CSCI 3110 Tutorial 5

Reviewed June 14, 2019

1. Consider the task of sorting an unsorted array $A[1 \ldots n]$: a task we can perform by merge sort in time $\mathrm{O}(n \lg n)$. Show that any algorithm that accesses the array only via comparisons (that is, by asking questions of the form "is $A[i] \leq z$?"), must take $\Omega(n \lg n)$ steps.

2. Given a sequence of real numbers, $X = (x_1, x_2, \ldots, x_n)$, an *exchanged pair* in X is a pair (x_i, x_j) such that $i < j$ and $x_i > x_j$. Note that an element x_i can be part of up to $n-1$ exchanged pairs. In particular, the maximal possible number of exchanged pairs in X is $\frac{n(n-1)}{2}$, which is achieved if the array is sorted in descending order. Assume that the sequence X is stored in an array of size n. Develop a divide-and-conquer algorithm that counts the exchanged pairs in X. (Your algorithm should take $\mathrm{O}(n \lg n)$ time.) Prove that your algorithm is correct. Argue briefly why your algorithm takes $\mathrm{O}(n \lg n)$ time.