CS 305
Design and Analysis of Algorithms

09 / 16 / 2024

Instructor: Michael Eckmann

Today's Topics

- Questions/Comments?
- Recall perfect a-ary tree and how many leaves there are - express the number of leaves in a different way (using the log facts presented last class)
- Add up the work in the perfect a-ary tree
- Leads to the T(n) formula for time at leaves + time at internal nodes
- Master theorem / method
  - Prove case 1 of Master Theorem

Recurrence Relations

- A divide and conquer algorithm that creates *a* subproblems each a factor of *1/b* the size of the original problem and takes *f(n)* amount of time to do the divide and combine steps.
- T(n) = a * T(n/b) + f(n)
  - Number of recursive calls is **a**
  - **n/b** is size of list in a recursive call
  - The time in a call (excluding the recursion) is **f(n)**

Logarithms

FROM SECTION 3.3.

1. $b^{\log_b a} = a$

2. $\log_b(a \cdot c) = \log_b a + \log_b c$

3. $\log_b(a^x) = x \log_b a$ (follows from 2.)

4. $\log_b a = \dfrac{\log_c a}{\log_c b}$ for some other base c for converting bases.

5. $\log_b\left(\frac{1}{a}\right) = \log_b 1 - \log_b a = -\log_b a$

6. $a^{\log_b n} = n^{\log_b a}$

Logarithms

- All bases of logs must be real, > 0 and != 1.
- x used as an exponent in #3 must be real.
- All others must be real and > 0.

Logarithms

- Let me prove to you that the change of base formula is true.

Logarithms

- Let's go back and look at the a-ary tree and recall there are a^($\log_b n$) --- that is a to the power $\log_b n$ number of leaves in the tree
- From log properties #6 this is = n^($\log_b a$)
- Let's add up all the work done in this tree

Exponents

- Let me show some identities of exponents.

Master Method for Recurrences

- The Master Method can be used to solve most recurrences of the form: $T(n) = a * T(n/b) + f(n)$
- There are 3 cases to consider --- each based on a relationship of time spent at leaves vs. at the root.
- As an example, and to help us shortly, let's now count the number of nodes in a perfect tree, say a 3-ary tree.