

Como medir o desempenho de algoritmos?

↳ Tempo (variável).

↳ Qtd de operações

Ex.: Encontrar o maior elemento num vetor.

```
INT MAIOR (INT V [], INT n) {
```

```
    INT m = V[0];
```

```
    FOR (INT i = 1; i < n; i++)
```

```
        IF (V[i] > m) m = V[i];
```

```
    RETURN m;
```

```
}
```

Pior caso

(melhor caso)

(caso médio)

Pior caso: último elemento é o maior.

↳ instância: vetor ordenado em ordem crescente

$$\begin{aligned} \text{Total de operações} &\leq \cancel{1} + \cancel{1} + n + \cancel{n-1} + \cancel{n-1} + \cancel{n-1} + \cancel{n-1} \\ &= 4n. \end{aligned}$$

Tempo e entrada crescem juntos; contar n //

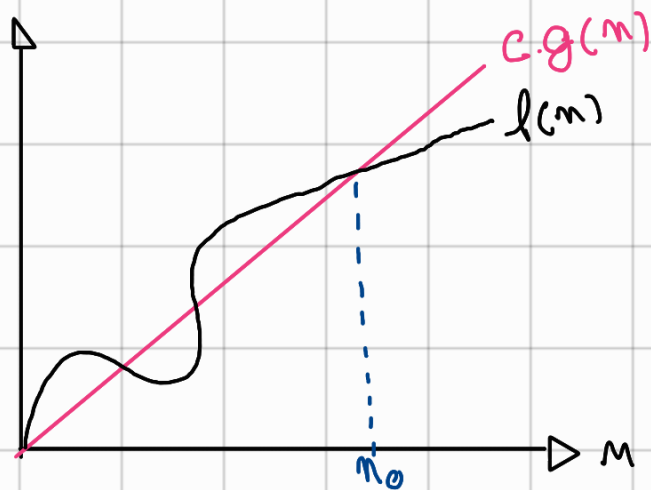
Contar exatamente pode ser complicado e exaustivo. O que se faz na prática, é estimar a ordem de

grandeza do total de operações em termos do tamanho da entrada. Para tanto, utilizamos uma ferramenta matemática chamada notação assintótica.

A notação assintótica tem por objetivo estimar como uma função cresce com relação ao crescimento de suas variáveis.
qtd. operações
Tamanho da entrada

Def.: Dizemos que uma função $f: \mathbb{N} \rightarrow \mathbb{N}$ é $O(g)$, para outra função $g: \mathbb{N} \rightarrow \mathbb{N}$, se existem constantes C e n_0 tais que:

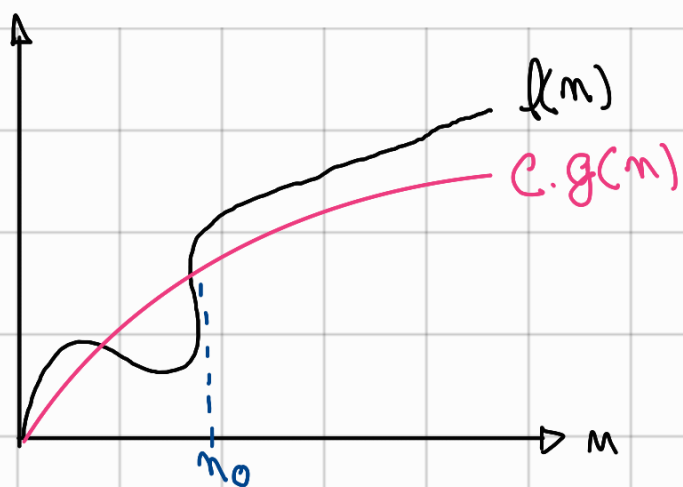
$$f(n) \leq C \cdot g(n), \forall n \geq n_0.$$



(Pior caso)

Analogamente, dizemos que f é $\Omega(g)$ se existem constantes C e n_0 tais que:

$$f(n) \geq C \cdot g(n), \forall n \geq n_0.$$



(Melhor caso)

$c.g(n)$

$f(n)$

n

n_0

Dizemos que f é $\Theta(g)$ se f é $O(g)$ e f é $\Omega(g)$

O algoritmo para encontrar o menor elemento no vetor é $\Theta(n)$ (pois é $O(n)$ e $\Omega(n)$).

Neste momento, há um conjunto de funções g padrões que vamos procurar enquadrar as estimativas do total de operações dos nossos algoritmos:

- $O(1) \rightarrow$ constante
- $O(\lg n) \rightarrow$ logarítmico
- $O(n) \rightarrow$ linear
- $O(n \lg n) \rightarrow$ linearítmico
- $O(n^k), k \geq 2 \rightarrow$ k -polinomial (quadrático, cúbico, ...)
- $O(k^n), k \geq 2 \rightarrow$ exponencial

complexidade assintótica

ou

custos

Suponha um computador que consome 0,001s/op

n	$O(n)$	$O(n \lg n)$	$O(n^2)$	$O(2^n)$
16	0,016s	0,064s	0,256s	1m4s
32	0,032s	0,16s	1s	46 dias
512	0,512s	9s	4m22s	10^{137} séc