Melhores momentos

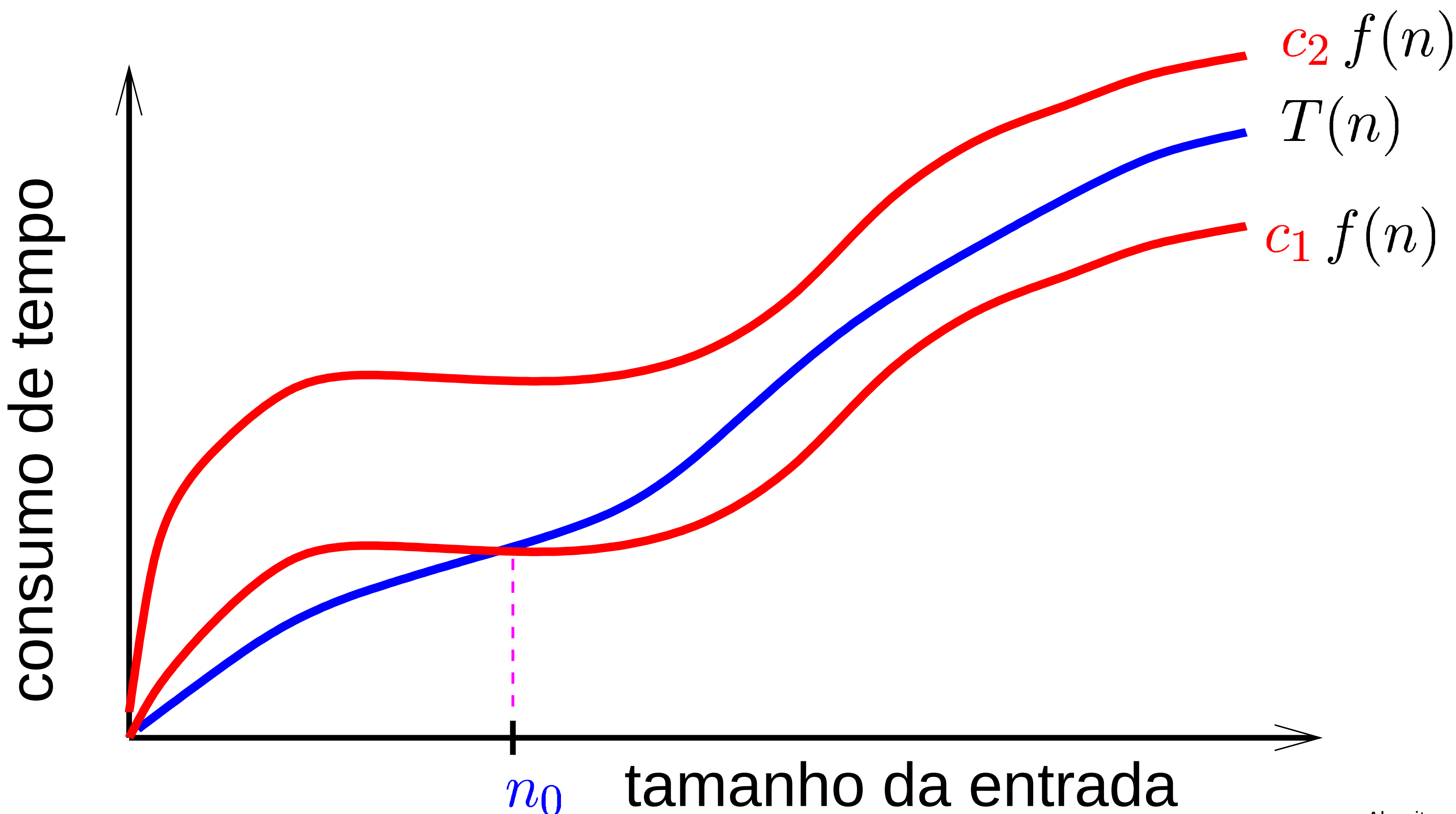
AULA 4

Definição

Dizemos que $T(n)$ é $\Theta(f(n))$ se se existem constantes positivas c_1, c_2 e n_0 tais que

$$c_1\, f(n) \;\leq\; T(n) \;\leq\; c_2\, f(n)$$

para todo $n \geq n_0$.



Análise da intercalação

Problema: Dados $A[p \,.\,.\, q]$ e $A[q{+}1 \,.\,.\, r]$ crescentes, rearranjar $A[p \,.\,.\, r]$ de modo que ele fique em ordem crescente.

