Homework 1
Fundamental Algorithms, Fall 2004, Professor Yap

Due: Thu Sep 23, in class

INSTRUCTIONS:

- Please read questions carefully. When in doubt, please ask.
- There are links from the homework page to the old homeworks from previous classes, including solutions. Feel free to study these.

---

1. (15 Points)
   In the first lecture, we described a conventional program for merging two sorted lists.
   (a) Please draw a comparison-tree for merging two sorted lists of numbers, $(x < y)$ and $(a < b < c < d)$. Your comparison-tree should be obtained by "unwinding" the algorithm described in class. HINT: you can appeal to symmetry to only draw a portion of the comparison-tree.
   (b) What is the height of your comparison tree?
   (c) Determine $M(2, 4)$ (give as good an upper and lower bound as you can). Do not quote some known result – you need to do an argument from first principles. HINT: make two separate arguments for upper and lower bounds, respectively.

2. (20 Points)
   We want to consider the "best case time" in comparison trees. If T is a tree, let $B(T)$ denote the length of a *shortest* path from the root to a leaf of T.
   (a) Define $M'(m, n)$ to be the length of a **shortest path** in some comparison-tree for merging two lists of sizes m and n. More precisely, $M'(m, n) = \min_T \{B(T)\}$ where T ranges over all comparison-trees for merging two lists of sizes m and n. Determine $M'(m, n)$.
   (b) Define $S'(n)$ to be the length of the length of a **shortest path** in some comparison-tree for sorting a list of size n, i.e., $S'(n) = \min_T \{B(T)\}$ where T ranges over all comparison-trees for sorting n elements. Determine $S'(n)$.

   HINT: (a) and (b) are quite easy, once you understand what the problem is about.

3. (15 Points)
   Use the Rote Method to solve the following recurrence:

   $$T(n) = n \lg n + 4T(n/2).$$

   Be sure to indicate each of the EGVS steps. You can choose your own initial conditions (the "strong form"). NOTE: $\lg n$ means $\log_2 n$.

4. (20 Points)
   Consider the real recurrences:

   $$T_0(n) = n + 2T_0(n/2)$$

   and

   $$T_1(n) = n + 2T_1(\lfloor n/2 \rfloor + 2).$$

   The initial conditions are given by $T_0(n) = T_1(n) = 0$ for $n < 4$.
   (a) Use real induction to show that $T_0(n) \le T_1(n)$. HINT: Use the fact that $T_1(n)$ is non-decreasing in n.
   (b) Fix $k_0 > 0$. Use real induction to show that for all $0 \le k \le k_0$, $T_0(n + k) = T_0(n) + O(k \lg n)$.
   (c) Show that $T_1(n) = O(T_0(n))$. HINT: you can reduce to the case (b) by showing $T_1(n) \le T_0(n+k)$.