

Análise e Síntese de Algoritmos Divide & Conquer. Teorema Mestre. CLRS Cap. 4

Prof. Pedro T. Monteiro

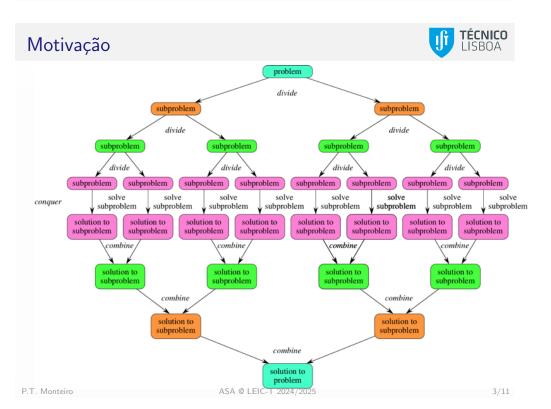
IST - Universidade de Lisboa

2024/2025

P.T. Monteiro

ASA @ LEIC-T 2024/2025

1/11



Contexto



- Revisão [CLRS, Cap.1-13]
 - Fundamentos; notação; exemplos
- Técnicas de Síntese de Algoritmos [CLRS, Cap.15-16]
 - Programação dinâmica [CLRS, Cap.15]
 - Algoritmos greedy [CLRS, Cap.16]
- Algoritmos em Grafos [CLRS, Cap.21-26]
 - Algoritmos elementares
 - Caminhos mais curtos [CLRS, Cap.22,24-25]
 - Árvores abrangentes [CLRS, Cap.23]
 - Fluxos máximos [CLRS, Cap.26]
- Programação Linear [CLRS, Cap.29]
 - Algoritmos e modelação de problemas com restrições lineares
- Tópicos Adicionais
 - Emparelhamento de Cadeias de Caracteres [CLRS, Cap.32]
 - Complexidade Computacional [CLRS, Cap.34]

P.T. Monteiro

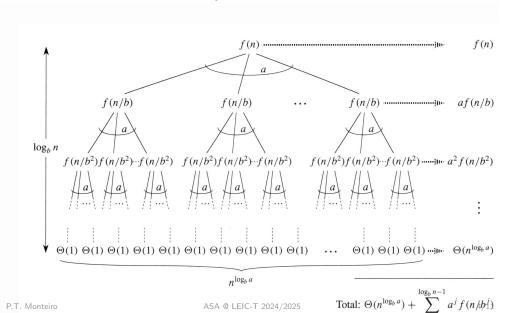
ASA @ LEIC-T 2024/2025

0/11

Recorrências



Recorrências divide-and-conquer



Teorema Mestre



Teorema Mestre

Sejam $a \ge 1, b > 1, d \ge 0$ constantes, e seja T(n) definido por

$$T(n) = aT(n/b) + \Theta(n^d)$$

T(n) é limitado assimptoticamente da seguinte forma:

$$T(n) = \begin{cases} \Theta(n^{\log_b a}) & \text{if } d < \log_b a \\ \Theta(n^d \log n) & \text{if } d = \log_b a \\ \Theta(n^d) & \text{if } d > \log_b a \end{cases}$$

- Em cada um dos 3 casos estamos a comparar d com log_b a
- Solução da recorrência é determinada pela maior das duas funções!

Definição do livro Papadimitriou. Definição alternativa no livro do Cormen!

P.T. Monteiro

ASA @ LEIC-T 2024/2025

5/11

Teorema Mestre



Série geométrica

$$\sum_{k=0}^{n} a.r^{k} = a.r^{0} + a.r^{1} + ... + a.r^{k} = a.\frac{1-r^{n+1}}{1-r}$$

Na notação assimptótica, a série converge para:

- Se r < 1 a série converge para O(a) (1º termo)
- Se r > 1 a série converge para $O(a.r^n)$ (último termo)

Teorema Mestre



Nível # Problemas Custo cada problema

0	$a^{0} = 1$	$O(n^d)$
1	a^1	$O((n/b^1)^d)$
2	a^2	$O((n/b^2)^d)$
3	a^3	$O((n/b^3)^d)$
k	a^k	$O((n/b^k)^d)$

Custo total no nível k

$$a^k \times O((\frac{n}{b^k})^d) = O(n^d) \times (\frac{a}{b^d})^k$$

À medida que k aumenta o nível de 0 (raíz) até $\log_b n$ (folhas), forma uma série geométrica com rácio $\frac{a}{b^d}$

P.T. Montei

ASA @ LEIC-T 2024/2025

Teorema Mestre



Custo total da recorrência

$$\sum_{k=0}^{log_b n} O(n^d) \times (\frac{a}{b^d})^k$$

(série geométrica)

Caso 1: $log_b a > d$

Se $\frac{a}{b^d}>1$, na série geométrica o último termo é o maior

$$\sum_{k=0}^{\log_b n} O(n^d) \times \left(\frac{a}{b^d}\right)^k$$

$$= O\left(O(n^d) \times \left(\frac{a}{b^d}\right)^{\log_b n}\right)$$

$$= O\left(O(n^d) \times \frac{a^{\log_b n}}{b^{d \log_b n}}\right)$$

$$= O\left(O(n^d) \times \frac{n^{\log_b a}}{n^d}\right)$$

$$= O(n^{\log_b a})$$

$$T(n) = O(n^{\log_b a})$$

Teorema Mestre



Custo total da recorrência

$$\sum_{k=0}^{\log_b n} O(n^d) \times (\frac{a}{b^d})^k$$

(série geométrica)

Caso 3: $log_b a < d$

Se $\frac{a}{b^d} < 1$, na série geométrica o $1^{\underline{o}}$ termo é o maior

$$\begin{array}{rcl} & \sum_{k=0}^{\log_b n} O(n^d) \times (\frac{a}{b^d})^k \\ = & O\left(O(n^d) \times 1\right) \\ = & O(n^d) \end{array}$$

$$T(n) = O(n^d)$$

P.T. Monteiro

ASA @ LEIC-T 2024/2025

9/11

Questões?



Dúvidas?

P.T. Monteiro ASA @ LEIC-T 2024/2025 1:

Teorema Mestre



Custo total da recorrência

$$\sum_{k=0}^{log_b n} O(n^d) \times (\frac{a}{b^d})^k$$

(série geométrica)

Caso 2: $log_b a = d$

Se $\frac{a}{b^d} = 1$, temos de calcular todo o somatório

$$\sum_{k=0}^{\log_b n} O(n^d) \times \left(\frac{a}{b^d}\right)^k$$

$$= \sum_{k=0}^{\log_b n} O(n^d) \times 1$$

$$= (1 + \log_b n) \times O(n^d)$$

$$= O(n^d \times \log_b n)$$

$$T(n) = O(n^d \log n)$$

P.T. Monteiro ASA @ LEIC-T 2024/2025 10/1: