Assignment 0

It will be very helpful for me for students to complete this assignment to the best of their ability so that I can guage the level of mathematical background of the class. Because of that, some of these questions are difficult and **I absolutely do not expect all of them to answered completely.** A reminder that assignments such as these are graded only on completion; for this assignment in particular, if you have no idea how to respond to a question, a simple "idk" will suffice.

1. (a) Order the following functions of n by asymptotic growth from least to greatest (no need to show work):

$$n^2 \qquad 100n \qquad 20n \log n \qquad \log(n^3) \qquad 2^n \qquad 50$$

   (b) How many (none, some, most, all) of these do you think you could formally show to someone else if necessary?

2. (a) The worst case runtime of bubble sort on a list of n items is:

   (b) Convince me (your professor, who you can assume knows everything you've learned in your classes in CS so far) that bubble sort won't take longer than whatever bound you gave in (a).

   (c) Convince me that you can't give a better (ie, lower) bound than you gave in (a).

Assignment 0

3. Consider the following scheduling problem: a motivated student has collected a list of n classes of varying durations they would like to take, without any preferances between classes in their list. The student knows the start and end times of all the classes. Desiring to take as many classes as possible, the student decides to take the classes in order of shortest class length to greatest.
   Their strategy will not necessarily work to ensure they take as many classes as possible. Convince me of this.

4. Consider the following algorithm to determine if a list $L = (e_1, ..., e_n)$ never repeats an element (so $e_1 \neq e_2, e_3, ..., e_n$, and $e_2 \neq e_3, ..., e_n$, and so on):

   i) For each item e_i

      a) For each other item e_j where $j \neq i$, if $e_j = e_i$ announce that e_j is repeated and end.

   ii) If this line is reached, then announce that all items are unique and end.

   What is the worst case runtime of this algorithm as a function of n?