Technische Universität München
Institut für Informatik
Theoretische Informatik

WS 2015/16
Worksheet 4
9.11.2015

Fundamental Algorithms 4

Exercise 1

Try the Recursion Tree Method (compare lecture) for the following recurrence:

$$T(n) = T(n/3) + T(2n/3) + O(n)$$

Show that the height of the recursion tree is in $O(\log(n))$.

What could be a flaw using the recursion tree method for such unbalanced trees? Show that $T(n) \in O(n \log(n))$, anyway, by using the substitution method.

Exercise 2

For the so-called BFPRT Algorithm, an algorithm to determine the *median* element of an array, we obtain the following (slightly simplified) recurrence equation for its running time $T(n)$ (depending on the number n of elements):

$$T(n) = s\left(n,k\right) + T\left(\frac{n}{k}\right) + T\left(\frac{l}{2k}n\right).$$

k and l are parameters (k usually small, for example $k = 3$ or $k = 5$) where $k = 2l + 1$. For the function s, we can assume $s(n,k) \in \Theta(n \log k)$.

a) Show that $T(n) \in O(n)$.

b) Does it make sense to use large values for k (and l, resp.)?