Fundamental Algorithms, Section 003
Homework 1, Additional Problems, Fall 22.
Updated, Thursday September 8.

1.a. Let $f(n) = 2n^2 + n \log n$ and $g(n) = n^2$. Show that $f(n) = \Theta(g(n))$.

b. Let $f(n)$ and $g(n)$ be two positive functions of n, with $f(n) = O(g(n))$. Does $2^{f(n)} = O(2^{g(n)})$? If your answer is "yes", prove it, and if it is "no", provide a counter-example.

2. Consider the problem of "combining" k sets $S_1, S_2, \ldots, S_k$ of sizes $s_1 \le s_2 \le \cdots \le s_k$, respectively. The number of operations needed to combine S_i and S_j is $s_i \cdot s_j$.

Suppose, the combining is performed as follows: repeatedly, two sets are chosen and combined, and then returned to the remaining collection of uncombined sets, stopping when one set remains. The size of a combined set is the sum of the sizes of its two constituent sets.

a. Suppose $k = 3$. Show that the number of operations is the same regardless of the order in which the combinings are performed.

b. Suppose $k = 4$. Show that the number of operations is the same regardless of the order in which the combinings are performed.

c. Show the same result for arbitrary k. Look for a simple argument to justify this result.

Deferred till next week.
3. Suppose the moves in the Tower of Hanoi problem are restricted to be all in a clockwise direction, i.e. from Pole A to Pole B, B to C, and C to A. Give a recursive procedure to move n rings from Pole A to Pole B.
Hint. You will want to use two mutually recursive procedures.