Melhores momentos

AULA PASSADA

Chão e teto

$\lfloor x \rfloor$:= inteiro i tal que $i \leq x < i + 1$

$\lceil x \rceil$:= inteiro j tal que $j - 1 < x \leq j$

Exercício A1.B

Mostre que

$$\frac{n-1}{2} \leq \left\lfloor \frac{n}{2} \right\rfloor \leq \frac{n}{2} \qquad \text{e} \qquad \frac{n}{2} \leq \left\lceil \frac{n}{2} \right\rceil \leq \frac{n+1}{2}$$

para qualquer inteiro $n \geq 1$.

Ordenação por inserção

Algoritmo rearranja $A[1 \,.\,.\, n]$ em ordem crescente.

ORDENA-POR-INSERÇÃO (A, n)

para $j \leftarrow 2$ **até** n **faça**

  $chave \leftarrow A[j]$

  $i \leftarrow j - 1$

  enquanto $i \geq 1$ **e** $A[i] > chave$ **faça**

    $A[i+1] \leftarrow A[i]$ $\triangleright$ desloca

    $i \leftarrow i - 1$

  $A[i+1] \leftarrow chave$ $\triangleright$ insere

Invariantes

Correção de algoritmos iterativos e invariantes

Relação invariante chave:

(i0) na linha 1 vale que: $A[1\,..\,j-1]$ é crescente.

1						j				n
20	25	35	40	44	55	38	99	10	65	50

Supondo que a invariante vale. Correção do algoritmo é evidente!

No início da última iteração das linhas 1–7 tem-se que $j = n+1$. Da invariante concluí-se que $A[1\,..\,n]$ é crescente.

Consumo de tempo

Se a execução de cada linha de código consome 1 unidade de tempo o consumo total é:

linha	todas as execuções da linha
1	$= \quad n$
2	$= \quad n-1$
3	$= \quad n-1$
4	$\leq \quad 2+3+\cdots+n \;=\; (n-1)(n+2)/2$
5	$\leq \quad 1+2+\cdots+(n-1) \;=\; n(n-1)/2$
6	$\leq \quad 1+2+\cdots+(n-1) \;=\; n(n-1)/2$
7	$= \quad n-1$
total	$\leq \quad (3/2)n^2+(7/2)n-4$

$(3/2)n^2 + (7/2)n - 4$ **versus** $(3/2)n^2$

n	$(3/2)n^2 + (7/2)n - 4$	$(3/2)n^2$
64	6364	6144
128	25020	24576
256	99196	98304
512	395004	393216
1024	1576444	1572864
2048	6298620	6291456
4096	25180156	25165824
8192	100691964	100663296
16384	402710524	402653184
32768	1610727420	1610612736

$(3/2)n^2$ domina os outros termos

Consumo de tempo

Se a execução da linha i consome t_i unidades de tempo, para $i = 1, \ldots, 7$, o consumo totalde tempo é:

linha	todas as execuções da linha	
1	$= \quad n$	$\times t_1$
2	$= \quad n - 1$	$\times t_2$
3	$= \quad n - 1$	$\times t_3$
4	$\leq \quad 2 + 3 + \cdots + n \;=\; (n-1)(n+2)/2$	$\times t_4$
5	$\leq \quad 1 + 2 + \cdots + (n-1) \;=\; n(n-1)/2$	$\times t_5$
6	$\leq \quad 1 + 2 + \cdots + (n-1) \;=\; n(n-1)/2$	$\times t_6$
7	$= \quad n - 1$	$\times t_7$

$$
\begin{array}{lll}
\text{total} & \leq & ((t_4 + t_5 + t_6)/2) \times n^2 \\
 & + & (t_1 + t_2 + t_3 + t_4/2 - t_5/2 - t_6/2 + t_7) \times n \\
 & - & (t_2 + t_3 + t_4 + t_7)
\end{array}
$$

Consumo de tempo

linha	todas as execuções da linha	
1	$= \quad n$	$\times t_1$
2	$= \quad n-1$	$\times t_2$
3	$= \quad n-1$	$\times t_3$
4	$\leq \quad 2+3+\cdots+n \; = \; (n-1)(n+2)/2$	$\times t_4$
5	$\leq \quad 1+2+\cdots+(n-1) \; = \; n(n-1)/2$	$\times t_5$
6	$\leq \quad 1+2+\cdots+(n-1) \; = \; n(n-1)/2$	$\times t_6$
7	$= \quad n-1$	$\times t_7$

total $\leq \quad c_2 \times n^2 + c_1 \times n + c_0$

c_2, c_1, c_0 são constantes que dependem da máquina.

n^2 é para sempre! Está nas entranhas do algoritmo!

Notação assintótica

CLRS 3.1

AU 3.4, p.96–100 (muito bom!)

