CS311:	**Homework 1**	Due in class Tuesday, Sep. 9, 2008

Cooperation: Students are encouraged to talk to each other about the subject matter of this class and help each other. It is fine to discuss the problems and ask questions about them. I encourage such questions in class and office hours as well as elsewhere. However, there is a line past which you must not go, e.g., sharing or copying a solution is not okay and could result in failure. If a significant part of one of your solutions is due to someone else, or something you've read then **you must acknowledge your source!** Failure to do so is a serious academic violation, likely to result in failure of the course or worse. Furthermore, **all solutions must be written by yourself, in your own words**. You may get an idea from somewhere or someone and acknowledge that, but you must still understand it and explain it yourself. **A copied solution part, even with the source acknowledged, will be considered plagiarism.** The exception is if it is in quotation marks and cited specifically. But in this case, don't bother because you won't get credit for quoting someone else's solution.

Note: Part of the goal of this first problem set is for you to recall the precision of mathematical notation and reasoning that you learned in CS250. This will be valuable in CS311 as we learn to calculate good upper bounds on algorithm running times. That is, we will require mathematical proofs as iron-clad guarantees that an algorithm takes at most a given number of steps for all inputs of a given size. Throughout this course, we will use "n" as the size of the input. (In some cases we may consider more than one parameter to measure the size of the input, e.g., for graphs we will measure running times in terms of n, the number of vertices of the input graph, and m, the number of edges.)

Problems:

0. You are encouraged to send me an email telling me a bit about yourself, what you would like to be called, and any hopes or concerns you have about the course.

1. [25 pts.] All technical texts require active reading. You should jot down examples and work through the examples given in the text and you should try a bit to prove each claim before you read the proof in the text. Trying to do it yourself keeps you more invested, and thus understanding better. If you can prove it yourself without reading the proof, that's great and should be rewarding, i.e., "I'm really getting this." If you can't, that's fine, and the more usual situation, but now you are well motivated to notice the new idea you hadn't thought of that lets the proof go through.

   I like our textbook a lot, but it is sometimes a bit brief. That is, it gives a good intuition and a clear statement but sometimes skips details. This means that you will need to learn to figure out the missing details, or at least ask about them if you notice that something is missing but you have trouble filling it in yourself. Obviously noticing that something is missing is an important step towards being able to fill in the missing details yourself.

   At the bottom of page 6, the text gives the following definition of Big-Oh notation:

   Definition 1: "Let $f(n)$ and $g(n)$ be functions from positive integers to positive reals. We say $f = O(g)$ (which means that 'f grows no faster than g') if there is a constant $c > 0$ such that $f(n) \leq c \cdot g(n)$."

A more standard definition is the following:

Definition 2: "Let $f, g : \mathbf{Z}^+ \to \mathbf{R}^+$. We say $f = O(g)$ iff

$$\exists N \in \mathbf{Z}^+ \, \exists c \in \mathbf{R}^+ \, \forall n \in \mathbf{Z}^+ (n \geq N \;\; \to \;\; f(n) \leq c \cdot g(n))$$

Prove that Definitions 1 and 2 are equivalent. To do this, you must show that for all appropriate functions f and g, if they satisfy Def. 1 then they satisfy Def. 2 and if they satisfy Def. 2 then they satisfy Def. 1.

After you have done this, tell me which definition you prefer, and why you think that Def. 2 might be more standard.

2. [25 pts.] In each of the following four examples, say whether $f = O(g)$ or $f = \Omega(g)$, or both, i.e., $f = \Theta(g)$. Justify your answers in each case, i.e., why something holds or why it does not.

	$f(n)$	$g(n)$
(a)	$100n + \log n$	$n + (\log n)^2$
(b)	$n \log n$	n
(c)	$n \log n$	$n^{1.1}$
(d)	$n2^n$	3^n

3. [25 pts.] Do problem 0.2, p. 9 of the text: how to evaluate a geometric sum. (I'll copy it here – just in hw1 – because some of you may not have the text yet.)

   Let $g(n) = \sum_{i=1}^{n} c^i$ where $c \in \mathbf{R}^+$. Show that

   (a) If $c < 1$ then $g(n) = \Theta(1)$, and

   (b) If $c = 1$ then $g(n) = \Theta(n)$, and

   (c) If $c > 1$ then $g(n) = \Theta(c^n)$.

4. [25 pts.] [Problem 1.4, p. 38] Show that $\log(n!) = \Theta(n \log n)$.

   [Hint: to show an upper bound, compare $n!$ with n^n. To show a lower bound, compare it with $(n/2)^{n/2}$.