Coding in advance of the Feb 5, 2018 class

We'll do some coding in this course, which I hope will help build your understanding of the concepts we discuss. Use any language you like; it will also be helpful to be able to plot results. What I'll do is ask you at the end of one class to write code for the next class. I won't check your results; this is for you! You should read the lecture notes for the class before trying the coding exercise, because in the notes I will often go over a similar example, to get you started.

Write a code that draws x from an exponential distribution, $P(x) = e^{-x}$ (where x can be from 0 to ∞), n times, computes the arithmetic mean, and then does it again, a total of m times. To do a single draw, select a random number y uniformly between 0 and 1 (this is called a uniform random deviate, with a range from 0 to 1). Then $x = -\ln(y)$ selects properly from the distribution $P(x) = e^{-x}$ (you might want to think about how to prove this).

Your goal is that, given n and m, your code will produce a plot with the probability distribution of the arithmetic mean. In particular, please do so for $n = 1, 2, 5, 10$, and 100, and do $m = 10,000$ iterations each time. The details here will be useful when we think more about continuous probability functions. How should you gather your results to show the distribution? For example, as we'll see in the second class, we expect that the distribution of the arithmetic mean should narrow more and more as n becomes larger, so it's not helpful to just plot the same range of x for any n.

If you have the time, you might want to compute the arithmetic mean and the standard deviation expected for the function $P(x) = e^{-x}$. Then you can determine the accuracy of the central limit theorem as you compute the distribution of the arithmetic mean of more and more draws from this distribution.