Funções de n

$5n^2 - 9n \qquad 4n + 8 \qquad \lfloor n/3 \rfloor + 4 \qquad 2\sqrt{n} + 7$

$2^{n-3} \qquad \lg n$ ($= \log_2 n$)

Qual é maior: $n^2 - 9$ ou $4n + 8$?

Funções de n

$5n^2 - 9n$ $\quad$ $4n + 8$ $\quad$ $\lfloor n/3 \rfloor + 4$ $\quad$ $2\sqrt{n} + 7$

2^{n-3} $\quad$ $\lg n$ $\quad$ ($= \log_2 n$)

Qual é maior: $n^2 - 9$ ou $4n + 8$?

Depende do valor de n!

Funções de n

$5n^2 - 9n$ $\quad$ $4n + 8$ $\quad$ $\lfloor n/3 \rfloor + 4$ $\quad$ $2\sqrt{n} + 7$

2^{n-3} $\quad$ $\lg n$ $\quad$ ($= \log_2 n$)

Qual é maior: $n^2 - 9$ ou $4n + 8$?

Depende do valor de n!

Qual cresce mais?

Funções de n

$5n^2 - 9n$ $\quad$ $4n + 8$ $\quad$ $\lfloor n/3 \rfloor + 4$ $\quad$ $2\sqrt{n} + 7$

2^{n-3} $\quad$ $\lg n$ $\quad$ $(= \log_2 n)$

Qual é maior: $n^2 - 9$ ou $4n + 8$?

Depende do valor de n!

Qual cresce mais?

Comparação assintótica, ou seja, para n ENORME.

Funções de n

$5n^2 - 9n$ $\qquad 4n + 8$ $\qquad \lfloor n/3 \rfloor + 4$ $\qquad 2\sqrt{n} + 7$

2^{n-3} $\qquad \lg n \quad (= \log_2 n)$

Qual é maior: $n^2 - 9$ ou $4n + 8$?

Depende do valor de n!

Qual cresce mais?

Comparação assintótica, ou seja, para n ENORME.

$\lg n$ $\qquad 2\sqrt{n} + 7$ $\qquad \lfloor n/3 \rfloor + 4$ $\qquad 4n + 8$ $\qquad 5n^2 - 9n$

2^{n-3}

Notação O

Intuitivamente...

$\mathrm{O}(f(n))$ $\approx$ funções que não crescem mais rápido que $f(n)$

$\approx$ funções menores ou iguais a um múltiplo de $f(n)$

n^2 $(3/2)n^2$ $9999n^2$ $n^2/1000$ etc.

crescem todas com a mesma velocidade

Notação O

Intuitivamente...

$\mathrm{O}(f(n))$ $\approx$ funções que não crescem mais rápido que $f(n)$

$\approx$ funções menores ou iguais a um múltiplo de $f(n)$

n^2 $(3/2)n^2$ $9999n^2$ $n^2/1000$ etc.

