CSE 2321 Homework 8

Turn In: Submit to the Carmen dropbox a PDF file generated from LaTex source (see the template file provided with this homework and the Piazza post on LaTex).

Reminder: Homework should be worked on individually. If you are stuck on a problem, please spend time thinking about the problem and trying different approaches before seeking help in office hours. If you come to office hours you will benefit more if you have already attempted these problems.

1. Consider the array $A = [8, 3, 4, 5, 2, 1, 0, 9, 7, 6]$.

   (a) (7 pts) Demonstrate how the InsertionSort algorithm works, by showing what A looks like at the end of each for loop iteration. (Class notes on Carmen have the algorithm).

   (b) (7 pts) Demonstrate how the MergeSort algorithm works, by showing how the array is split for each recursive call, and how the algorithm then merges the sorted subarrays. (Class notes on Carmen have the algorithm).

2. (20 pts) The transpose of a directed graph $G = (V, E)$ is the graph $G^T = (V, E^T)$, where

   $$E^T = \{(v, u) : (u, v) \in E\}$$

   Thus, G^T is G with all of the edges reversed. Create algorithms for computing G^T from G, for both the adjacency-list and adjacency-matrix representations of G. Analyze the running times of your algorithms, taking into account the differences in the data structures (i.e. checking if there is an edge from x to y takes time $\Theta(1)$ in an adjacency matrix, but takes longer in an adjacency list). Algorithms with lower running times will get more points.

3. Let

   $$A = a_1, a_2, \ldots, a_n$$

   be a list of n values with

   $$a_1 \leq a_2 \leq \ldots \leq a_n.$$

Suppose Alice creates a new list A' in the following way:
Alice chooses an integer i such that $1 \leq i \leq \lfloor \sqrt{n} \rfloor$, and reverses the order of the elements $a_i, a_{i+1}, \ldots, a_{i+\lfloor \sqrt{n} \rfloor}$, so that

$$A' = a_1, \ldots, a_{i-1}, a_{i+\lfloor \sqrt{n} \rfloor}, a_{i+\lfloor \sqrt{n} \rfloor - 1}, \ldots, a_{i+1}, a_i, a_{i+\lfloor \sqrt{n} \rfloor + 1}, \ldots, a_n.$$

(a) (5 pts) Give the running time (Θ) for insertion sort to sort A'. Provide a short explanation for your answer (1-2 sentences).

(b) (5 pts) Give the running time (Θ) for merge sort to sort A'. Provide a short explanation for your answer (1-2 sentences).

4. (5 pts) In the lecture on MergeSort I used this recurrence relation

$$\begin{aligned} T(n) &= 2T(n/2) + n \\ T(1) &= 1 \end{aligned}$$

to analyze the running time, getting that $T(n) = \Theta(n \log(n))$.
A more accurate recurrence relation for the MergeSort algorithm is

$$\begin{aligned} M(n) &= 2M(n/2) + n && \text{For even } n \\ M(n) &= M(\lfloor n/2 \rfloor + 1) + M(\lfloor n/2 \rfloor) + n && \text{For odd } n \\ M(1) &= 1 \end{aligned}$$

Consider the following analysis of $M(n)$:
Let $k \in \mathbb{N}$. Observe that

$$M(2^k) = T(2^k)$$

Therefore, for all n such that $2^{k-1} < n < 2^k$ we have

$$\begin{aligned} M(n) &\leq 2M(\lfloor n/2 \rfloor + 1) + n \\ &\leq 2M(2^k) + n && \text{since } n < 2^k \\ &= 2T(2^k) + n \\ &\leq 2T(2^k) + 2 \cdot 2^k && \text{since } n < 2^k \\ &= T(2 \cdot 2^k) && \text{substitute } 2 \cdot 2^k \text{ for } n \text{ in } T(n) \\ &\leq T(2 \cdot 2n) && \text{since } 2^k < 2n \\ &= T(4n) \end{aligned}$$

Thus, for all values of $n > 0$ we have $M(n) \leq T(4n)$.
We can use the recursion tree method to solve $T(4n)$:

$$\begin{aligned}
T(4n) &= 2T(2n) + 4n \\
&= 2[2T(n) + 2n] + 4n \\
&= 2[2[2T(n/2) + n] + 2n] + 4n \\
&= 2^{k+1}T(n/2^{k-1}) + 4(k+1)n & \text{after } k \text{ substitutions} \\
&= 2^{\log_2(n)+2}T(n/2^{\log_2(n)}) + 4(\log_2(n) + 2)n & k = \log_2(n) + 1 \\
&= 2^2 \cdot 2^{\log_2(n)}T(1) + 4n\log_2(n) + 8n \\
&= 4n + 4n\log_2(n) + 8n \\
&= \Theta(n\log(n))
\end{aligned}$$

and therefore $M(n) = O(n\log(n))$.

Come up with a similar argument to show that $M(n) = \Omega(n\log(n))$.