Entra:

	p				q				r
A	22	33	55	77	99	11	44	66	88

Sai:

	p				q				r
A	11	22	33	44	55	66	77	88	99

Intercalação

INTERCALA (A, p, q, r)
$\triangleright$ $B[p\,..\,r]$ é um vetor auxiliar
para $i \leftarrow p$ **até** q **faça**
    $B[i] \leftarrow A[i]$
para $j \leftarrow q + 1$ **até** r **faça**
    $B[r + q + 1 - j] \leftarrow A[j]$
$i \leftarrow p$
$j \leftarrow r$
para $k \leftarrow p$ **até** r **faça**
    se $B[i] \leq B[j]$
        então $A[k] \leftarrow B[i]$
            $i \leftarrow i + 1$
        senão $A[k] \leftarrow B[j]$
            $j \leftarrow j - 1$

Consumo de tempo

Se a execução de cada linha de código consome 1 unidade de tempo o consumo total é ($n := r - p + 1$):

linha		todas as execuções da linha
1	$=$	$q - p + 2 \;=\; n - r + q + 1$
2	$=$	$q - p + 1 \;=\; n - r + q$
3	$=$	$r - (q + 1) + 2 \;=\; n - q + p$
4	$=$	$r - (q + 1) + 1 \;=\; n - q + p - 1$
5	$=$	1
6	$=$	1
7	$=$	$r - p + 2 \;=\; n + 1$
8	$=$	$r - p + 1 \;=\; n$
9–12	$=$	$2\,(r - p + 1) \;=\; 2n$
total	$=$	$8n - 2(r - p + 1) + 5 = 6n + 5$

Consumo de tempo

Quanto tempo consome em função de $n := r - p + 1$?

linha	consumo de todas as execuções da linha
1–4	$\Theta(n)$
5–6	$\Theta(1)$
7	$n\Theta(1) = \Theta(n)$
8	$n\Theta(1) = \Theta(n)$
9–12	$n\Theta(1) = \Theta(n)$
total	$\Theta(4n + 1) = \Theta(n)$

Conclusão:

> O algoritmo consome $\Theta(n)$ unidades de tempo.

Merge-Sort

Rearranja $A[p \,..\, r]$, com $p \leq r$, em ordem crescente.

MERGE-SORT (A, p, r)
se $p < r$
então $q \leftarrow \lfloor (p + r)/2 \rfloor$
MERGE-SORT (A, p, q)
MERGE-SORT $(A, q + 1, r)$
INTERCALA (A, p, q, r)

	p				q				r
A	55	33	66	44	99	11	77	22	88

Merge-Sort

Rearranja $A[p\,..\,r]$, com $p \leq r$, em ordem crescente.

MERGE-SORT (A, p, r)
    se $p < r$
        então $q \leftarrow \lfloor (p + r)/2 \rfloor$
            MERGE-SORT (A, p, q)
            MERGE-SORT $(A, q+1, r)$
            INTERCALA (A, p, q, r)

	p				q				r
A	33	44	55	66	99	11	77	22	88

Merge-Sort

Rearranja $A[p\,..\,r]$, com $p \leq r$, em ordem crescente.

MERGE-SORT (A, p, r)
  se $p < r$
    então $q \leftarrow \lfloor (p + r)/2 \rfloor$
      MERGE-SORT (A, p, q)
      MERGE-SORT $(A, q+1, r)$
      INTERCALA (A, p, q, r)

	p				q				r
A	33	44	55	66	99	11	22	77	88

Merge-Sort

Rearranja $A[p\,..\,r]$, com $p \leq r$, em ordem crescente.

```
MERGE-SORT (A, p, r)
    se p < r
        então q ← ⌊(p + r)/2⌋
              MERGE-SORT (A, p, q)
              MERGE-SORT (A, q + 1, r)
              INTERCALA (A, p, q, r)
```

	p				q				r
A	11	22	33	44	55	66	77	88	99

Merge-Sort

Rearranja $A[p\,..\,r]$, com $p \leq r$, em ordem crescente.

MERGE-SORT (A, p, r)
1 **se** $p < r$
2 **então** $q \leftarrow \lfloor (p+r)/2 \rfloor$
3 MERGE-SORT (A, p, q)
4 MERGE-SORT $(A, q+1, r)$
5 INTERCALA (A, p, q, r)

Consumo de tempo?

$T(n) :=$ consumo de tempo máximo quando $n = r - p + 1$

Merge-Sort

```
MERGE-SORT (A, p, r)
1    se p < r
2        então q ← ⌊(p + r)/2⌋
3              MERGE-SORT (A, p, q)
4              MERGE-SORT (A, q + 1, r)
5              INTERCALA (A, p, q, r)
```

linha	consumo na linha
1	$\Theta(1)$
2	$\Theta(1)$
3	$T(\lceil n/2 \rceil)$
4	$T(\lfloor n/2 \rfloor)$
5	$\Theta(n)$

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

Merge-Sort

$T(n)$:= consumo de tempo máximo quando $n = r - p + 1$

$$T(1) = \Theta(1)$$
$$T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n) \quad \text{para } n = 2, 3, 4, \ldots$$

Solução: $T(n)$ é $\Theta(???)$.

Demonstração: $\ldots$

Veremos, mas antes estudaremos Recorrências.

Recorrências

$$
\begin{aligned}
T(1) &= 1 \\
T(n) &= T(n-1) + 3n + 2 \quad \text{para } n = 2, 3, 4, \ldots
\end{aligned}
$$

Define função T sobre inteiros positivos:

n	1	2	3	4	5	6
$T(n)$	1	9	20	34	51	71

$T(n)$
1 **se** $n = 1$
2 **então devolva** 1
3 **senão devolva** $T(n-1) + 3n + 2$

Método da substituição

Chute:
Eu acho que $T(n) = \frac{3}{2}n^2 + \frac{7}{2}n - 4$.

