Need more problems? A few recommended exercises follow.

Out: Thursday, January 26, 2017. Please do NOT submit solutions to these problems. Solutions will be posted after February 6.

1. In the table below, indicate the relationship between functions f and g for each pair (f, g) by writing "yes" or "no" in each box. For example, if $f = O(g)$ then write "yes" in the first box. Here $\log^b x = (\log_2 x)^b$.

f	g	O	o	Ω	ω	Θ
$\log^2 n$	$25 \log n$					
$\sqrt{\log n}$	$(\log \log n)^4$					
$3n \log n$	$n \log 3n$					
$n^{3/5}$	$\sqrt{n} \log n$					
$\sqrt{n} + \log n$	$2\sqrt{n}$					
$n^2 2^n$	3^n					
$\sqrt{n} 2^n$	$2^{n/2 + \log n}$					
$n \log 3n$	$\frac{n^2}{\log n}$					
$n!$	2^n					
$\log n!$	$\log n^n$					

2. Show that, if λ is a positive real number, then $f(n) = 1 + \lambda + \lambda^2 + \ldots + \lambda^n$ is

   (a) $\Theta(1)$ if $\lambda < 1$.

   (b) $\Theta(n)$ if $\lambda = 1$.

   (c) $\Theta(\lambda^n)$ if $\lambda > 1$.

   Therefore, in big-Θ notation, the sum of a geometric series is simply the first term if the series is strictly decreasing, the last term if the series is strictly increasing, or the number of terms if the series is unchanging.

3. (16 points) Give tight asymptotic bounds for the following recurrences.

   - $T(n) = 4T(n/2) + n^3 - 1$.
   - $T(n) = 8T(n/2) + n^2$.
   - $T(n) = 6T(n/3) + n$.
   - $T(n) = T(\sqrt{n}) + 1$.

4. (31 points) The Fibonacci numbers are defined by the recurrence

$$F_0 = 0, F_1 = 1$$
$$F_n = F_{n-1} + F_{n-2} (n \geq 2)$$

   (a) (4 points) Show that $F_n \geq 2^{n/2}$, $n \geq 6$.

   (b) Assume that the cost of adding, subtracting, or multiplying two integers is $O(1)$, independent of the size of the integers.

   - (4 points) Write pseudocode for an algorithm that computes F_n based on the recursive definition above. Develop a recurrence for the running time of your algorithm and give an asymptotic lower bound for it.
   - (4 points) Write pseudocode for a non-recursive algorithm that asymptotically performs fewer additions than the recursive algorithm. Discuss the running time of the new algorithm.
   - (10 points) Show how to compute F_n in $O(\log n)$ time using only integer additions and multiplications.
     (Hint: Express F_n in matrix notation and consider the matrix

$$\begin{pmatrix} 0 & 1 \\ 1 & 1 \end{pmatrix}$$

     and its powers.)

   (c) (9 points) Now assume that adding two m-bit integers requires $\Theta(m)$ time and that multiplying two m-bit integers requires $\Theta(m^2)$ time. What is the running time of the three algorithms under this more reasonable cost measure for the elementary arithmetic operations?