CS 332: Algorithms
Homework #2

Assigned: Wednesday, January 30, 2002

Due: Wednesday, Feb 6, 2002

1. Rank the following functions by order of growth; that is, find an arrangement $g_1, g_2, \ldots, g_{25}$ of the functions satisfying $g_1 = \Omega(g_2)$, $g_2 = \Omega(g_3)$, … $g_{24} = \Omega(g_{25})$. Partition your list into equivalence classes such that $f(n)$ and $g(n)$ are in the same class iff $f(n) = \Theta(g(n))$.

$(3/2)^n$	$\left(\sqrt{2}\right)^{\lg n}$	$\lg^* n$	n^2	$(\lg n)!$
n^3	$\lg^2 n$	$\lg(n!)$	2^{2^n}	$n^{1/\lg n}$
$\lg\lg n$	$n \cdot 2^n$	$n^{\lg\lg n}$	$\ln n$	2^n
$2^{\lg n}$	$(\lg n)^{\lg n}$	$4^{\lg n}$	$(n+1)!$	$\sqrt{\lg n}$
$n!$	$2^{\sqrt{2\lg n}}$	n	$n\lg n$	1

2. Argue informally that the quicksort routine presented in the book will run in time $\Theta(n \lg n)$ when all elements in the array are equal.

3. Is $2^{n+1} = O(2^n)$? Is $2^{2n} = O(2^n)$?

4. A sorting algorithm is described as *stable* if equal elements are in the same relative order in the sorted sequence as in the unsorted sequence. Which of insertion sort, quicksort, and mergesort are stable? Give a simple fix to make the unstable sorts stable.

5. CLR 4-1 a-f (also 4-1 in old book)

6. CLR 6-2 a-d (7-2 in old book)