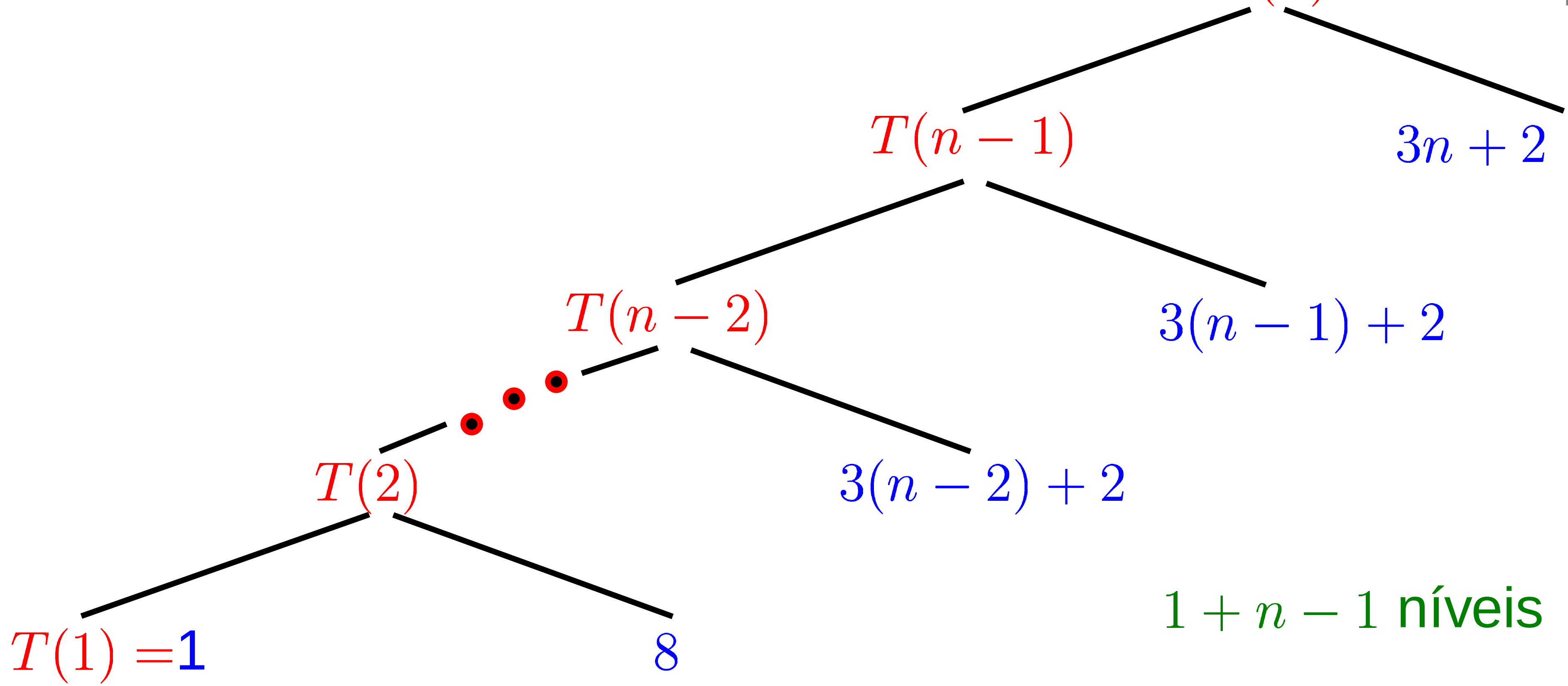
Verificação por indução:
Se $n = 1$ então $T(n) = 1 = \frac{3}{2} + \frac{7}{2} - 4$.
Tome $n \geq 2$ e suponha que a fórmula está certa para $n - 1$:

$$\begin{aligned} T(n) &= T(n-1) + 3n + 2 \\ &\stackrel{\text{hi}}{=} \tfrac{3}{2}(n-1)^2 + \tfrac{7}{2}(n-1) - 4 + 3n + 2 \\ &= \tfrac{3}{2}n^2 - 3n + \tfrac{3}{2} + \tfrac{7}{2}n - \tfrac{7}{2} - 4 + 3n + 2 \\ &= \tfrac{3}{2}n^2 + \tfrac{7}{2}n - 4\,. \end{aligned}$$

Bingo!

Como adivinhei fórmula fechada?

Árvore da recorrência:



$$T(n) = (3n+2) + (3n-1) + \cdots + 8 + 1$$

$$= \frac{3}{2}n^2 + \frac{7}{2}n - 4$$

Exemplo 3

$$\begin{aligned} G(1) &= 1 \\ G(n) &= 2G(n/2) + 7n + 2 \quad \text{para } n = 2, 4, 8, 16 \ldots, 2^i, \ldots \end{aligned}$$

n	1	2	4	8	16
$G(n)$	1	18	66	190	494

Fórmula fechada: $G(n) =$???

