

# COMPLEXIDADE DE ALGORITMOS E CLASSES DE COMPLEXIDADE

DCE529 - Algoritmos e Estruturas de Dados III

Atualizado em: 29 de fevereiro de 2024

Iago Carvalho

Departamento de Ciência da Computação



Um algoritmo é uma sequência finita e bem definida de instruções ou passos lógicos que, quando seguidos corretamente, realizam uma tarefa específica ou resolvem um problema

- É uma abordagem sistemática e precisa para resolver um problema, que pode ser implementada por um computador, mas também pode ser executada manualmente

Os algoritmos são usados em muitos campos, incluindo

- |                         |               |
|-------------------------|---------------|
| ○ Ciência da Computação | ○ Estatística |
| ○ Física                | ○ Engenharias |
| ○ Matemática            | ○ Biologia    |
| ○ ...                   |               |

Existem diversos algoritmos diferentes para resolver um mesmo problema

- Existem diversas maneiras para se percorrer o caminho entre a cantina e a sala de aula
- Existem múltiplos meios para se desmontar uma caixa de papelão
- ...

Como podemos comparar (e avaliar) qual é a qualidade destes algoritmos?

- Tempo de processamento
- Espaço de memória

# TEMPO DE PROCESSAMENTO

É a medida que costuma ser mais importante

Existem três tipos de tempo de processamento que vale a pena serem estudados

- Melhor caso
- Caso médio
- Pior caso

Estes três tipos valem a pena serem estudados para alguns algoritmos

- Por exemplo, algoritmos de ordenação

Em projeto e análise de algoritmos, no geral, vamos analisar somente o **pior caso**

# TEMPO DE PROCESSAMENTO

Função de custo	Tamanho $n$					
	10	20	30	40	50	60
$n$	0,00001 s	0,00002 s	0,00003 s	0,00004 s	0,00005 s	0,00006 s
$n^2$	0,0001 s	0,0004 s	0,0009 s	0,0016 s	0,0.35 s	0,0036 s
$n^3$	0,001 s	0,008 s	0,027 s	0,64 s	0,125 s	0.316 s
$n^5$	0,1 s	3,2 s	24,3 s	1,7 min	5,2 min	13 min
$2^n$	0,001 s	1 s	17,9 min	12,7 dias	35,7 anos	366 séc.
$3^n$	0,059 s	58 min	6,5 anos	3855 séc.	$10^8$ séc.	$10^{13}$ séc.

A notação  $\mathcal{O}$  é utilizada para estudarmos o comportamento assintótico de funções

- Utilizada para estudar a taxa de crescimento de funções
- Também conhecido como a ordem de uma função

Esta notação estabelece um limite superior para o crescimento de uma função

- Utilizada para demonstrar o maior valor que uma função pode atingir para determinado valor de entrada
- Assim, utilizada para estudar o comportamento no pior caso de um algoritmo

## FORMALIZANDO A NOTAÇÃO $\mathcal{O}$

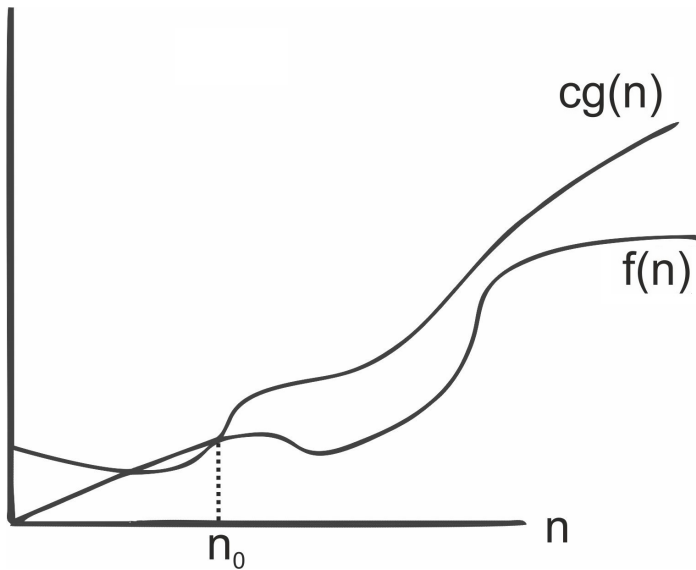
Sejam  $f$  e  $g$  duas funções definidas no mesmo subconjunto dos números reais pode-se dizer que

$$f(n) = \mathcal{O}(g(n)), n \rightarrow \infty$$

se e somente se existe uma constante positiva  $M$  tal que para todo valor suficientemente grande de  $n$ , o valor absoluto de  $f(n)$  é no máximo  $c$  multiplicado pelo valor absoluto de  $g(n)$