crescem todas com a mesma velocidade

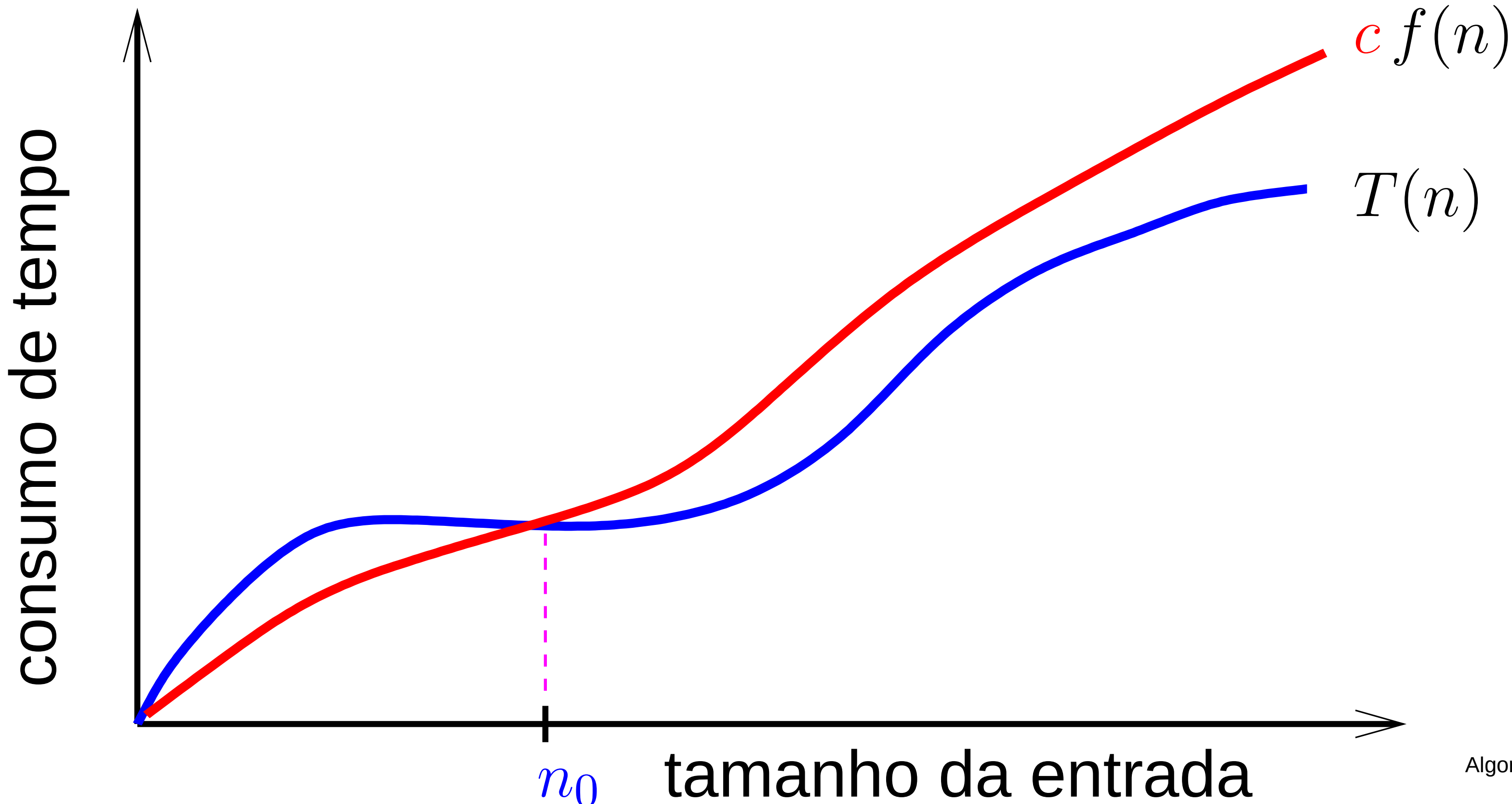
- $n^2 + 99n$ é $\mathrm{O}(n^2)$
- $33n^2$ é $\mathrm{O}(n^2)$
- $9n + 2$ é $\mathrm{O}(n^2)$
- $0{,}00001n^3 - 200n^2$ não é $\mathrm{O}(n^2)$

Definição

Sejam $T(n)$ e $f(n)$ funções dos inteiros nos reais. Dizemos que $T(n)$ é $\mathrm{O}(f(n))$ se existem constantes positivas c e n_0 tais que

$$T(n) \;\leq\; c\, f(n)$$

para todo $n \geq n_0$.

