Monday April 30, 2017.

Problem 1: Write a function that will shift an array by an arbitrary amount using a convolution (yes, I know there are easier ways to do this). The function should take 2 arguments - an array, and an amount by which to shift the array. Plot a gaussian that started in the centre of the array shifted by half the array length. (10)

Problem 2: The correlation function $f \star g$ is $\int f(x)g(x+y)dx$. Through a similar proof, one can show $f \star g$=ift(dft(f)*conj(dft(g))). Write a routine to take the correlation function of two arrays. Plot the correlation function of a Gaussian with itself. (10)

Problem 3: Using the results of part 1 and part 2, write a routine to take the correlation function of a Gaussian (shifted by an arbitrary amount) with itself. How does the correlation function depend on the shift? Does this surprise you? (10)

Problem 4: The circulant (wrap-around) nature of the dft can sometimes be problematic. Write a routine to take the convolution of two arrays *without* any danger of wrapping around. You may wish to add zeros to the end of the input arrays. (10)

Problem 5: Complete the complex class definition to support -,*, and / (__sub__, __mul__, and __div__). Recall that a/b = a*conj(b)/(b*conj(b)). Show from a few sample cases that your functions work. (10)

Bonus 1: You have a sample code that calculates an FFT of an array whose length is a power of 2. Using that routine as a guideline, write an FFT routine that works on an array whose length is a power of 3 (e.g. 9, 27, 81). Verify that it gives the same answer as numpy.fft.fft (10)

Bonus 2: Extend the complex class to also support arbitrary (i.e. non-integer) powers (keyword is __pow__). +3 if the routine works if ab works for complex a and real b, +5 if it works for complex a and complex b. (you may ignore branch cuts).