Ou seja,  $f(n) = \mathcal{O}(g(n))$  se e somente se existe um número real positivo  $c$  e um número real  $n_0$  tal que

$$|f(n)| \leq c|g(n)| \quad \forall n \geq n_0$$





# NOTAÇÃO $\mathcal{O}$

A notação  $\mathcal{O}$  não é afetada por fatores constantes ou termos de ordem menor

- Na prática, isso significa que devemos nos preocupar somente com o termo de maior expoente
- Também diz que qualquer constante que multiplica os termos pode ser ignorada
- Além disso, também só se considera o termo que possui relação com o tamanho  $n$  da entrada

## Exemplos

- $n^3 + 5n = \mathcal{O}(n^3)$
- $5n^2 + 3 = \mathcal{O}(n^2)$
- $n + 10^{10} = \mathcal{O}(n)$
- $3n^4 - n^3 = \mathcal{O}(n^4)$
- $n + b^2 = \mathcal{O}(n)$

## NOTAÇÃO $\mathcal{O}$ - EXEMPLO

1. Se  $f(n)$  é a soma de vários termos, o que possuir maior taxa de crescimento é mantido, e todos os outros são omitidos
2. Se  $f(n)$  é um produto de diversos fatores, quaisquer constantes (termos do produto que não dependem de  $n$ ) são omitidos

$$f(n) = 3n^4 - 40n^3 + 52$$

Queremos utilizar a notação  $\mathcal{O}$  para representar a taxa de crescimento desta função.

Como podemos proceder?

## NOTAÇÃO $\mathcal{O}$ - EXEMPLO

Esta função tem três termos

- ☐  $3n^4$
- ☐  $-40n^3$
- ☐  $52$

O termo que tem a maior taxa de crescimento é o que tem o maior expoente. Neste caso, é  $3n^4$ .

Neste termo, o 3 é uma constante. Assim, podemos ignorá-lo. Então temos que

$$3n^4 - 40n^3 + 52 = \mathcal{O}(n^4)$$

## NOTAÇÃO $\mathcal{O}$ - EXEMPLO

$$3n^4 - 40n^3 + 52 = \mathcal{O}(n^4)$$

Seja  $f(n) = 3n^4 - 40n^3 + 52$  e  $g(n) = n^4$ .

Temos que mostrar que  $|f(n)| \leq c|g(n)|$  para um valor  $c$  real e para todo valor de  $n \geq n_0$ .

$$|3n^4 - 40n^3 + 52| \leq 3n^4 + |40n^3| + 52$$

$$|3n^4 - 40n^3 + 52| \leq 3n^4 + 40n^4 + 52n^4$$

$$|3n^4 - 40n^3 + 52| \leq 95n^4$$

O termo de maior crescimento é quem determina a ordem de  $f(n)$

$$f(n) = 3n - 5 \log(n) + 20n + n^2 = \mathcal{O}(n^2)$$

Seja  $f_1(n) = \mathcal{O}(g_1(n))$  e  $f_2(n) = \mathcal{O}(g_2(n))$

Podemos estabelecer duas regras de produtos

1.  $f_1(n)f_2(n) = \mathcal{O}(g_1(n)g_2(n))$
2.  $f(n) \mathcal{O}(g(n)) = \mathcal{O}(f(n)g(n))$

Seja  $f_1(n) = \mathcal{O}(g_1(n))$  e  $f_2(n) = \mathcal{O}(g_2(n))$ . Além disso, seja  $f_3(n) = \mathcal{O}(g_1(n))$

Podemos estabelecer três regras de soma

1.  $f_1(n) + f_2(n) = \mathcal{O}(|g_1(n)| + |g_2(n)|)$
2.  $f_1(n) + f_3(n) = \mathcal{O}(g_1(n))$
3. Se  $f(n)$  e  $g(n)$  forem positivas, então  $f(n) + \mathcal{O}(g(n)) = \mathcal{O}(f(n) + g(n))$

Seja  $f(n) = \mathcal{O}(g(n))$ . Além disso, seja  $k$  uma constante diferente de zero.

