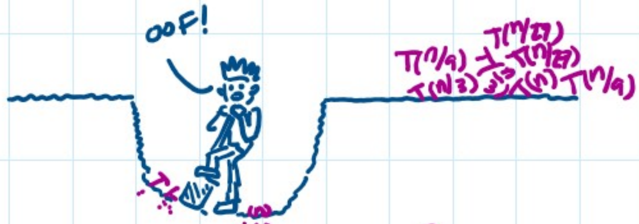
CMSC351-2023-09-29

Friday, September 29, 2023 8:36 AM



DIGGING DOWN - ONE MORE EXAMPLE!

$T(n) = 2T\left(\frac{n}{3}\right) + n$ ⊛ w/ $T(1) = 7$

we get

⊛ Think $T(■) = 2T\left(\frac{■}{3}\right) + ■$

■ = anything except 1.

$T(n) = 2T\left(\frac{n}{3}\right) + n$ ⊛ w/ $T\left(\frac{n}{3}\right)$

$= 2\left[2T\left(\frac{n/3}{3}\right) + \frac{n}{3}\right] + n$

$= 2^2 T\left(\frac{n}{3^2}\right) + \frac{2}{3}n + n$ ⊛ w/ $T\left(\frac{n}{3^2}\right)$

$= 2^2\left[2T\left(\frac{n}{3^3}\right) + \frac{n}{3^2}\right) + \frac{2}{3}n + n$

$= 2^3 T\left(\frac{n}{3^3}\right) + \frac{2^2}{3^2}n + \frac{2}{3}n + n$ ← nicer un-summed!

$= \dots$ what's the general expression

$= 2^k T\left(\frac{n}{3^k}\right) + \sum_{i=0}^{k-1}\left(\frac{2}{3}\right)^i n$

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

ends when $\frac{n}{3^k} = 1$ so $3^k = n$ so $k = \log_3 n$

then

$T(n) = 2^{\log_3 n} T(1) + n\sum_{i=0}^{k-1}\left(\frac{2}{3}\right)^i$ ← correct but ☹

correct but icky! Can't see Θ

$T(n) = 2^{\log_3 n}(7) + n\left[\frac{\left(\frac{2}{3}\right)^k - 1}{\frac{2}{3} - 1}\right] = 2^{\log_3 n}(7) - 3n\left[\left(\frac{2}{3}\right)^{\log_3 n} - 1\right]$

now: $2^{\log_3 n} = 2^{\frac{\log_2 n}{\log_2 3}} = \left(2^{\log_2 n}\right)^{1/\log_2 3}$

$= n^{\log_3 2}$

Log C.O.B:

$$\log_a b = \frac{\log_c b}{\log_c a}$$

so

$T(n) = 7n^{\log_3 2} - 3n\left[\frac{n^{\log_3 2}}{n} - 1\right]$

special case:

$$\log_a b = \frac{\log_b b}{\log_b a} = \frac{1}{\log_b a}$$

$T(n) = 7n^{\log_3 2} - 3n^{\log_3 2} + 3n$

$T(n) = 4n^{\log_3 2} + 3n$ ← oh how sweet it is!

note $\log_3 2 < 1$ so $T(n) = \Theta(n)$

Recurrence Trees

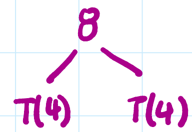
Recurrence Trees

① How might we analyze recurrence relns. differently?

ex: Consider $T(n) = 2T(\frac{n}{2}) + n$ w/ $T(1) = 3$

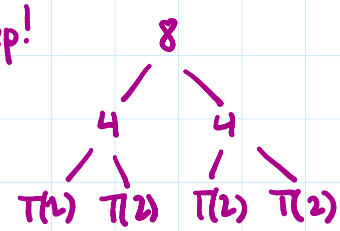
Let's find $T(8)$.

Hmmm... $T(8)$ ... let's draw a tree! First



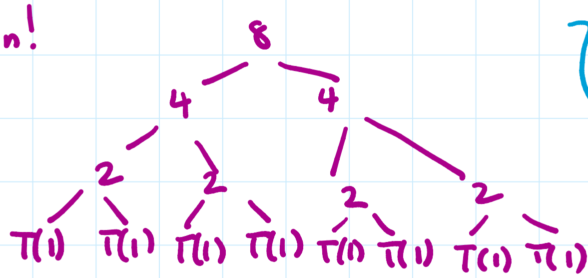
sum = T(8)

one more step!

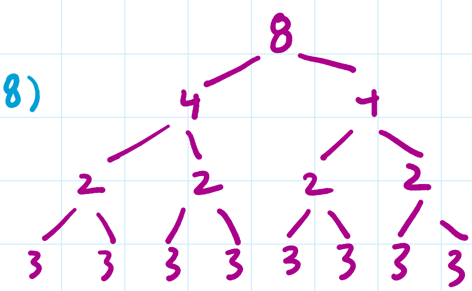


sum = T(8)

and again!



sum = T(8)



So $T(8) = 8+4+4+2+2+2+2+3+3+3+3+3+3+3+3 = \ldots$

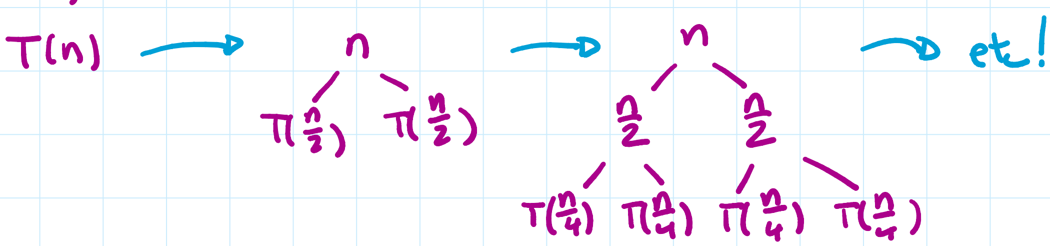
② what did we do? we used the R.R. to expand down until the leaves are $T(1)$, then add all the entries to get our $T(8)$.

This works neatly for specific numbers. How about for $T(n)$ in general?

we'll do the same thing just more carefully...

ex $T(n) = 2T(\frac{n}{2}) + n$ w/ $T(1) = 7$

Tree time!



this would go until the leaves are $T(1)$.

Think levels w/ level $i=0$ being the top

$i=1$ next down, etc

as we build, level i is initially $T(n/2^i)$ (each entry) i-1 next down, etc

as we build, level i is initially $T(n/2^i)$ (each entry)

ends when $n/2^i = 1$ so $i = \lg n$

Let's make a table!

level	# nodes	time/node	level total = (# nodes)(time/node)
0	1	n	n
1	2	$n/2$	n
2	4	$n/4$	n
⋮	⋮	⋮	⋮
$(\lg n)-1$	$2^{\lg n - 1}$	$n/2^{\lg n - 1}$	n
$\lg n$	$2^{\lg n}$	7	$7 \cdot 2^{\lg n}$

b/c leaves are $T(1)=7$

total sum $T(n)$ = sum of rightmost column!

$$T(n) = n(\lg n) + 7 \cdot 2^{\lg n}$$

$$T(n) = n\lg n + 7n = \Theta(n \lg n)$$