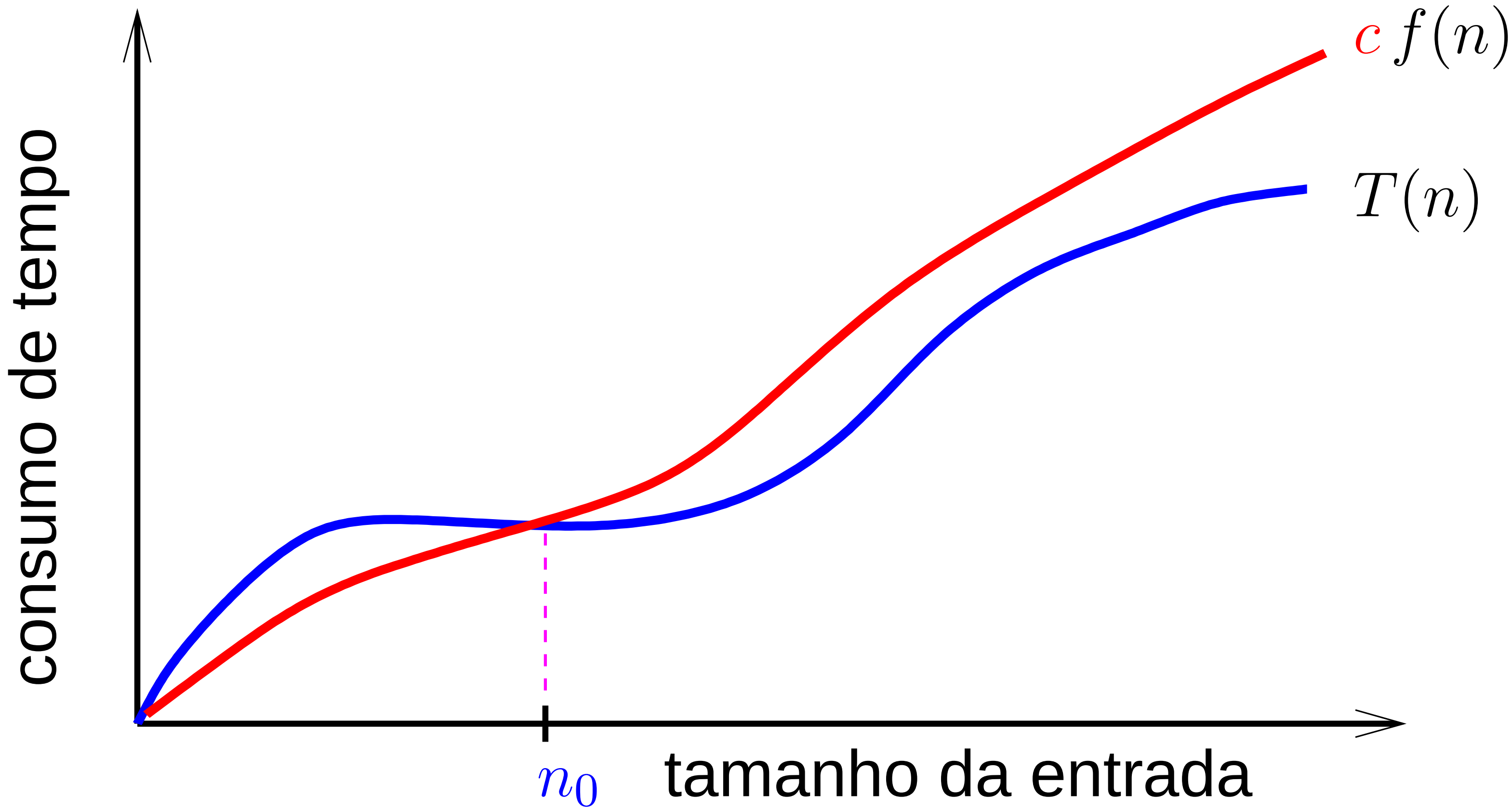


Mais informal

$T(n)$ é $\mathrm{O}(f(n))$ se existe $c > 0$ tal que

$$T(n) \;\leq\; c\, f(n)$$

para todo n suficientemente GRANDE.



Exemplos

$T(n)$ é $\mathrm{O}(f(n))$ lê-se "$T(n)$ é O de $f(n)$" ou
"$T(n)$ é da ordem de $f(n)$"

Exemplos

$T(n)$ é $\mathrm{O}(f(n))$ lê-se "$T(n)$ é O de $f(n)$" ou
"$T(n)$ é da ordem de $f(n)$"

Exemplo 1

Se $T(n) \leq 500 f(n)$ para todo $n \geq 10$, então $T(n)$ é $\mathrm{O}(f(n))$.

Exemplos

$T(n)$ é $\mathrm{O}(f(n))$ lê-se "$T(n)$ é O de $f(n)$" ou
"$T(n)$ é da ordem de $f(n)$"

Exemplo 1

Se $T(n) \leq 500 f(n)$ para todo $n \geq 10$, então $T(n)$ é $\mathrm{O}(f(n))$.

Prova: Aplique a definição com $c = 500$ e $n_0 = 10$.

Exemplos

$T(n)$ é $\mathrm{O}(f(n))$ lê-se "$T(n)$ é O de $f(n)$" ou
"$T(n)$ é da ordem de $f(n)$"

Exemplo 1

Se $T(n) \leq 500 f(n)$ para todo $n \geq 10$, então $T(n)$ é $\mathrm{O}(f(n))$.

Prova: Aplique a definição com $c = 500$ e $n_0 = 10$.

Exemplo 2

$10n^2$ é $\mathrm{O}(n^3)$.

Exemplos

$T(n)$ é $\mathrm{O}(f(n))$ lê-se "$T(n)$ é O de $f(n)$" ou
"$T(n)$ é da ordem de $f(n)$"

Exemplo 1

Se $T(n) \leq 500 f(n)$ para todo $n \geq 10$, então $T(n)$ é $\mathrm{O}(f(n))$.

Prova: Aplique a definição com $c = 500$ e $n_0 = 10$.

Exemplo 2

$10n^2$ é $\mathrm{O}(n^3)$.

Prova: Para $n \geq 0$, temos que $0 \leq 10n^2 \leq 10\, n^3$.

Outra prova: Para $n \geq 10$, temos $0 \leq 10n^2 \leq n \times n^2 = 1\, n^3$.

Mais exemplos

Exemplo 3

$\lg n$ é $\mathrm{O}(n)$.

Mais exemplos

Exemplo 3

$\lg n$ é $\mathrm{O}(n)$.

Prova: Para $n \geq 1$, tem-se que $\lg n \leq 1\, n$.

Mais exemplos

Exemplo 3

$\lg n$ é $\mathrm{O}(n)$.