Aqui, podemos estabelecer duas regras

1.  $\mathcal{O}(kg(n)) = \mathcal{O}(g(n))$
2.  $f(n) = \mathcal{O}(g(n)) \rightarrow k = \mathcal{O}(g(n))$



Igualdades de caminho único

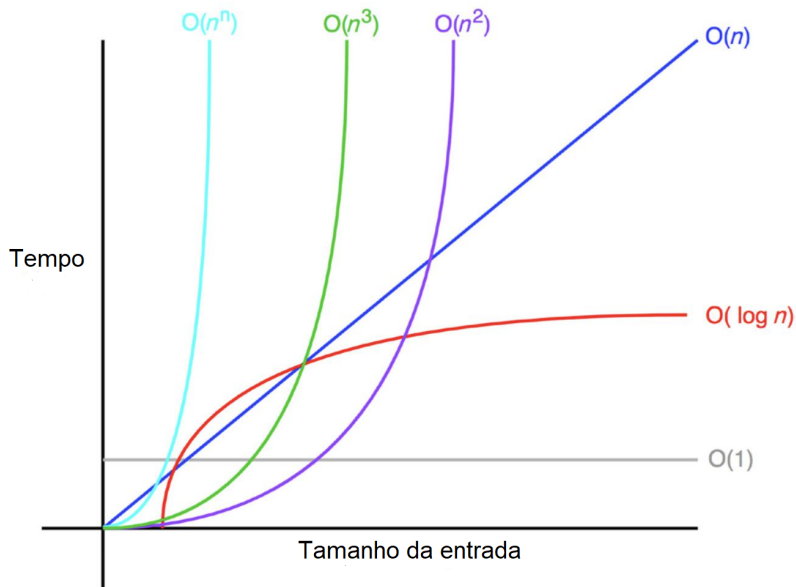
$$f(n) = \mathcal{O}(g(n)) \text{ não implica que } g(n) = \mathcal{O}(f(n))$$

Outras operações aritméticas

$$f(n) = h(n) + \mathcal{O}(g(n)) \rightarrow f(n) - h(n) = \mathcal{O}(g(n))$$

$$(n+3)^2 = n^2 + \mathcal{O}(n)$$

# COMPORTAMENTO ASSINTÓTICO DE DIFERENTES FUNÇÕES



A notação  $o$  é uma notação  $\mathcal{O}$  *afrouxada*

$$f(n) = o(g(n)) \leftarrow |f(n)| \leq \epsilon |g(n)| \quad \forall n \geq n_0, \quad \forall \epsilon \in \mathbb{R}_+$$

Assim, temos que

$$2n = o(n^2)$$

$$\frac{1}{n} = o(n)$$

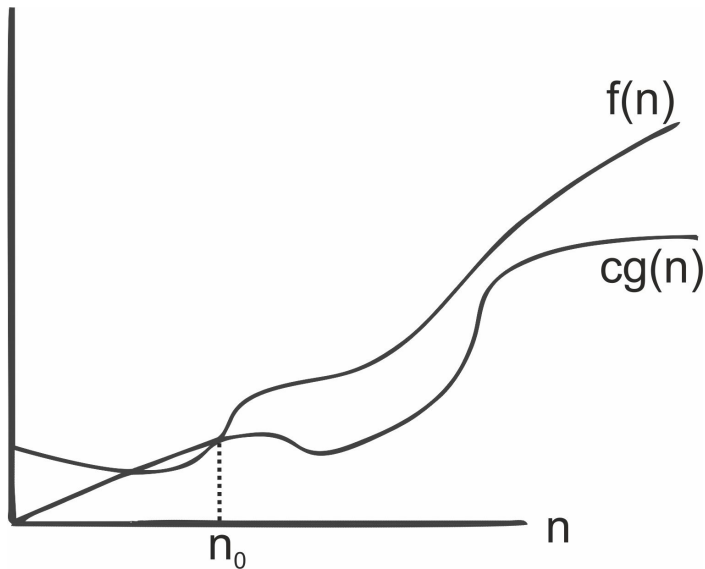
$$n \neq o(n)$$

Estabelece um limite inferior para o tempo de crescimento de uma função, sendo o inverso da notação  $\mathcal{O}$ . Isto é

$$\begin{aligned} \text{se } f(n) &= \mathcal{O}(g(n)), \\ \text{então } g(n) &= \Omega(f(n)) \end{aligned}$$

