Final Exam Pre-Reading Problem Sample Solution

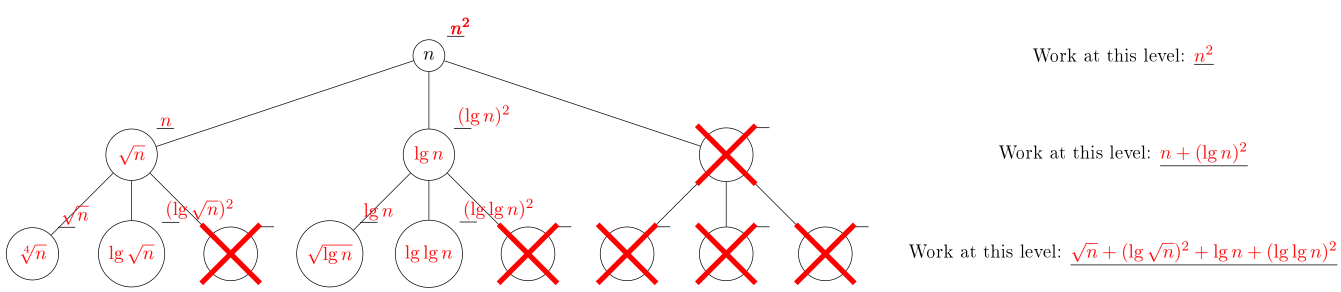
December 8, 2017

In this problem, you complete recurrence trees for two recurrence relations, writing problem size and work in terms of n. Specifically, we give you the first two levels of a recurrence tree with **more nodes than needed** for the recurrence. You should:

1. complete only the leftmost nodes needed at each level,
2. write **problem size inside the node**,
3. write the **amount of work at that node in the blank** above-right of the node,
4. write the **work per level in the blank on the far right** of each level,
5. put an **X through nodes that are not needed**, and finally
6. write a **good big-O bound on the height of the tree** below the tree.

Since the root node has problem size n regardless of the form of the recurrence, we have filled that node in for you. Now, complete trees for these two recurrences:

$$T(n) = \begin{cases} 1 & \text{when } n \leq 2 \\ T(\sqrt{n}) + T(\lg n) + n^2 & \text{otherwise} \end{cases}$$



Height of the tree $\in O(\lg \lg n)$.

(Side notes: **WE CORRECTED THE BASE CASE** to $n \leq 2$. Next, finding the height of this tree would be a **very** hard problem. Fortunately, we know that repeated application of lg will reach the base case faster than repeated appliaction of $\sqrt{\ }$; so, we can focus our attention on the $\sqrt{\ }$ branch. Further, we already figured out on an assignment the height of such a branch. Finally, just for notational interest, note that $\lg \sqrt{n} = \frac{1}{2} \lg n$ and for some notational uses, $\lg \lg n = \lg^{(2)} n$ which is not the same as $\lg^2 n = (\lg n)^2$.)