Prova: Para $n \geq 1$, tem-se que $\lg n \leq 1\, n$.

Exemplo 4

$20n^3 + 10n \log n + 5$ é $\mathrm{O}(n^3)$.

Mais exemplos

Exemplo 3

$\lg n$ é $\mathrm{O}(n)$.

Prova: Para $n \geq 1$, tem-se que $\lg n \leq 1\, n$.

Exemplo 4

$20n^3 + 10n \log n + 5$ é $\mathrm{O}(n^3)$.

Prova: Para $n \geq 1$, tem-se que

$$20n^3 + 10n \lg n + 5 \leq 20n^3 + 10n^3 + 5n^3 = 35\, n^3.$$

Outra prova: Para $n \geq 10$, tem-se que

$$20n^3 + 10n \lg n + 5 \leq 20n^3 + n\, n \lg n + n \leq 20n^3 + n^3 + n^3 = 22\, n^3.$$

Mais exemplos ainda

Exemplo 5

$3 \lg n + \lg \lg n$ é $\mathrm{O}(\lg n)$.

Mais exemplos ainda

Exemplo 5

$3 \lg n + \lg \lg n$ é $\mathrm{O}(\lg n)$.

Prova: Para $n \geq 2$, tem-se que

$$3 \lg n + \lg \lg n \leq 3 \lg n + \lg n = 4 \lg n.$$

[Note que $\lg \lg n$ não é definida para $n = 1$.]

Mais exemplos ainda

Exemplo 5

$3 \lg n + \lg \lg n$ é $\mathrm{O}(\lg n)$.

Prova: Para $n \geq 2$, tem-se que

$$3 \lg n + \lg \lg n \leq 3 \lg n + \lg n = 4 \lg n.$$

[Note que $\lg \lg n$ não é definida para $n = 1$.]

Exemplo 6

10^{100} é $\mathrm{O}(1)$.

Mais exemplos ainda

Exemplo 5

$3 \lg n + \lg \lg n$ é $\mathrm{O}(\lg n)$.

Prova: Para $n \geq 2$, tem-se que

$$3 \lg n + \lg \lg n \leq 3 \lg n + \lg n = 4 \lg n.$$

[Note que $\lg \lg n$ não é definida para $n = 1$.]

Exemplo 6

10^{100} é $\mathrm{O}(1)$.

Prova: Para $n \geq 1$, tem-se que

$$10^{100} = 10^{100}\, n^0 = 10^{100} \times 1.$$

[Note que n não precisa aparecer, já que estamos lidando com funções constantes.]

Uso da notação O

$\mathrm{O}(f(n)) = \{T(n)$: existem c e n_0 tq $T(n) \leq cf(n), n \geq n_0\}$

"$T(n)$ é $\mathrm{O}(f(n))$" deve ser entendido como "$T(n) \in \mathrm{O}(f(n))$".

"$T(n) = \mathrm{O}(f(n))$" deve ser entendido como "$T(n) \in \mathrm{O}(f(n))$".

"$T(n) \leq \mathrm{O}(f(n))$" é feio.

"$T(n) \geq \mathrm{O}(f(n))$" não faz sentido!

"$T(n)$ é $g(n) + \mathrm{O}(f(n))$" significa que existe constantes positivas c e n_0 tais que

$$T(n) \leq g(n) + c\,f(n)$$

para todo $n \geq n_0$.

Nomes de classes O

classe	nome
$\mathrm{O}(1)$	constante
$\mathrm{O}(\lg n)$	logarítmica
$\mathrm{O}(n)$	linear
$\mathrm{O}(n \lg n)$	$n \log n$
$\mathrm{O}(n^2)$	quadrática
$\mathrm{O}(n^3)$	cúbica
$\mathrm{O}(n^k)$ com $k >= 1$	polinomial
$\mathrm{O}(2^n)$	exponencial
$\mathrm{O}(a^n)$ com $a > 1$	exponencial

Exercícios

Exercício 2.A

Prove que $n^2 + 10n + 20 \;=\; \mathrm{O}(n^2)$

Exercício 2.B

Prove que $300 \;=\; \mathrm{O}(1)$

Exercício 2.C

Prove que $\lceil n/3 \rceil \;=\; \mathrm{O}(n)$

É verdade que $n = \mathrm{O}(\lfloor n/3 \rfloor)$?

Exercício 2.D

Prove que $\lg n \;=\; \mathrm{O}(\log_{10} n)$

Exercício 2.E

Prove que $n = \mathrm{O}(2^n)$

Exercício 2.F

Prove que $\lg n = \mathrm{O}(n)$

Exercício 2.G

Prove que $n/1000$ não é $\mathrm{O}(1)$

Exercício 2.H

Prove que $\frac{1}{2}\,n^2$ não é $\mathrm{O}(n)$

Mais exercícios

Exercício 2.I

Suponha T definida para $n = 0, 1, \ldots$

Se $T(n) = \mathrm{O}(1)$, mostre que existe c' tal que $T(n) \leq c'$ para todo $n \geq 0$.