Resolvendo a recorrência

$G(n) = 7n\lg n + 3n - 2$ para $n = 1, 2, 4, 8, 16, 32\ldots$

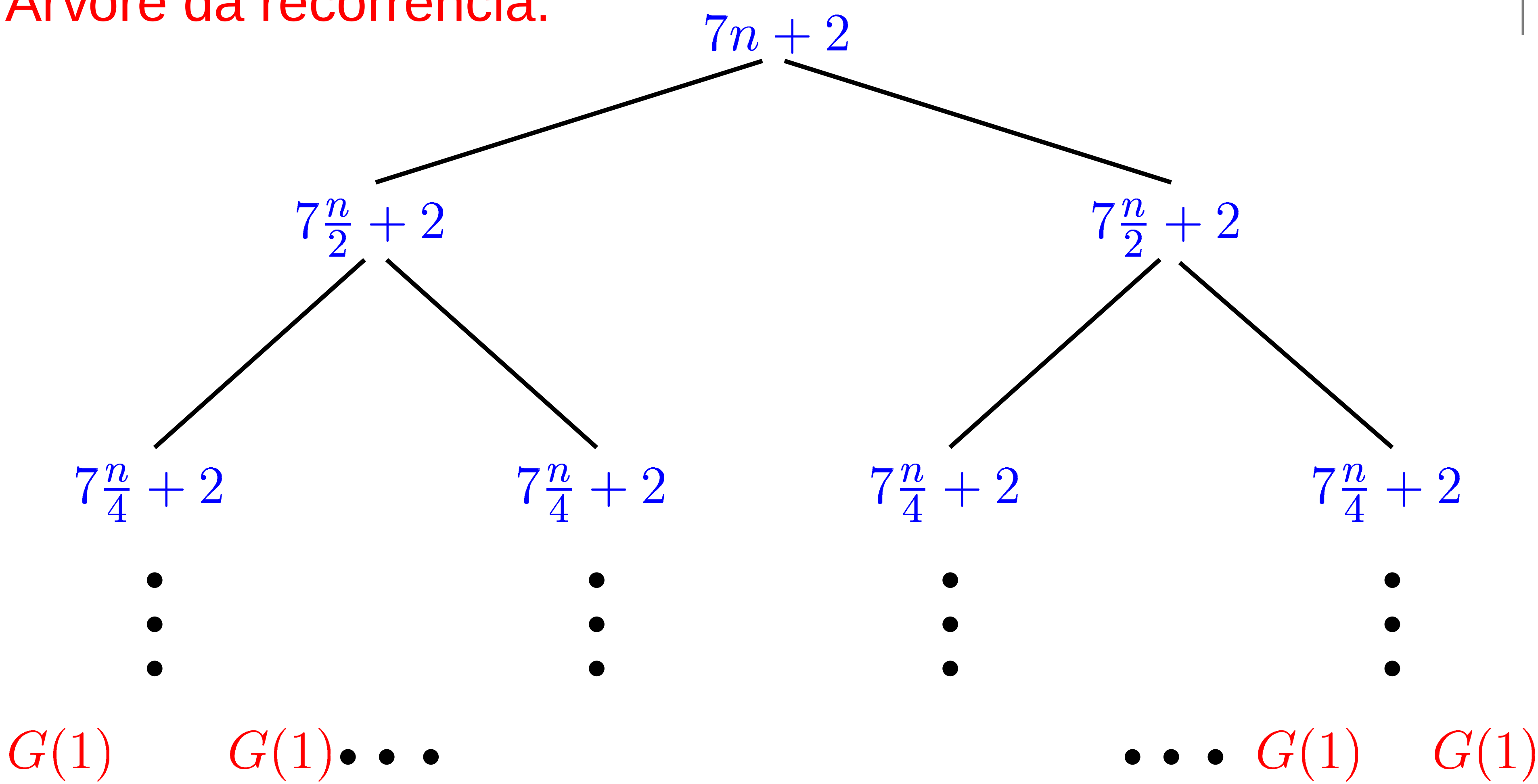
Prova: Se $n = 1$ então $G(n) = 1 = 7 \cdot 1 \lg 1 + 3 \cdot 1 - 2$.
Se $n \geq 2$ então

$$
\begin{aligned}
G(n) &= 2G(\tfrac{n}{2}) + 7n + 2 \\
&\overset{\text{hi}}{=} 2\left(7\tfrac{n}{2}\lg\tfrac{n}{2} + 3\,\tfrac{n}{2} - 2\right) + 7n + 2 \\
&= 7n(\lg n - 1) + 3n - 4 + 7n + 2 \\
&= 7n\lg n - 7n + 3n - 2 + 7n \\
&= 7n\lg n + 3n - 2
\end{aligned}
$$

Iiiiiéééésss!

Como adivinhei fórmula fechada?

Árvore da recorrência:



total de $1 + \lg n$ níveis

Contas

nível	0	1	2	$\ldots$	$k-1$	k
soma	$7n+2$	$7n+4$	$7n+8$	$\ldots$	$7n+2^k$	$2^k G(1)$

$n = 2^k \qquad k = \lg n$

$$
\begin{aligned}
G(n) &= 7n + 2^1 + 7n + 2^2 + \cdots + 7n + 2^k + 2^k G(1) \\
&= 7n\,k + (2 + 4 + \cdots + 2^k) + 2^k \\
&= 7n\,k + 2 \cdot 2^k - 2 + n \\
&= 7n \lg n + 2n - 2 + n \quad (k = \lg n) \\
&= 7n \lg n + 3n - 2 \qquad (0)
\end{aligned}
$$

Iiiiééééssss

Exemplo 3 (continuação)

