Árvore de recorrência
$T(n) = 4T(n/2) + n$

David Pierre
Mateus Fittipaldi
Otávio Alves

Use a árvore de recorrência para $T(n) = 4T(n/2) + n$ e obtenha a classe Θ a qual a solução pertence

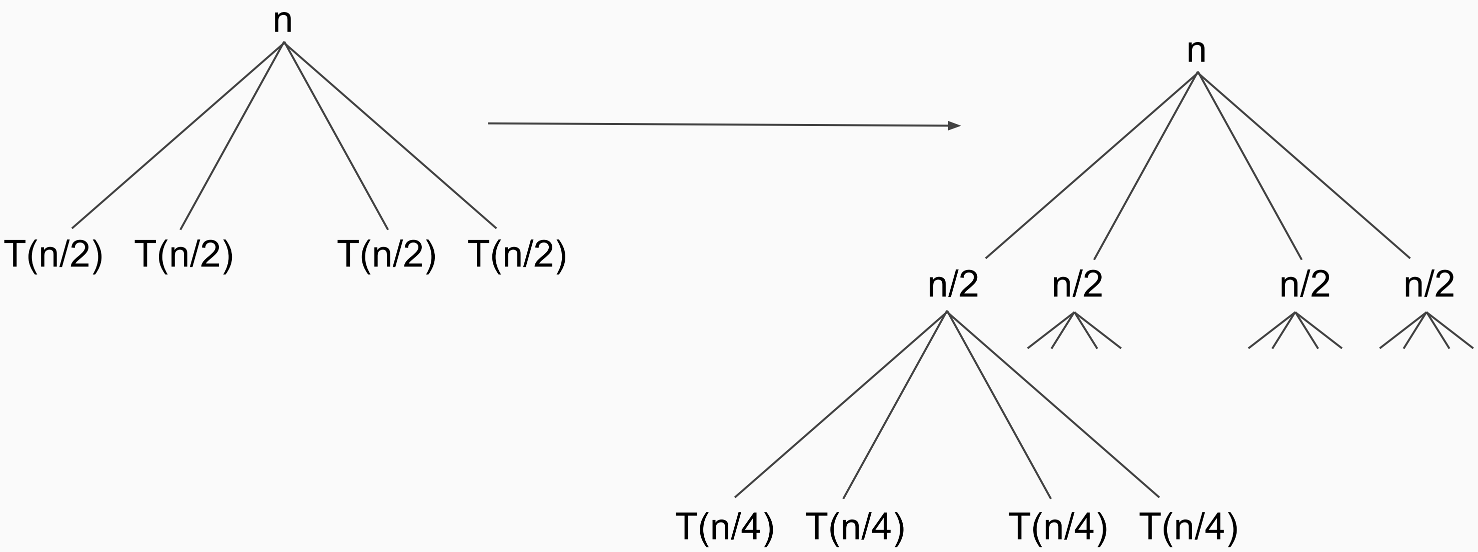
Passos:

1. Construir a árvore de recursão.
2. Determinar:
   a. Número de níveis.
   b. Custo por nível.
   c. Número de nós no último nível.
   d. Custo do último nível.
3. Faça o somatório do custo de todos os níveis para achar o custo de $T(n)$.