De forma similar, temos a notação  $\omega$ :

$$\begin{aligned} \text{se } f(n) &= o(g(n)), \\ \text{então } g(n) &= \omega(f(n)) \end{aligned}$$



A notação  $\Theta$  estabelece um limite assintótico firme para uma função  $f(n)$ . Isto é, se

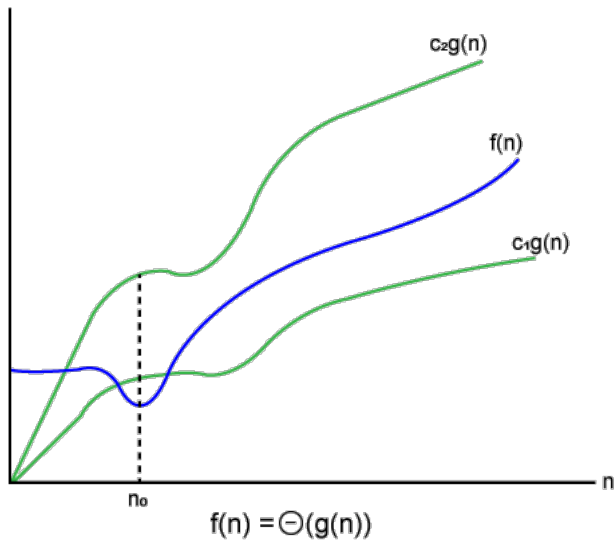
$$f(n) = \Theta(g(n))$$

então dizemos que existem constantes  $c_1$  e  $c_2$  reais e positivas tais que

$$c_1g(n) \leq f(n) \leq c_2g(n)$$

para todo valor  $n \geq n_0$

# NOTAÇÃO $\Theta$



## NOTAÇÃO $\Theta$ - EXEMPLO

Seja  $f(n) = 3n^3 + 6n + 7 = \Theta(n^3)$ . Temos que

$$\begin{aligned} 3n^3 &\leq 3n^3 + 6n + 7 \leq 3n^3 + 6n^2 + 7n^3 \\ 3n^3 &\leq f(n) \leq 16n^3 \end{aligned}$$

Assim, temos que  $c_1 = 3$ ,  $c_2 = 16$  e  $n = 1$



# RESUMO

Dizemos que  $f(n)$  é  $\mathcal{O}(g(n))$  se

- $f(n)$  cresce a uma taxa **menor ou igual** à  $g(n)$

Dizemos que  $f(n)$  é  $\Omega(g(n))$  se

- $f(n)$  cresce a uma taxa **maior ou igual** à  $g(n)$

Dizemos que  $f(n)$  é  $o(g(n))$  se

- $f(n)$  cresce a uma taxa **menor** que  $g(n)$

Dizemos que  $f(n)$  é  $\omega(g(n))$  se

- $f(n)$  cresce a uma taxa **maior** que  $g(n)$

Dizemos que  $f(n)$  é  $\Theta(g(n))$  se

- $f(n)$  cresce a uma taxa **similar** à de  $g(n)$

# COMPLEXIDADE E LIMITES

Também é possível estudar o comportamento assintótico de um algoritmo utilizando limites

- Aplicação direta da teoria aprendida em Cálculo I

Forma Assintótica	Definição
$f(n) \in \Theta(g(n))$	$0 < \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} < \infty$
$f(n) \in O(g(n))$	$0 \leq \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} < \infty$
$f(n) \in \Omega(g(n))$	$0 < \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)}$
$f(n) \in o(g(n))$	$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 0$
$f(n) \in \omega(g(n))$	$\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = \infty$

# EXERCÍCIOS

# EXERCÍCIOS

1. Verdadeiro ou falso: Se  $f(n) = \mathcal{O}(g(n))$ , então  $g(n) = \mathcal{O}(f(n))$
2. Verdadeiro ou falso:  $n - 1000 \log n = \mathcal{O}(n)$
3. Sejam  $f(n) = n^2 + n$  e  $g(n) = n^2$ . Mostre, usando limites, que  $f(n) = \mathcal{O}(g(n))$
4. Sejam  $f(n)$  e  $g(n)$  como acima definidos. Encontre os valores  $c_1, c_2$  e  $n$  tal que  $f(n) = \Theta(g(n))$