É mais fácil mostrar que $G(n) = \mathrm{O}(n \lg n)$.
Vou provar que $G(n) \leq 9n \lg n$ para $n = 2, 4, 8, 16, \ldots, 2^i, \ldots$

Prova: Se $n = 2$, $G(n) = 18 = 9 \cdot 2 \cdot \lg 2$.
Se $n \geq 4$,

$$
\begin{aligned}
G(n) &= 2G(n/2) + 7n + 2 \\
&\overset{\text{hi}}{\leq} 2 \cdot 9(n/2) \lg(n/2) + 7n + 2 \\
&= 9n\,(\lg n - 1) + 7n + 2 \\
&= 9n \lg n - 2n + 2 \\
&< 9n \lg n \quad \text{(pois } n \geq 2\text{)}
\end{aligned}
$$

Da linha 1 para a linha 2, a hipótese de indução vale pois $2 \leq n/2 < n$.

Exemplo 3 (novamente)

É mais fácil mostrar que $G(n) = \mathrm{O}(n \lg n)$.
Vou provar que $G(n) \leq 8n \lg n$ para $n = 8, 16, \ldots, 2^i, \ldots$

Prova: Se $n = 8$, $G(n) = 190 < 192 = 8 \cdot 8 \cdot \lg 8 = 64 \cdot 3$.
Se $n \geq 16$,

$$
\begin{aligned}
G(n) &= 2G(n/2) + 7n + 2 \\
&\overset{\text{hi}}{\leq} 2 \cdot 8(n/2) \lg(n/2) + 7n + 2 \\
&= 8n\,(\lg n - 1) + 7n + 2 \\
&= 8n \lg n - n + 2 \\
&< 9n \lg n \quad \text{(pois } n \geq 16\text{)}
\end{aligned}
$$

Da linha 1 para a linha 2, a hipótese de indução vale pois $2 \leq n/2 < n$.

AULA 5

Recorrências (continuação)

CLRS 4.1–4.2

AU 3.9, 3.11

Exemplo 4

$$\begin{aligned} T(1) &= 1 \\ T(n) &= T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + 6n + 5 \quad \text{para } n = 2, 3, 4, 5, \ldots \end{aligned}$$

n	1	2	3	4	5	6	7	8	9	10
$T(n)$	1	19	43	67	97	127	157	187	223	259

Exemplo 4

$$
\begin{aligned}
T(1) &= 1 \\
T(n) &= T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + 6n + 5 \quad \text{para } n = 2, 3, 4, 5, \ldots
\end{aligned}
$$

n	1	2	3	4	5	6	7	8	9	10
$T(n)$	1	19	43	67	97	127	157	187	223	259

$T(n) = \mathrm{O}(???)$

Exemplo 4

Vou mostrar que $T(n) \leq 20\, n \lg n$ para $n = 2, 3, 4, \ldots$

n	1	2	3	4	5	6	7	8	9	10
$T(n)$	1	19	43	67	97	127	157	187	223	259
$20\, n \lfloor \lg n \rfloor$	0	40	60	160	200	240	280	480	540	600

Exemplo 4

Vou mostrar que $T(n) \leq 20\,n \lg n$ para $n = 2, 3, 4, \ldots$

n	1	2	3	4	5	6	7	8	9	10
$T(n)$	1	19	43	67	97	127	157	187	223	259
$20\,n\lfloor \lg n \rfloor$	0	40	60	160	200	240	280	480	540	600

Prova:
Se $n = 2, 3$, então $T(n) \leq 20\,n\lfloor \lg n \rfloor \leq 20\,n \lg n$, como mostra a tabela.

Note que a base da indução é $n = 2, 3$. Por quê???

Exemplo 4

Prova: (continuação) Se $n \geq 4$

$$
\begin{aligned}
T(n) &= T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 6n + 5 \\
&\overset{\text{hi}}{\leq} 20 \left\lceil \frac{n}{2} \right\rceil \lg \left\lceil \frac{n}{2} \right\rceil + 20 \left\lfloor \frac{n}{2} \right\rfloor \lg \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \\
&\leq 20 \left\lceil \frac{n}{2} \right\rceil \lg n + 20 \left\lfloor \frac{n}{2} \right\rfloor \lg \frac{n}{2} + 6n + 5 \\
&= 20 \left\lceil \frac{n}{2} \right\rceil \lg n + 20 \left\lfloor \frac{n}{2} \right\rfloor (\lg n - 1) + 6n + 5 \\
&= 20 \left(\left\lceil \frac{n}{2} \right\rceil + \left\lfloor \frac{n}{2} \right\rfloor \right) \lg n - 20 \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \\
&= 20\, n \lg n - 20 \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \\
&\leq 20\, n \lg n \quad \text{(pois } n \geq 4\text{)}
\end{aligned}
$$

Iiiiééééssss!

Como achei as contantes?

$$
\begin{aligned}
T(1) &= 1 \\
T(n) &= T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 6n + 5 \quad \text{para } n = 2, 3, 4, \ldots
\end{aligned}
$$

Supeito que $T(n) = \mathrm{O}(n \lg n)$.

Como achei as contantes?

$$\begin{aligned} T(1) &= 1 \\ T(n) &= T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 6n + 5 \quad \text{para } n = 2, 3, 4, \ldots \end{aligned}$$

Supeito que $T(n) = \mathrm{O}(n \lg n)$.

