Reading Quiz #2

This is a preview of the published version of the quiz

Started: Feb 1 at 11:42am

Quiz Instructions

To prepare for this quiz, please read sections 5.1 to 5.3 (inclusive) in the Kleinberg + Tardos textbook.

The goal of this quiz is to lightly assess a first quick reading of these resources to prepare for class. You should definitely return to this material for a more thorough read to solidify your learning and prepare for assignments and exams.

Note that you are limited to only ***3 attempts*** on this quiz.

Best of luck! :-)

Question 1

1 pts

What is the tightest bound for a function T with the following recursion formula:

$$T\left(n\right) = T\left(\frac{n}{4}\right) + T\left(\frac{3n}{4}\right) + cn$$

$$T\left(1\right) = 1$$

○ O(n)

○ O(n^(3/2))

○ O(n * log(n))

○ O(n * log(log(n)))

Question 2

1 pts

Is this approach for showing merge-sort is O(n logn) correct? (Think about why or why not)

By guessing T(n) is O(n logn) we have for $n > N_0$:

$$T(n) = 2T\left(\frac{n}{2}\right) + cn \le 2k\left(\frac{n}{2}\right)\log\left(\frac{n}{2}\right) + cn\ ,$$

so:

$$\le 2k\left(\frac{n}{2}\right)\log\left(\frac{n}{2}\right) + cn\ \log\left(\frac{n}{2}\right) = (k + c)\,n\ \log\left(\frac{n}{2}\right) \le (k + c)\,n\ \log(n)$$

,

and now we take $\mathbf{k{+}c}$ to be our constant and we're done.

(You can assume that the base-case of T(1) is handled correctly, and to simplify things, n is always a power of 2, so that $\left\lfloor\frac{n}{2}\right\rfloor = \frac{n}{2} = \left\lceil\frac{n}{2}\right\rceil$ everywhere)

○ Yes, this approach is correct.

○ No, this approach is not correct.

Question 3

1 pts

What are the tightest bounds for the functions T_1 and T_2 (respectively from left to right) such that:

$$T_1(n) = T_1\left(\frac{3n}{4}\right) + cn$$

$$T_2(n) = T_2(n - 4) + cn$$

○ O(n), O(n^2)

○ O(n), O(n)

○ O(n logn), O(n logn)

○ O(n), O(n logn)

Question 4

1 pts

Can the merging step in “*Counting Inversions*” problem with n integers, be done in $O(n)$ time? Why?

○ No. When merging, we should sort all the values form the first and second half of elements. This takes time $O(nlogn)$.

○ No. When merging, we should first find the minimum of the smallest elements in the first half of elements and second half of elements. Then we should compare this value with all the elements in the opposite set. (If minimum is in the first half, it should be compared to all the values in the second half.) This takes time $O(n^2)$.

○ Yes. When merging, we compare the smallest value of first half of elements with the smallest value of the second half of elements and, add the minimum of these two to the output and erase if from its list. This takes time $O\left(n\right)$.

○ Yes. When merging, we add the values of first half of elements to the output first. Then we add the values of second half of elements. This takes time $O(n)$.

Not saved

Submit Quiz