Analysis of Algorithms: Assignment 2

Due date: September 9 (Thursday)

Problem 1 (3 points)
Write pseudocode for the $\textsc{Merge}(A, p, q, r)$ procedure.

Problem 2 (3 points)
For each of the following functions, give an asymptotically tight bound (Θ-notation). Make your expression inside Θ as simple as possible.

Example: $2n^3 + 3n^2 = \Theta(n^3)$.

(a) $(n^2 + n + 1)^{10}$

(b) $(\sqrt{n} + \sqrt[3]{n} + \lg n)^{10}$

(c) $n^{10} + 1.01^n$

(d) $n^{10} + 0.99^n$

(e) $2^n + n! + n^n$

(f) $2^{\lg n}$

Problem 3 (4 points)
Give an example of functions $f(n)$ and $g(n)$ such that $f(n) \neq O(g(n))$ and $f(n) \neq \Omega(g(n))$.

Problem 4 (bonus)
This problem is optional, and it does not affect your grade for the homework; however, if you solve it, then you will get 2 bonus points toward your **final grade** *for the course. You cannot submit this bonus problem after the deadline.*

Suppose that we have four algorithms, called A_0, A_1, A_2, and A_3, whose respective running times are n, n^2, $\lg n$, and 2^n. If we use a certain old computer, then the maximal sizes of problems solvable in an hour by these algorithms are s_0, s_1, s_2, and s_3.

Suppose that we have replaced the old computer with a new one, which is k times faster. Now the maximal size of problems solvable in an hour by A_0 is $k \cdot s_0$. What are the maximal problem sizes for the other three algorithms, if we run them on the new computer?