Vamos tentar mostrar isto!

Precisamos encontrar um número real positivo c e um número inteiro positivo n_0 tais que

$$T(n) \leq c\, n \lg n$$

para todo $n \geq n_0$.

Rascunho

Suponha que existam as tais constantes c e n_0.
Vamos descobrir a "cara" delas.

$$
\begin{aligned}
T(n) &= T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 6n + 5 \\
&\overset{\text{hi}}{\leq} c \left\lceil \frac{n}{2} \right\rceil \lg \left\lceil \frac{n}{2} \right\rceil + c \left\lfloor \frac{n}{2} \right\rfloor \lg \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \\
&\leq c \left\lceil \frac{n}{2} \right\rceil \lg n + c \left\lfloor \frac{n}{2} \right\rfloor \lg n + 6n + 5 \\
&= c \left(\left\lceil \frac{n}{2} \right\rceil + \left\lfloor \frac{n}{2} \right\rfloor \right) \lg n + 6n + 5 \\
&= c\, n \lg n + 6n + 5 \ldots \quad \text{Hmmm, não deu...}
\end{aligned}
$$

Nova tentativa

Suponha que existam c e n_0.

$$
\begin{aligned}
T(n) &= T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 6n + 5 \\
&\overset{\text{hi}}{\leq} c \left\lceil \frac{n}{2} \right\rceil \lg \left\lceil \frac{n}{2} \right\rceil + c \left\lfloor \frac{n}{2} \right\rfloor \lg \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \\
&\leq c \left\lceil \frac{n}{2} \right\rceil \lg n + c \left\lfloor \frac{n}{2} \right\rfloor \lg \frac{n}{2} + 6n + 5 \\
&= c \left\lceil \frac{n}{2} \right\rceil \lg n + c \left\lfloor \frac{n}{2} \right\rfloor (\lg n - 1) + 6n + 5 \\
&= c \left(\left\lceil \frac{n}{2} \right\rceil + \left\lfloor \frac{n}{2} \right\rfloor \right) \lg n - c \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \\
&= c\, n \lg n - c \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \quad \text{Agora vai!}
\end{aligned}
$$

Agora vai

Queremos saber “quando”

$$c\left\lfloor\frac{n}{2}\right\rfloor \geq 6n+5.$$

Bem, para $c = 20$ e todo $n \geq n_0 = 4$ temos que

$$\begin{aligned} c\left\lfloor\frac{n}{2}\right\rfloor &\geq c\,\frac{n-1}{2} \\ &= 20\,\frac{n-1}{2} \\ &= 10n - 10 \\ &> 6n + 5 \quad \text{(pois } n \geq 4\text{)} \end{aligned}$$

Eis constantes que fazem o serviço: $c = 20$ e $n_0 = 4$

Conclusão

Rearranja $A[p \, . \, . \, r]$, com $p \leq r$, em ordem crescente.

MERGE-SORT (A, p, r)
1 **se** $p < r$
2 **então** $q \leftarrow \lfloor (p + r)/2 \rfloor$
3 MERGE-SORT (A, p, q)
4 MERGE-SORT $(A, q + 1, r)$
5 INTERCALA (A, p, q, r)

Conclusão:

O MERGE-SORT consome $\mathrm{O}(n \lg n)$ unidades de tempo.

Exemplo 4 (continuação)

Vou mostrar que $T(n) \geq n \lg n$ para $n = 1, 2, 3, 4, \ldots$

n	1	2	3	4	5	6	7	8	9	10
$T(n)$	1	19	43	67	97	127	157	187	223	259
$n\lceil \lg n \rceil$	0	2	6	8	15	18	21	24	36	40

Exemplo 4 (continuação)

Vou mostrar que $T(n) \geq n \lg n$ para $n = 1, 2, 3, 4, \ldots$

n	1	2	3	4	5	6	7	8	9	10
$T(n)$	1	19	43	67	97	127	157	187	223	259
$n\lceil \lg n \rceil$	0	2	6	8	15	18	21	24	36	40

Prova:

Se $n = 1$, então $T(1) = 1 > 1 \cdot \lg 1 = 0$.

Exemplo 4

Prova: (continuação) Se $n \geq 2$, então

$$
\begin{aligned}
T(n) &= T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + 6n + 5 \\
&\overset{\text{hi}}{\geq} \left\lceil \frac{n}{2} \right\rceil \lg \left\lceil \frac{n}{2} \right\rceil + \left\lfloor \frac{n}{2} \right\rfloor \lg \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \\
&\geq \left\lceil \frac{n}{2} \right\rceil \lg \frac{n}{2} + \left\lfloor \frac{n}{2} \right\rfloor \lg \frac{n-1}{2} + 6n + 5 \\
&= \left\lceil \frac{n}{2} \right\rceil (\lg n - 1) + \left\lfloor \frac{n}{2} \right\rfloor (\lg(n-1) - 1) + 6n + 5 \\
&\geq \left\lceil \frac{n}{2} \right\rceil (\lg n - 1) + \left\lfloor \frac{n}{2} \right\rfloor (\lg n - 2) + 6n + 5 \\
&= (\left\lceil \frac{n}{2} \right\rceil + \left\lfloor \frac{n}{2} \right\rfloor) \lg n - \left\lceil \frac{n}{2} \right\rceil - 2 \left\lfloor \frac{n}{2} \right\rfloor + 6n + 5 \\
&\geq n \lg n - 3 \left\lceil \frac{n}{2} \right\rceil + 6n + 5 \geq n \lg n.
\end{aligned}
$$

Iiiiééééssss!

