HW1, Due Sept 18.

1. Asymptotically, $f(n) = ?(g(n))$: Prove your answer.
   (a) $f(n) = 2^n, g(n) = 3^n$.
   (b) $f(n) = \log_2 n, g(n) = \log_3 n$.
   (c) Let $f(n) = \log(\sqrt{n}), g(n) = (\log n)$.

2. Arrange the following functions in nondecreasing order of their asymptotic growth, using o, Θ, whichever is most appropriate between successive functions in your list.

$$\log n, 2^{\sqrt{2 \log \log n}}, n^{1000/\log n}, 2^{2^n + \log n}, (n+1)!, 0.001n^3, n2^{2^n}$$

3. Solve: $T(2) = 1$ and $T(n) = T(\sqrt{n}) + 1$. What is $T(n)$?: $O(n)$, $O(\log n)$, $O(\log \log n)$ or $O(1)$? Verify your answer.

4. Solve the following using the Master Theorem:
   (a) $T(1) = 1$ and $T(n) = 2T(n/3) + 10$.
   (b) $T(2) = 15$ and $T(k) = T(k/2) + \log k$.
   (c) $T(1) = 1$ and $T(n) = 5T(n/4) + \sqrt{n}$.

5. Write a recurrence equation that will be difficult (if not impossible) to solve using the Master Theorem. Also, describe a problem and the algorithm for it, for which the analysis leads to your recurrence equation.

Suugested Exercise. Basics, to refresh Discrete Math Prove: A binary tree with n nodes has depth at least $\lfloor \log n \rfloor$ and at most $n - 1$. (Hint: Show that if a binary tree has depth d and has n nodes, then $n \leq 2^{d+1} - 1$.)

Extra Credit Solve: $T(1) = 1$ and

$$T(n) = n + \sum_{i=1}^{n-1} T(i)$$

.