Se $T(n) = \mathrm{O}(n)$, mostre que existe c' tal que $T(n) \leq c'n$ para todo $n \geq 1$.

Exercício 2.J

Prove que $n^2 + 999n + 9999 = \mathrm{O}(n^2)$.

Exercício 2.K

Prove que $\frac{1}{2}n(n+1) \; = \; \mathrm{O}(n^2)$.

Exercício 2.L

É verdade que $\frac{1}{100}n^2 - 999n - 9999 = \mathrm{O}(n)$? Justifique.

Exercício 2.M

Suponha que $f(n) = n^2$ quando n é par e $f(n) = n^3$ quando n é ímpar.

É verdade que $f(n) = \mathrm{O}(n^2)$?

É verdade que $f(n) = \mathrm{O}(n^3)$?

É verdade que $n^2 = \mathrm{O}(f(n))$?

É verdade que $n^3 = \mathrm{O}(f(n))$?

Mais exercícios ainda

Exercício 2.N
É verdade que $n^2 = \mathrm{O}(2^n)$?

Exercício 2.O
É verdade que $\lg n = \mathrm{O}(\sqrt{n})$?

Exercício 2.P
Suponha $f(n) = 64n \lg n$ e $g(n) = 8n^2$, com n inteiro positivo.
Para que valores de n temos $f(n) \leq g(n)$?

Exercício 2.Q (bom!)
Suponha T e f definidas para $n = 1, 2, \ldots$ Mostre que se $T(n) = \mathrm{O}(f(n))$ e $f(n) > 0$ para $n \geq 1$ então existe c' tal que $T(n) \leq c' f(n)$ para todo $n \geq 1$.

Exercício 2.R (bom!)
Faz sentido dizer "$T(n) = \mathrm{O}(n^2)$ para $n \geq 3$"?

Mais exercícios ainda ainda

Exercício 2.S

É verdade que $2^n = \mathrm{O}(n)$?

É verdade que $n = \mathrm{O}(\lg n)$?

Justifique.

Exercício 2.T

É verdade que $n + \sqrt{n}$ é $\mathrm{O}(n)$?

É verdade que n é $\mathrm{O}(\sqrt{n})$?

É verdade que $n^{2/3}$ é $\mathrm{O}(\sqrt{n})$?

É verdade que $\sqrt{n} + 1000$ é $\mathrm{O}(n)$?

Exercício 2.U

É verdade que $\lg n = \mathrm{O}(n^{1/2})$?

É verdade que $\sqrt{n} = \mathrm{O}(\lg n)$?

É verdade que $\lg n = \mathrm{O}(n^{1/3})$?

Justifique. (Sugestão: prove, por indução, que $\lg x \leq x$ para todo número real $x \geq 1$.)

Exercício 2.V

É verdade que $\lceil \lg n \rceil = \mathrm{O}(\lg n)$?

Análise com notação O

CLRS 2.1–2.2

AU 3.3, 3.6 (muito bom)

Ordenação por inserção

Algoritmo rearranja $A[p \,..\, r]$ em ordem crescente

ORDENA-POR-INSERÇÃO (A, p, r)

para $j \leftarrow p+1$ **até** r **faça**
    $chave \leftarrow A[j]$

    $i \leftarrow j-1$
    enquanto $i \geq p$ **e** $A[i] > chave$ **faça**
        $A[i+1] \leftarrow A[i]$ $\triangleright$ desloca
        $i \leftarrow i-1$

    $A[i+1] \leftarrow chave$ $\triangleright$ insere

Quanto tempo o algoritmo consome?

Ordenação por inserção

Algoritmo rearranja $A[p \,.\,.\, r]$ em ordem crescente

ORDENA-POR-INSERÇÃO (A, p, r)

1 **para** $j \leftarrow p + 1$ **até** r **faça**

2 $\quad chave \leftarrow A[j]$

3 $\quad i \leftarrow j - 1$

4 $\quad$ **enquanto** $i \geq p$ **e** $A[i] > chave$ **faça**

5 $\quad\quad A[i+1] \leftarrow A[i]$ $\quad \triangleright$ desloca

6 $\quad\quad i \leftarrow i - 1$

7 $\quad A[i+1] \leftarrow chave$ $\quad \triangleright$ insere

Quanto tempo o algoritmo consome?

"Tamanho" do problema: $n := r - p + 1$

Consumo de tempo

linha	consumo de todas as execuções da linha
1	?
2	?
3	?
4	?
5	?
6	?
7	?
total	?