Exemplo 4

$$
\begin{aligned}
T(1) &= 1 \\
T(n) &= T(\lfloor n/2 \rfloor) + T(\lceil n/2 \rceil) + 6n + 5 \quad \text{para } n = 2, 3, 4, 5, \ldots
\end{aligned}
$$

n	1	2	3	4	5	6	7	8	9	10
$T(n)$	1	19	43	67	97	127	157	187	223	259

Conclusão:

$T(n)$ é $\Theta(n \lg n)$.

Conclusão da conclusão

O consumo de tempo do MERGE-SORT é $\Theta(n \lg n)$ no pior caso.

Exercício. Mostre que:

O consumo de tempo do MERGE-SORT é $\Theta(n \lg n)$.

Hmmmm... Qual a diferença entra as duas afirmações?

Recorrências com notação O

CLRS 4.1–4.2

AU 3.9, 3.11

Classe O da solução de uma recorrência

Não faço questão de solução exata: basta solução aproximada (em notação O, se possível Θ)

Classe O da solução de uma recorrência

Não faço questão de solução exata: basta solução aproximada (em notação O, se possível Θ)

Exemplo:

$$\begin{aligned} G(1) &= 1 \\ G(n) &= 2G(n/2) + 7n + 2 \quad \text{para } n = 2, 4, 8, 16, \ldots \end{aligned}$$

Solução exata: $G(n) = 7n \lg n + 3n - 2$

Solução aproximada: $G(n) = \mathrm{O}(n \lg n)$ $\quad (G(n) = \Theta(n \lg n))$

Classe O da solução de uma recorrência

Não faço questão de solução exata: basta solução aproximada (em notação O, se possível Θ)

Exemplo:

$$\begin{aligned} G(1) &= 1 \\ G(n) &= 2G(n/2) + 7n + 2 \quad \text{para } n = 2, 4, 8, 16, \ldots \end{aligned}$$

Solução exata: $G(n) = 7n \lg n + 3n - 2$

Solução aproximada: $G(n) = \mathrm{O}(n \lg n)$ $\quad$ ($G(n) = \Theta(n \lg n)$)

Em geral, é mais fácil obter e provar solução aproximada que solução exata

Dica prática (sem prova)

A solução da recorrência

$$\begin{aligned} T(1) &= 1 \\ T(n) &= 2T(\lfloor n/2 \rfloor) + 7n + 2 \quad \text{para } n = 2, 3, 4, 5, \ldots \end{aligned}$$

está na mesma classe Θ que a solução de

$$\begin{aligned} T'(1) &= 1 \\ T'(n) &= 2T'(n/2) + n \quad \text{para } n = 2, 2^2, 2^3, \ldots \end{aligned}$$

Dica prática (sem prova)

A solução da recorrência

$$\begin{aligned} T(1) &= 1 \\ T(n) &= 2T(\lfloor n/2 \rfloor) + 7n + 2 \quad \text{para } n = 2, 3, 4, 5, \ldots \end{aligned}$$

está na mesma classe Θ que a solução de

$$\begin{aligned} T'(1) &= 1 \\ T'(n) &= 2T'(n/2) + n \quad \text{para } n = 2, 2^2, 2^3, \ldots \end{aligned}$$

e na mesma classe Θ que a solução de

$$\begin{aligned} T''(4) &= 10 \\ T''(n) &= 2T''(n/2) + n \quad \text{para } n = 2^3, 2^4, 2^5, \ldots \end{aligned}$$

Recorrências com O do lado direito

A “recorrência”

$$T(n) = 2T(n/2) + \mathrm{O}(n)$$

representa todas as recorrências da forma $T(n) = 2T(n/2) + f(n)$ em que $f(n)$ é $\mathrm{O}(n)$.

Recorrências com O do lado direito

A "recorrência"

$$T(n) = 2T(n/2) + \mathrm{O}(n)$$

representa todas as recorrências da forma
$T(n) = 2T(n/2) + f(n)$ em que $f(n)$ é $\mathrm{O}(n)$.

Melhor: representa todas as recorrências do tipo

$$\begin{aligned} T'(n) &\le a \quad \text{para } n = k, k+1, \ldots, 2k-1 \\ T'(n) &\le 2T'(\lfloor n/2 \rfloor) + b\,n \quad \text{para } n \ge 2k \end{aligned}$$

quaisquer que sejam a, $b > 0$ e $k > 0$
(poderíamos tomar $n_0 = 1$; veja ex ??.I)