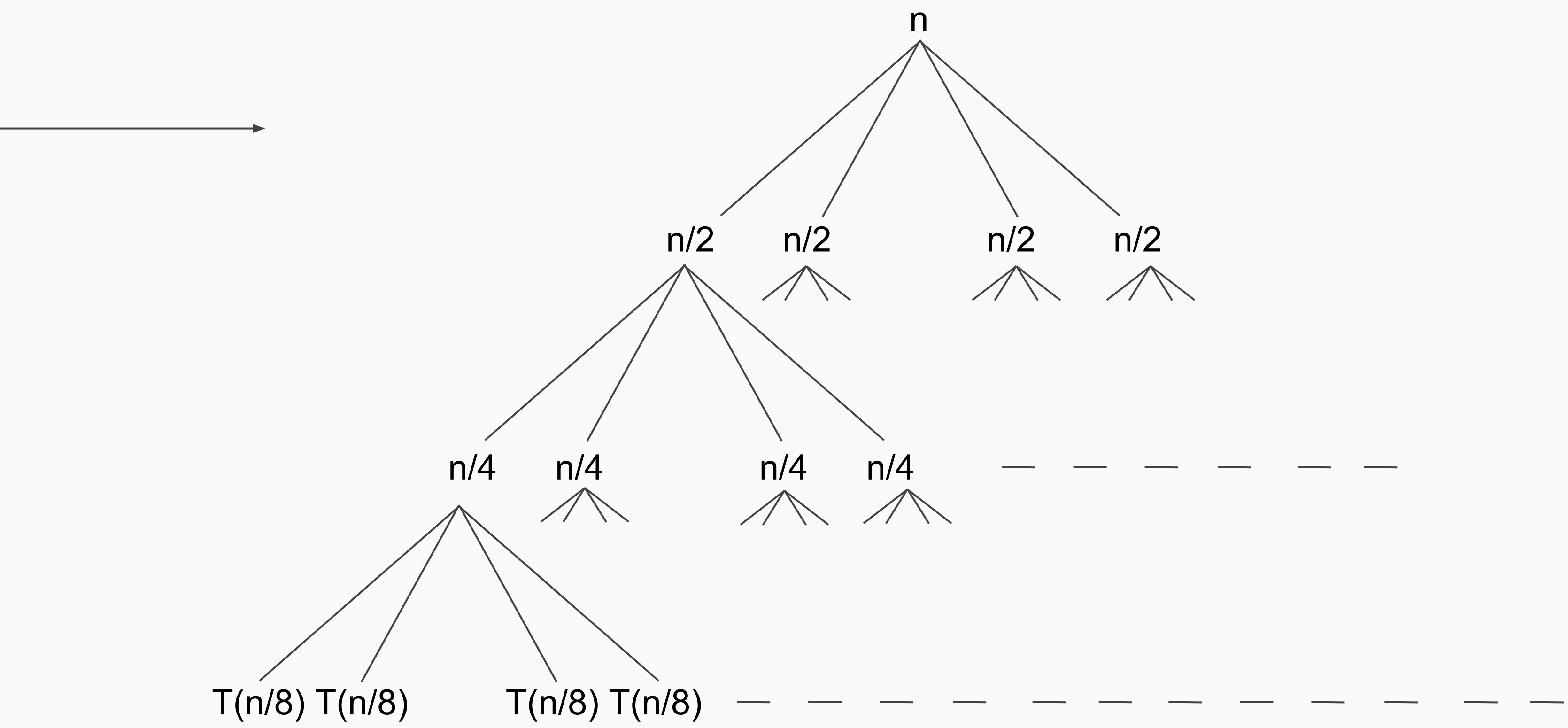
Construir a árvore de recursão

Considerando $T(n) = 4T(n/2) + n$:

$T(n/2) = 4T(n/4) + n/2$:



T(n/4) = 4T(n/8) + n/4:



Determinar: número de níveis, custo por nível, número de nós no último nível e custo do último nível

Nível	Num. de nós	Árvore de recursão	Custo do nível
0	1	n	n
1	4^1	n/2 n/2 n/2 n/2	4(n/2) = 2n
2	4^2	n/4 n/4 n/4 n/4	$4^2(n/4) = 4n$
3	4^3	n/8 n/8	$4^3(n/8) = 8n$
...	...	....	...
i	4^i	$n/2^i$	$4^i(n/2^i) = 2^i n$
h	4^h	T(1) T(1) T(1) T(1) T(1) T(1) T(1) T(1)	4^h

Somatório do custo de todos os níveis para achar o custo de T(n)

$$T(n) = 4^h\, T(1) + (n + 2n + 4n + 8n + \dots + 2^i n)$$

$$T(n) = 4^h\, T(1) + \sum_{i=0}^{h-1} 2^i n \longrightarrow \textit{(Eq.1*)}$$

$$T(n) = 4^{\log_2 n}\, T(1) + \sum_{i=0}^{h-1} 2^i n$$

$$T(n) = n^{\log_2 4}\, T(1) + \sum_{i=0}^{h-1} 2^i n$$

$$T(n) = n^2\, T(1) + \sum_{i=0}^{h-1} 2^i n \longrightarrow \textit{(Eq.2*)}$$

$$T(n) = n^2\, T(1) + n\,(n - 1)$$

$$T(n) = n^2 \,.\, 1 + n^2 - n$$

$$\mathbf{T(n) = 2n^2 - n = \Theta(n^2)}$$

Eq.1:

$n/2^h = T(1) = 1$

$n = 2^h$

$h = \log_2 n$

Eq.2:

$$\sum_{i=0}^{h-1} 2^i\, n = n \sum_{i=0}^{h-1} 2^i$$

Série:

$r = 2,\ a = 1,\ r > 1$

$$S = a + ar + ar^2 + \dots + ar^{k-1} = \frac{a(r^k - 1)}{r - 1}$$

$$\frac{1(2^h - 1)}{2 - 1} = 2^h - 1$$

$$2^h - 1 = 2^{\log_2 n} - 1$$

$$n^{\log_2 2} - 1 = n - 1 \Rightarrow n\,(n - 1)$$

Exercício:

$$T(n) = 5T(n/5) + n^3$$

Obrigado!