

Sorting: Homework 2

2/4/2020

1. Generalize the SELECT algorithm to deal also with repeated values and prove that it still belongs to $O(n)$.
2. Download the latest version of the code from

https://github.com/albertocasagrande/AD_sorting

and

- Implement the SELECT algorithm of Ex. 1.
 - Implement a variant of the QUICK SORT algorithm using above-mentioned SELECT to identify the best pivot for partitioning.
 - Draw a curve to represent the relation between the input size and the execution-time of the two variants of QUICK SORT (i.e, those of Ex. 2 and Ex. 1 31/3/2020) and discuss about their complexities.
3. (Ex. 9.3-1 in [1]) In the algorithm SELECT, the input elements are divided into chunks of 5. Will the algorithm work in linear time if they are divided into chunks of 7? What about chunks of 3?
 4. (Ex. 9.3-5 in [1]) Suppose that you have a “black-box” worst-case linear-time subroutine to get the position in A of the value that would be in position $n/2$ if A was sorted. Give a simple, linear-time algorithm that solves the selection problem for an arbitrary position i .
 5. Solve the following recursive equations by using both the recursion tree and the substitution method:
 - $T_1(n) = 2 * T_1(n/2) + O(n)$
 - $T_2(n) = T_2(\lceil n/2 \rceil) + T_2(\lfloor n/2 \rfloor) + \Theta(1)$
 - $T_3(n) = 3 * T_3(n/2) + O(n)$
 - $T_4(n) = 7 * T_4(n/2) + \Theta(n^2)$

References

- [1] T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein. *Introduction to Algorithms*. The MIT Press. MIT Press, 2009.