Recorrências com O do lado direito

A "recorrência"

$$T(n) = 2T(n/2) + \mathrm{O}(n)$$

também representa todas as recorrências do tipo

$$\begin{aligned} T''(n) &= a \quad \text{para } n = 2^{k-1} \\ T''(n) &\le 2T''(n/2) + b\,n \quad \text{para } n = 2^k, 2^{k+1}, \ldots \end{aligned}$$

quaisquer que sejam a, $b > 0$ e $k > 0$

Recorrências com O do lado direito

A "recorrência"

$$T(n) = 2T(n/2) + \mathrm{O}(n)$$

também representa todas as recorrências do tipo

$$\begin{array}{rcll} T''(n) & = & a & \text{para } n = 2^{k-1} \\ T''(n) & \leq & 2T''(n/2) + b\,n & \text{para } n = 2^k, 2^{k+1}, \ldots \end{array}$$

quaisquer que sejam a, $b > 0$ e $k > 0$

As soluções exatas vão depender de a, b, k;
mas todas estarão na mesma classe O (no caso, em $\mathrm{O}(n \lg n)$)

Exemplo com O

Recorrência com O do lado direito

Na análise do consumo de tempo do MERGE-SORT tinhamos:

$T(n) :=$ consumo de tempo máximo quando $n = r - p + 1$

$$T(1) = \mathrm{O}(1)$$
$$T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \mathrm{O}(n) \quad \text{para } n = 2, 3, 4, \ldots$$

Solução aproximada: $T(n)$ é $\mathrm{O}(n \lg n)$.

Exemplo com Θ

Recorrências com Θ do lado direito

Na análise do consumo de tempo do MERGE-SORT poderíamos ter feito:

$T(n) :=$ consumo de tempo quando $n = r - p + 1$

$$T(1) = \Theta(1)$$
$$T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n) \quad \text{para } n = 2, 3, 4, \ldots$$

Solução aproximada: $T(n)$ é $\Theta(n \lg n)$.

Dica prática

Suponha dada uma "recorrência" como

$$T(n) = 2T(n/2) + \mathrm{O}(n).$$

Para ter uma idéia da classe O a que T pertence, resolva a recorrência

$$\begin{array}{rcl} T''(2^0) & = & 1 \\ T''(n) & \leq & 2T''(n/2) + n \quad \text{para } n = 2^1, 2^2, 2^3, \ldots \end{array}$$

Mais dicas práticas

recorrência	condição	solução
$T(n) = T(n-1) + 4n^3$		$\Theta(n^{3+1})$
$T(n) = 6T(n-1) + 4n^3$		$\Theta(6^n)$
$T(n) = aT(n/5) + 4n^3$	$a < 5^3$	$\Theta(n^3)$
$T(n) = aT(n/5) + 4n^3$	$a = 5^3$	$\Theta(n^3 \log n)$
$T(n) = aT(n/5) + 4n^3$	$a > 5^3$	$\Theta(n^{\log_5 a})$

Veja AU, sec 3.11, p.151

Mais dicas práticas (continuação)

No lugar de

- "$n/5$", posso escrever "$\lfloor n/5 \rfloor$" ou "$\lceil n/5 \rceil$"
- "5", posso escrever qualquer $b > 1$
- "4", posso escrever qualquer número real
- "$4n^3$" posso escrever qualquer polinômio de grau 3
- "3", posso escrever qualquer inteiro $k \geq 0$
- "6", posso escrever qualquer número $a > 1$

Mais dicas práticas ainda

A mesma coisa, escrita de maneira um pouco diferente (Master Theorem, CLRS, sec. 4.3, p.73):

Suponha $T(n) = aT(n/5) + f(n)$ para algum $a \geq 1$. Então, em geral,

$$
\begin{array}{lll}
\text{se } f(n) = \mathrm{O}(n^{\log_5 a - \epsilon}) & \text{então} & T(n) = \Theta(n^{\log_5 a}) \\
\text{se } f(n) = \Theta(n^{\log_5 a}) & \text{então} & T(n) = \Theta(n^{\log_5 a} \lg n) \\
\text{se } f(n) = \Omega(n^{\log_5 a + \epsilon}) & \text{então} & T(n) = \Theta(f(n))
\end{array}
$$

para qualquer $\epsilon > 0$

Exercícios

Exercício 9.A
A que classe O pertencem as solução de recorrência do tipo $T(n) = T(n/3) + \mathrm{O}(1)$?

Exercício 9.B
Seja T a função definida pela recorrência

$$\begin{aligned} T(1) &= 1 \\ T(n) &= 4T(\lfloor n/2 \rfloor) + n \qquad \text{para } n = 2, 3, 4, 5, \ldots \end{aligned}$$

A que ordem Θ pertence T?

Exercício 9.C [CLRS 4.2-1]
Seja T a função definida pela recorrência

$$\begin{aligned} T(1) &= 1 \\ T(n) &= 3T(\lfloor n/2 \rfloor) + n \qquad \text{para } n = 2, 3, 4, 5, \ldots \end{aligned}$$

A partir da árvore da recorrência, adivinhe a que classe Θ pertence $T(n)$. Prove a delimitação pelo método da substituição.

Mais exercícios

Exercício 9.D

Resolva a recorrência

$$\begin{aligned} T(1) &= 1 \\ T(n) &= 2T(\lceil n/2 \rceil) + 7n + 2 \quad \text{para } n = 2, 3, 4, 5, \cdots \end{aligned}$$

Exercício 9.E

Resolva a “recorrência” $T(n) = T(n-2) + \mathrm{O}(n)$.

Exercício 9.F

Resolva a “recorrência” $T(n) = 5T(n-1) + \mathrm{O}(n)$.