Consumo de tempo

linha	consumo de todas as execuções da linha
1	$\mathrm{O}(n)$
2	$\mathrm{O}(n)$
3	$\mathrm{O}(n)$
4	$n\mathrm{O}(n) = \mathrm{O}(n^2)$
5	$n\mathrm{O}(n) = \mathrm{O}(n^2)$
6	$n\mathrm{O}(n) = \mathrm{O}(n^2)$
7	$\mathrm{O}(n)$
total	$\mathrm{O}(3n^2 + 4n)$ = $\mathrm{O}(n^2)$

Justificativa

Bloco de linhas 4–6 é executado $\leq n$ vezes;
cada execução consome $O(n)$;
todas juntas consomem $nO(n)$.

Justificativa

Bloco de linhas 4–6 é executado $\leq n$ vezes;
cada execução consome $\mathrm{O}(n)$;
todas juntas consomem $n\mathrm{O}(n)$.

Êpa!

Quem garante que $n\mathrm{O}(n) = \mathrm{O}(n^2)$?
Quem garante que $\mathrm{O}(n^2) + \mathrm{O}(n^2) + \mathrm{O}(n^2) = \mathrm{O}(3n^2)$?
Quem garante que $\mathrm{O}(3n^2 + 4n) = \mathrm{O}(n^2)$?
Veja exercícios de Mais notação O.

$n\mathrm{O}(n) = \mathrm{O}(n^2)$

"$n\mathrm{O}(n) = \mathrm{O}(n^2)$" significa "$n\mathrm{O}(n) \subseteq \mathrm{O}(n^2)$".

Ou seja, se $T(n)$ é $\mathrm{O}(n)$, então $nT(n)$ é $\mathrm{O}(n^2)$.

$n\mathrm{O}(n) = \mathrm{O}(n^2)$

"$n\mathrm{O}(n) = \mathrm{O}(n^2)$" significa "$n\mathrm{O}(n) \subseteq \mathrm{O}(n^2)$".
Ou seja, se $T(n)$ é $\mathrm{O}(n)$, então $nT(n)$ é $\mathrm{O}(n^2)$.

De fato, se $T(n)$ é $\mathrm{O}(n)$ então existem constantes, digamos 10 e 10^{100}, tais que

$$T(n) \;\leq\; 10\, n$$

para todo $n \geq 10^{100}$. Desta forma,

$$n\, T(n) \;\leq\; n\, 10\, n \leq 10\, n^2$$

para todo $n \geq 10^{100}$. Logo $n\, T(n)$ é $\mathrm{O}(n^2)$.

$n\mathrm{O}(n) = \mathrm{O}(n^2)$

"$n\mathrm{O}(n) = \mathrm{O}(n^2)$" significa "$n\mathrm{O}(n) \subseteq \mathrm{O}(n^2)$".

Ou seja, se $T(n)$ é $\mathrm{O}(n)$, então $nT(n)$ é $\mathrm{O}(n^2)$.

De fato, se $T(n)$ é $\mathrm{O}(n)$ então existem constantes positivas c e n_0, tais que

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

para todo $n \geq n_0$. Desta forma,

$$n\,T(n) \ \leq \ n\,c\,n \leq c\,n^2$$

para todo $n \geq n_0$. Logo $n\,T(n)$ é $\mathrm{O}(n^2)$.

Conclusão

O algoritmo ORDENA-POR-INSERÇÃO consome $\mathrm{O}(n^2)$ unidades de tempo.

Notação O cai como uma luva!

Exercício

Exercício 3.A

Problema: rearranjar um vetor $A[p\,..\,r]$ em ordem crescente. Escreva um algoritmo “de seleção” para o problema. Analise a correção do algoritmo (ou seja, encontre e prove as invariantes apropriadas). Analise o consumo de tempo do algoritmo; use notação O.

Mais notação assintótica

CLRS 4.1
AU 4.5, p.101–108

Exercícios

Exercício 4.A
Interprete e prove a afirmação $\mathrm{O}(n^2) + \mathrm{O}(n^2) + \mathrm{O}(n^2) = \mathrm{O}(3n^2)$.

Exercício 4.B
Interprete e prove a afirmação $n\mathrm{O}(n) = \mathrm{O}(n^2)$.

Exercício 4.C
Interprete e prove a afirmação $\mathrm{O}(3n^2 + 4n) = \mathrm{O}(n^2)$.

Exercício 4.D (propriedade transitiva)
Suponha $T(n) = \mathrm{O}(f(n))$ e $f(n) = \mathrm{O}(g(n))$.
Mostre que $T(n) = \mathrm{O}(g(n))$.
Dê um exemplo interessante.

