

Algorithms Second Quiz

10:10 ~ 11:00

(2023/11/3)

(Note that if you design an algorithm, you must have pseudo code to show your algorithm and analyze the algorithm's time complexity in the worst case. It would help to put comments after your pseudo code to clarify your algorithm.)

1. (10%) The solution to the recurrence $T(n) = 4T(n/2) + n$ turns out to be $T(n) = \theta(n^2)$. Show that a substitution proof with the assumption $T(n) \leq cn^2$ fails. Then, show how to subtract a lower-order term to make a substitution proof work.
2. (10%) Show that the worst-case running time of HEAPSORT is $\Omega(n \lg n)$.
3. (10%) Show that the average-case running time of QUICKSORT is $O(n \lg n)$.
4. (10%) Please give a decision tree for insertion sort operating on four elements: a , b , c , and d .
5. (10%) It is known that $\Omega(n \log n)$ is a lower bound for sorting problems. However, we have seen the radix sort algorithm, which can sort n items in $O(n)$ time. Is there a contradiction? If not, please explain. Can we conclude that the execution time of the radix sort algorithm is better than merge sort or heapsort? If not, please explain.