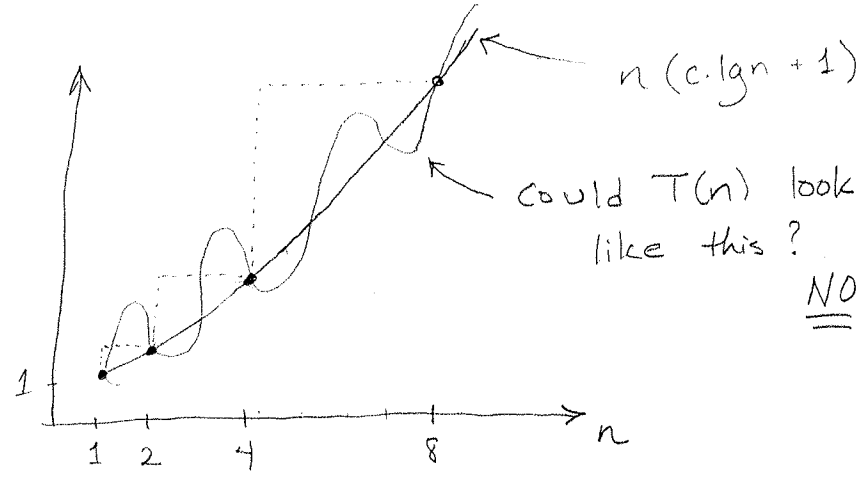


What about when $n \neq 2^k$?



① $T(n)$ is an increasing function: $T(n) \leq T(n+1)$ for all $n \in \mathbb{N}$
Proof (by induction)

base: $T(1) = 1 \leq T(2) = 2 + 2c$ (for $c \geq 0$)

$$T(n+1) = T(\lceil \frac{n+1}{2} \rceil) + T(\lfloor \frac{n+1}{2} \rfloor) + c(n+1) \quad c \geq 0$$

$n \geq \lceil \frac{n+1}{2} \rceil \geq \lfloor \frac{n}{2} \rfloor$ and I.H.

$n \geq \lfloor \frac{n+1}{2} \rfloor \geq \lfloor \frac{n}{2} \rfloor$ and I.H.

$$T(n+1) \geq T(\lfloor \frac{n}{2} \rfloor) + T(\lfloor \frac{n}{2} \rfloor) + cn = T(n)$$

because $2^{\lceil \lg n \rceil}$ is a power of 2 so our previous exact analysis holds.

$$\textcircled{1} \Rightarrow T(n) \leq T(2^{\lceil \lg n \rceil}) = 2^{\lceil \lg n \rceil} (c \lceil \lg n \rceil + 1)$$

Smallest power of 2 that is $\geq n$

$$\begin{aligned}
 &< 2^{(\lg n + 1)} (c(\lg n + 1) + 1) \\
 &= 2n (c \lg n + c + 1) \\
 &= 2c n \lg n + 2(c+1)n \\
 &\leq d n \lg n \quad \text{for } n \geq 2 \\
 &\quad d = 4c + 2
 \end{aligned}$$

Pitfalls

(A) Let $T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + n$, $T(1) = 1$

Wrong Guess: $T(n) \in O(n)$ i.e. $T(n) \leq c \cdot n$ for some $c > 0$ and all $n \geq n_0$

Verify by induction:

Inductive Step $T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + n$

$$\leq c \lceil n/2 \rceil + c \lfloor n/2 \rfloor + n \text{ by Ind. Hyp}$$

Note: $\lceil n/2 \rceil + \lfloor n/2 \rfloor = n$
for all $n \in \mathbb{N}$

$$= cn + n$$

$$= (c+1)n$$

$$\leq d \cdot n \text{ for } d \geq c+1 \text{ so } T(n) \in O(n)$$

Error! We did not prove $n \geq n_0$

the exact inductive hypothesis
that $T(n) \leq c \cdot n$

We cannot change c .

(B) Let $T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 1$, $T(1) = 1$

Guess $T(n) \leq c \cdot n$

Verify by induction

$$T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 1$$

$$\leq c \lceil n/2 \rceil + c \lfloor n/2 \rfloor + 1$$

$$= cn + 1 \text{ oops!}$$

Sometimes a stronger inductive hypothesis works.

Guess $T(n) \leq c \cdot n - 1$

$$T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 1$$

$$\leq c \lceil n/2 \rceil - 1 + c \lfloor n/2 \rfloor - 1 + 1$$

$$= cn - 1$$

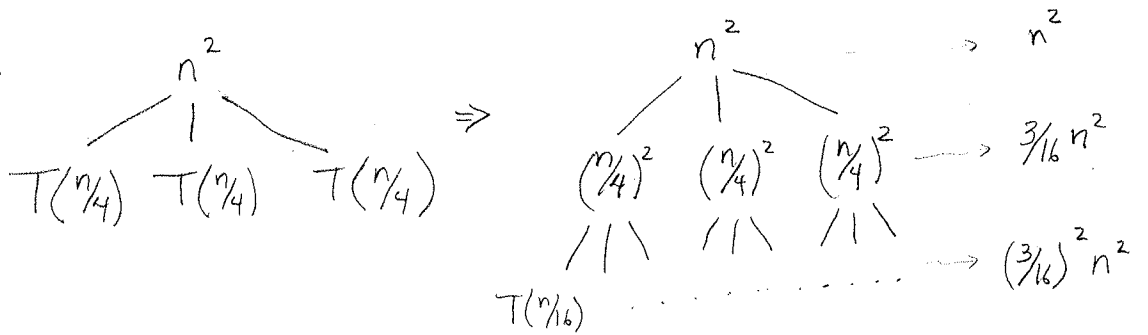
Base case $T(1) = 1 \leq c \cdot 1 - 1$ for $c \geq 2$

Recursion Tree method

- Use a tree to picture the expansion of a recurrence relation
 - record the nonrecursive contribution (cost) at a node
 - record the recursive contribution(s) in the node's subtrees

Example $T(n) = 3T(n/4) + n^2$, $T(1) = 1$

Assume n is a power of 4.