Exercício 4.E (regra da soma, caso especial)
Suponha que $T(n) = \mathrm{O}(f(n))$ e mostre que $T(n) + f(n) = \mathrm{O}(f(n))$.
Dê um exemplo interessante.

Exercício 4.E' (regra da soma, geral)
Suponha $T_1(n) = \mathrm{O}(f_1(n))$ e $T_2(n) = \mathrm{O}(f_2(n))$. Se $f_1(n) = \mathrm{O}(f_2(n))$, mostre que $T_1(n) + T_2(n) = \mathrm{O}(f_2(n))$.

Mais exercícios

Exercício 4.F
O que significa "$T(n) = n^2 + \mathrm{O}(n)$"?
Mostre que se $T(n) = n^2 + \mathrm{O}(n)$ então $T(n) = \mathrm{O}(n^2)$.

Exercício 4.G
O que significa "$T(n) = n\mathrm{O}(\lg n)$"? Mostre que $T(n) = n\mathrm{O}(\lg n)$ se e só se $T(n) = \mathrm{O}(n \lg n)$.

Exercício 4.H
Interprete e prove a afirmação $7 \cdot \mathrm{O}(n) = \mathrm{O}(n)$.

Exercício 4.I
Interprete e prove a afirmação $\mathrm{O}(n) + \mathrm{O}(n) = \mathrm{O}(n)$.

Exercício 4.J
Prove que $\mathrm{O}(n) = \mathrm{O}(n^2)$. É verdade que $\mathrm{O}(n^2) = \mathrm{O}(n)$?

Exercício 4.K
Interprete e prove a afirmação $(n + 2) \cdot \mathrm{O}(1) = \mathrm{O}(n)$.

Mais exercícios ainda

Exercício 4.L

Interprete e prove a afirmação $\underbrace{\mathrm{O}(1) + \cdots + \mathrm{O}(1)}_{n+2} = \mathrm{O}(n)$.

Exercício 4.M

Prove que $\mathrm{O}(1) + \mathrm{O}(1) + \mathrm{O}(1) = \mathrm{O}(1)$.

É verdade que $\mathrm{O}(1) = \mathrm{O}(1) + \mathrm{O}(1) + \mathrm{O}(1)$?

Exercício 4.N

Interprete e prove a afirmação $\mathrm{O}(f) + \mathrm{O}(g) = \mathrm{O}(f + g)$.

Mais análise com notação O

CLRS 2.1–2.2

AU 3.3, 3.6 (muito bom)

Análise da intercalação

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

Para que valores de q o problema faz sentido?

Entra:

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

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.

Para que valores de q o problema faz sentido?

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

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

Intercalação

	k								
A									

	i								j
B	22	33	55	77	99	88	66	44	11

Intercalação

		k							
A	11								

	i							j	
B	22	33	55	77	99	88	66	44	11

Intercalação

			k						
A	11	22							

		i						j	
B	22	33	55	77	99	88	66	44	11

Intercalação

				k					
A	11	22	33						

			i					j	
B	22	33	55	77	99	88	66	44	11

Intercalação

					k				
A	11	22	33	44					

			i				j		
B	22	33	55	77	99	88	66	44	11

Intercalação

						k			
A	11	22	33	44	55				

				i			j		
B	22	33	55	77	99	88	66	44	11

Intercalação

							k		
A	11	22	33	44	55	66			

				i		j			
B	22	33	55	77	99	88	66	44	11

Intercalação

								k	
A	11	22	33	44	55	66	77		

					i	j			
B	22	33	55	77	99	88	66	44	11

Intercalação

									k
A	11	22	33	44	55	66	77	88	

					$i = j$				
B	22	33	55	77	99	88	66	44	11

Intercalação

A	11	22	33	44	55	66	77	88	99

					j	i			
B	22	33	55	77	99	88	66	44	11

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:

linha	todas as execuções da linha
1	?
2	?
3	?
4	?
5	?
6	?
7	?
8	?
9–12	?
total	?

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

Conclusão

O algoritmo INTERCALA consome $6n+5$ unidades de tempo.

Consumo de tempo em O

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

linha	consumo de todas as execuções da linha
1–4	?
5–6	?
7	?
8	?
9–12	?
total	?

Consumo de tempo

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

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

Conclusão

O algoritmo INTERCALA consome $\mathrm{O}(n)$ unidades de tempo.

Também escreve-se

O algoritmo INTERCALA consome tempo $\mathrm{O}(n)$.

Exercício

Exercício 5.A

Analise a correção e o consumo de tempo do seguinte algoritmo:

EXERC (A, p, r)
1 $q \leftarrow \lfloor (p + r)/2 \rfloor$
2 ORDENA-POR-INSERÇÃO (A, p, q)
3 ORDENA-POR-INSERÇÃO $(A, q + 1, r)$
4 INTERCALA (A, p, q, r)