CS 173 Lecture 14a: Recursion Trees

Recall recursively defined functions

Unrolling: Guessing closed forms

Recursion trees:
- visualizing recursively defined fn's
- another way of guessing closed forms

$A(4) = c$ base case

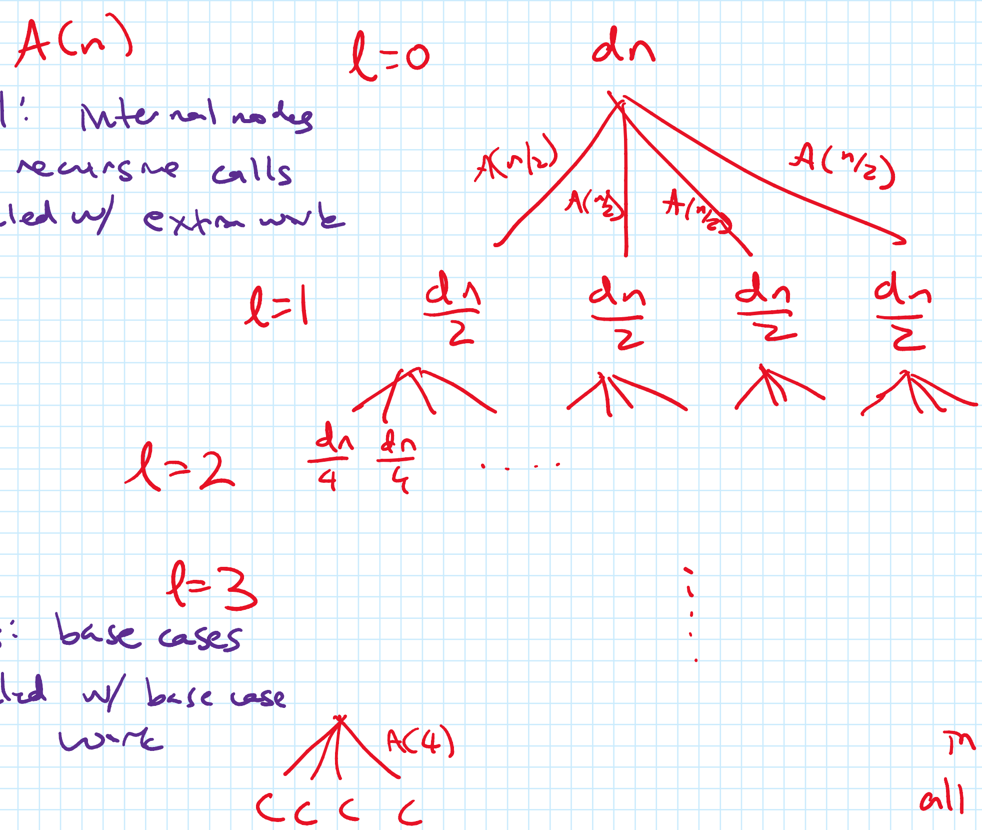
$A(n) = \underbrace{4A(n/2)}_{\text{recursive calls}} + \underbrace{dn}_{\text{extra work}}$ recursive case

often recursive fn's model running time (or "work") of recursive alg

root: $A(n)$, labeled w/ extra work

in general: internal nodes are recursive calls labeled w/ extra work

leaves: base cases labeled w/ base case work



total extra work?

dn

$2dn$

$4dn$

$8dn$

In this case all leaves are at $l = h$.

$$A(n) = \sum_{l=0}^{h-1} \text{total "extra work" at level } l$$

+ work at the leaves.

height? $\quad \frac{n}{2^h} = 4 \iff h = (\log_2 n) - 2.$

leaves? $\quad$ At level ℓ, # nodes is 4^ℓ

$\quad$ # leaves $= 4^h$

$$A(n) = dn \underbrace{\sum_{\ell=0}^{h-1} 2^\ell} + \underbrace{c\,4^h}$$

$$\sum_{\ell=0}^{h-1} a^\ell = \frac{a^h - 1}{a-1}$$

$$4^h = 4^{(\log_2 n)-2} = \frac{4^{\log_2 n}}{16}$$

$$\log_b c = \frac{\log_a c}{\log_a b}$$

$$\log_2 n = \log_2 4 \cdot \log_4 n$$

$$= dn\,\frac{2^{(\log_2 n)-2} - 1}{2-1} + \frac{c\,4^{\log_2 n}}{16}$$

$$= dn\left(\frac{n}{4} - 1\right) + \frac{c\,n^{\log_2 4}}{16}$$

$$= dn\left(\frac{n}{4} - 1\right) + \frac{c\,n^2}{16}.$$

Prove this correct: induction