Due Date: January 31, 2013

Problem 1: In each of the following cases, rank the functions by order of their growth. Here log \( n \) means \( \log_2 n \).

- \( 2^{\log n}, (\log n)^{\log n}, e^n, 4^{\sqrt{\log n}}, n!, \sqrt{\log n} \)
- \( (\frac{3}{2})^n, n^2, (\log n)^2, \log(n!), 2^{2n}, n^{\frac{1}{\log n}}, n^{\frac{1}{\log \log n}} \).

Problem 2: Show that, if \( c \) is a positive real number, then \( g(n) = 1 + c + c^2 + c^3 + \cdots + c^n \) is:

- \( \Theta(1) \) if \( c < 1 \)
- \( \Theta(n) \) if \( c = 1 \)
- \( \Theta(c^n) \) if \( c > 1 \)

Problem 3: Solve the following recurrences by expanding the terms or using induction and give a \( \Theta \) bound for each of them. If you use induction, you can use the master theorem to guess the bound. In all the cases, assume \( T(k) = O(1) \) if \( k \) is a constant.

- \( T(n) = 5T(n/4) + n \)
- \( T(n) = T(\sqrt{n}) + 1 \)
- \( T(n) = T(n - 1) + n^c \)

Problem 4: Give an efficient algorithm to compute the least common multiple of two \( n \)-bit numbers \( x \) and \( y \), that is, the smallest number divisible by both \( x \) and \( y \). What is the running time of your algorithm as a function of \( n \)?

Problem 5: The \( k \)th quantiles of an \( n \)-element set are the \( k - 1 \) order statistics that divide the sorted set into \( k \) equal-sized sets (to within 1). That is, compute the elements of rank \( \lceil in/k \rceil \) for all \( 1 \leq i < k \). Give an \( O(n \log k) \)-time algorithm to list the \( k \)th quantiles of a set.