Notação Assintótica: O

André Vignatti

DINF- UFPR

Notação Assintótica

Vamos expressar complexidade através de funções em variáveis que descrevam o tamanho de instâncias do problema.

Exemplos:

- Problemas de aritmética de precisão arbitrária: número de bits (ou bytes) dos inteiros.
- Problemas em grafos: número de vértices e/ou arestas
- Problemas de ordenação de vetores: tamanho do vetor.
- Busca em textos: número de caracteres do texto ou padrão de busca.

- Vamos supor que funções que expressam complexidade são sempre positivas, já que estamos medindo número de operações.

Notação assintótica

É um jeito de descrever o comportamento de funções no limite.

- A notação assintótica descreve o crescimento de funções.
- Foca no que é importante, ao abstrair os termos de baixa ordem e fatores constantes
- É o modo padrão de indicar o tempo de execução de algoritmos.

No decorrer da disciplina, iremos estudar a eficiência assintótica do algoritmos e estrutura de dados.

Comparação de Funções

- Vamos comparar funções assintoticamente, ou seja, para valores grandes, desprezando constantes multiplicativas e termos de menor ordem.

	$n = 100$	$n = 1000$	$n = 10^4$	$n = 10^6$	$n = 10^9$
$\log n$	2	3	4	6	9
n	100	1000	10^4	10^6	10^9
$n \log n$	200	3000	$4 \cdot 10^4$	$6 \cdot 10^6$	$9 \cdot 10^9$
n^2	10^4	10^6	10^8	10^{12}	10^{18}
$100n^2 + 15n$	$1,0015 \cdot 10^6$	$1,00015 \cdot 10^8$	$\approx 10^{10}$	$\approx 10^{14}$	$\approx 10^{20}$
2^n	$\approx 1,26 \cdot 10^{30}$	$\approx 1,07 \cdot 10^{301}$	?	?	?

Comparando funções

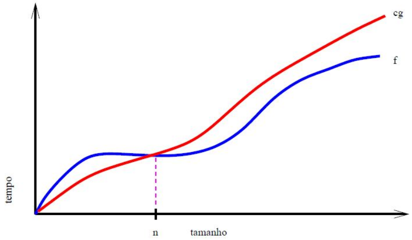
Com a notação assintótica, podemos comparar formalmente o "tamanho" das funções.

$$\begin{array}{ccc} O & \approx & \leq \\ \Omega & \approx & \geq \\ \Theta & \approx & = \\ o & \approx & < \\ \omega & \approx & > \end{array}$$

Notação O

Definição

$f(n)$ é $O(g(n))$ se existe constantes $c > 0$ e n_0 tal que $f(n) \leq cg(n)$, para todo $n \geq n_0$.



Exemplo

Seja $f(n) = n^2 + 2n + 1$. É verdade que $f(n) = O(n^2)$?

- $n^2 + 2n + 1 \leq n^2 + 2n^2 + n^2 = 4n^2$, sempre que $n \geq 1$. Então pegamos $c = 4$ e $n_0 = 1$.
- $n^2 + 2n + 1 \leq n^2 + n^2 + n^2 = 3n^2$, sempre que $n \geq 2$. Então pegamos $c = 3$ e $n_0 = 2$.

Exemplo

Seja $f(n) = n^2$. É verdade que $f(n) = O(n)$?

- Ou seja, $\exists c, n_0$ contantes tal que $n^2 \leq cn$ para $n \geq n_0$?
- (dividindo ambos os lados por n) $n \leq c$?
- Não é verdade!

Exemplo

Seja $f(n) = \log_2 n$. É verdade que $f(n) = O(\log_5 n)$?

- Note que $\log_2 n = \frac{\log_5 n}{\log_5 2}$ (mudança de base de logaritmo).
- Então, basta mostrar que $\exists c, n_0$ contantes tal que $\frac{\log_5 n}{\log_5 2} \leq c \log_5 n$, para $n \geq n_0$.
- (divide os dois lados por $log_5 n$) Assim, basta escolher $c \geq \frac{1}{\log_5 2}$ e $n_0 \geq 0$.

Podemos generalizar para obter o seguinte resultado:

Teorema (Exercício)

$\log_b n = O(\log_a n)$ *para todo* $a > 1, b > 1$.

Exemplo

As funções a seguir são $O(n^2)$

- n^2
- $n^2 + n$
- $n^2 + 1000n$
- $1000n^2 + 1000n$
- n
- $n/1000$
- $n^{1.9999999}$
- $n/\log_2 \log_2 \log_2 n$

Teorema da Soma

Sejam $\overline{f}(n), \overline{g}(n)$ funções não negativas tais que $\overline{f}(n) = O(f(n))$ e $\overline{g}(n) = O(g(n))$. Então

$$\overline{f}(n) + \overline{g}(n) = O(f(n) + g(n)).$$

Demonstração.

- Pela definição, $\exists c_1, n_1$ tal que $\overline{f}(n) \leq c_1 f(n)$ para $n \geq n_1$.
- Pela definição, $\exists c_2, n_2$ tal que $\overline{g}(n) \leq c_2 g(n)$ para $n \geq n_2$.
- Assim,

$$\begin{aligned} \overline{f}(n) + \overline{g}(n) &\leq c_1 f(n) + c_2 g(n) \\ &\leq \max\{c_1, c_2\}(f(n) + g(n)) \end{aligned}$$

  para $n \geq \max\{n_1, n_2\}$.
- Portanto, $\overline{f}(n) + \overline{g}(n) \leq c(f(n) + g(n))$ para algum $n \geq n_0$.

$\square$

Teorema da Multiplicação

Sejam $\overline{f}(n), \overline{g}(n)$ funções não negativas tais que $\overline{f}(n) = O(f(n))$ e $\overline{g}(n) = O(g(n))$. Então

$$\overline{f}(n)\overline{g}(n) = O(f(n)g(n)).$$

Demonstração.

Exercício $\square$

Usando os Teoremas da Soma e Multiplicação

Entrada: instância I tal que $|I| = n$

```
1 inicio
2     procedimento_1(I)
3     procedimento_2(I)
```

- Linha 2: $\overline{f}(n) = O(f(n))$.
- Linha 3: $\overline{g}(n) = O(g(n))$.
- Tempo Total $= \overline{f}(n) + \overline{g}(n) = O(f(n) + g(n))$.

Entrada: instância I tal que $|I| = n$

```
1 inicio
2     para i ← 1 até n hacer
3         procedimento_1(I)
```

- Linha 2: $O(n)$.
- Linha 3: $\overline{f}(n) = O(f(n))$.
- Tempo Total $= O(nf(n))$.

Use os Teoremas da Soma e Multiplicação para os seguintes exemplos:

Exemplo

Dê uma estimativa usando a notação O para $f(n) = 3n\log(n!) + (n^2 + 3)\log n$, onde n é inteiro positivo.

Exemplo

Dê uma estimativa usando a notação O para $f(n) = (n+1)\log(n^2+1) + 3n^2$.

Demonstração.

Exercício. $\square$