- Calculate the cost done on each level

i^{th} level $\rightarrow (3/16)^i n^2$

- Add up all cost on all levels

$$\text{Total} = \sum_{i=0}^{\log_4 n - 1} (3/16)^i n^2 + \underbrace{3^{\log_4 n}}_{\text{leaves } T(1)=1}$$

Since $x^{\log_b y} = y^{\log_b x}$

$$\log_b (x^{\log_b y}) = \log_b (y^{\log_b x})$$

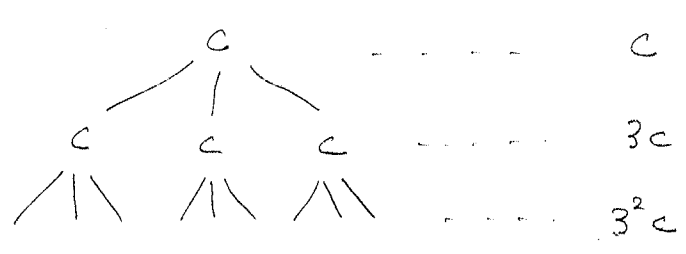
$$\log_b y \log_b x = \log_b x \log_b y$$

$$\begin{aligned} &< \sum_{i=0}^{\infty} (3/16)^i n^2 + 3^{\log_4 n} \\ &= n^2 \left(\frac{1}{1 - 3/16} \right) + 3^{\log_4 n} \\ &= \frac{16}{13} n^2 + 3^{\log_4 n} = \frac{16}{13} n^2 + n^{\log_4 3} \end{aligned}$$

$\Rightarrow T(n) \in O(n^2)$

- Verify using induction [if you're not sure]

Example 2 $T(n) \leq 3T(\lfloor n/2 \rfloor) + c$, $T(1) \leq d$ (14)



Assume n is a power of 2

number of levels before we reach leaves

$$= \sum_{i=0}^{\lfloor \lg n \rfloor - 1} 3^i \cdot c + d \cdot 3^{\lfloor \lg n \rfloor}$$

cost at leaves

$$= c \cdot \frac{3^{\lfloor \lg n \rfloor} - 1}{3 - 1} + d \cdot 3^{\lfloor \lg n \rfloor}$$

$$< c \cdot 3^{\lg n} + d \cdot 3^{\lg n}$$

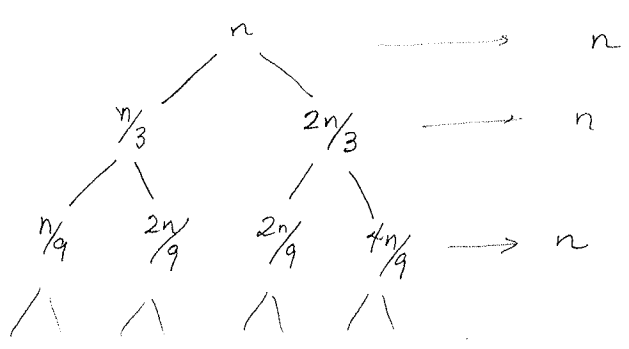
$$= (c+d) n^{\lg 3} \in O(n^{\lg 3})$$

Remember

$$\sum_{i=0}^k a^i = \frac{a^{k+1} - 1}{a - 1}$$

Example 3

$$T(n) = T(\lfloor n/3 \rfloor) + T(\lfloor 2n/3 \rfloor) + n, T(1) = 1$$



Let's suppose floor $\lfloor \rfloor$ doesn't matter

NOTE: Tree is not balanced (ie some root-to-leaf paths are longer than others)

max number of levels before we reach leaves

$$\leq \sum_{i=0}^{\log_{3/2} n - 1} n$$

cost at leaves

$$\leq n \log_{3/2} n +$$

total cost at internal nodes of tree

$$\leq c \cdot n \log n$$

this is our guess

$$\leq 2 \log_{3/2} n = \log_{3/2} 2^n = n^{1.7...}$$

That's too big

Verifying $T(n) \in O(n \log n)$

i.e. $T(n) \leq c \cdot n / \log_3 n$ for some c and $n \geq n_0$

← choosing base 3 is convenient

Ind. Step

$$\begin{aligned}
T(n) &= T(\lfloor n/3 \rfloor) + T(\lfloor \frac{2n}{3} \rfloor) + n \\
&\leq c \lfloor n/3 \rfloor \log_3 \lfloor n/3 \rfloor + c \lfloor \frac{2n}{3} \rfloor \log_3 \lfloor \frac{2n}{3} \rfloor + n \\
&\leq c (n/3) \log_3 (n/3) + c (\frac{2n}{3}) \log_3 (\frac{2n}{3}) + n \\
&\leq c \frac{n}{3} (\log_3 n - 1) + c (\frac{2n}{3}) (\log_3 n - \log_3 \frac{3}{2}) + n \\
&\leq c \frac{n}{3} \log_3 n - c \frac{n}{3} + c \frac{2n}{3} \log_3 n + n \\
&= c n \log_3 n - c \frac{n}{3} + n \quad \text{true for } c \geq 3
\end{aligned}$$

Base case then holds for $n \geq n_0 = 3$