# Algorithm hw5

学号：U201514632
姓名：王宏斌

### For the divide and conquer algorithm of finding the k-th minimum element in an array of n elements. The complexity is T(n)=T(3/4 n)+ T(1/5 n)+ cn . Prove that T(n)=O(n).

> Proof with recursion-tree:
>
> - The longest path from the root to a leaf:
>
> $$n->(1/5)n->(1/5)^2n->...->1$$
>
> $$(1/5)^i n=1 <=> i=log_5 n$$
>
> - Cost of the problem at level is
>
> $$c*(19/20)^i n$$
>
> - Total cost:
>
> $$W(n)=c*\sum_{i=0}^{log_5 n}(19/20)^i n<cn*1/(1-19/20)=20cn$$
>
> Thus W(n) = O(n)

### Show the complexity for T(n)=3 T(1/3 n) + cn, and provide a proof.

According to Master Theory, the complexity for T(n)=3 T(1/3 n) + cn is obviously O(nlgn)

Proof with recursion-tree:

- Subproblem size at level i is:

$$n/3^i$$

- At level i: Cost of each node is:

$$cn/3^i$$

- h = Height of the tree ->

$$n/3^h = 1$$

$$h = log_3 n$$

- Total cost:

$$W(n) = \sum_{i=0}^{log_3 n} cn = (log_3 n + 1)n$$

Thus, W(n) = O(nlogn)