

CS375 Small Assignment 5 Solutions

Due Oct 9th 10PM ET

Use the recursion tree method to determine the runtime of the following functions. You can use the Master Theorem to check your answer for correctness:

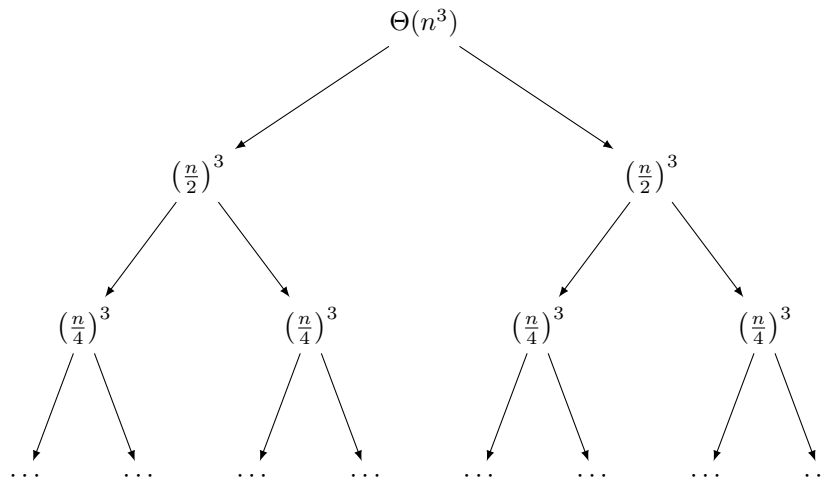
Master Theorem

Suppose $T(n) = aT(\frac{n}{b}) + \Theta(f(n))$:

- a) if there exists an $\epsilon > 0$ such that $f(n) = O(n^{\log_b(a) - \epsilon})$, then $T(n) = \Theta(n^{\log_b(a)})$
- b) if there exists a $k \geq 0$ such that $f(n) = \Theta(n^{\log_b(a)} \log^k n)$, then $T(n) = \Theta(n^{\log_b(a)} \log^{k+1} n)$
- c) if there exists an $\epsilon > 0$ such that $f(n) = \Omega(n^{\log_b(a) + \epsilon})$ and there exists a $c < 1$ such that $af(\frac{n}{b}) < cf(n)$, then $T(n) = \Theta(f(n))$.

1. $T(n) = 2T(\frac{n}{2}) + \Theta(n^3)$

Solution:

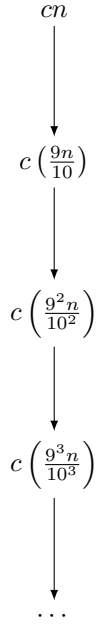


Note that the total work on the first level is n^3 , and $\frac{1}{4}n^3$ on the second, and $\frac{1}{16}n^3$ on the third, and so on. That is, every level down the work shrinks by a factor of $\frac{1}{4}$. As the input size decreases at a factor of $\frac{1}{2}$ per level of the tree, this means there is $\Theta(\log_2(n))$ levels

to the tree, giving a combined work of $\sum_{i=0}^{\log_2(n)} \frac{1}{4^i} n^3 = n^3 \sum_{i=0}^{\log_2(n)} \frac{1}{4^i} = \Theta(n^3)$. Further, note that $f(n) = n^3 = \Omega(n^3) = \Omega(n^{\log_2(2)+2}) = \Omega(n^{\log_b(a)+\epsilon})$ for $\epsilon = 2$, and $af(\frac{n}{b}) = 2f(\frac{n}{2}) = 2(\frac{n}{2})^3 = 2\frac{1}{8}n^3 = \frac{1}{4}n^3 \leq cf(n)$ for $c = \frac{1}{4}$, which confirms by the master theorem that $T(n) = \Theta(f(n)) = \Theta(n^3)$.

2. $T(n) = T(9n/10) + \Theta(n)$

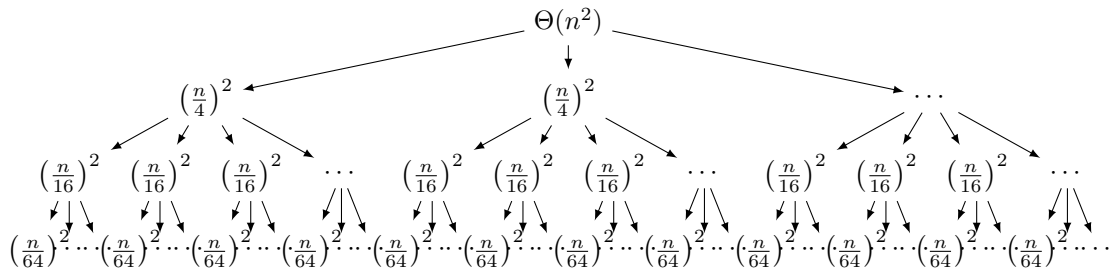
Solution:



The analysis for this one is basically the same as from a.

3. $T(n) = 16T(\frac{n}{4}) + \Theta(n^2)$

Solution:



This time, the work on the top level is n^2 . On the next level down, there are 16 copies of $\frac{n^2}{4^2} = \frac{n^2}{16}$, meaning the total work is *still* n^2 . So the work per level is constant, so we just need the number of levels, which is $\log_4(n)$. Hence the runtime is $\Theta(n^2 \log(n))$, which is consistent with the master theorem as $f(n) = n^2 = \Theta(n^2 \log^0(n)) = \Theta(n^{\log_4(16)} \log^0(n))$.