Lista 1

18:39 19:48
19:16

1 - ◦ Complexidade de tempo: Indica a rapidez com que um determinado algoritmo é executado.

◦ Complexidade de espaço: Indica a quantidade de unidades de memória utilizadas pelo algoritmo

* Em geral nos concentramos na complexidade de tempo.

↳ Como medir a complexidade de tempo?

↳ "Contando" o número de operações básicas realizadas em função do tamanho da entrada

* Em geral, estamos mais interessados na ordem de crescimento do que no número exato de operações.

◦ Operação básica: É a operação que mais contribui para o tempo de execução do algoritmo. É a operação mais custosa que é executada pelo maior número de vezes.

Tempo de execução

$$T(n) \approx C_{op} \, C(n)$$

↳ Complexidade de tempo

↳ Custo da operação básica.

Exemplo:

$$C(n) = \frac{1}{2} n(n-1) = \frac{1}{2} n^2 - \frac{1}{2} n \approx \frac{1}{2} n^2$$

$$\frac{T(2n)}{T(n)} \approx \frac{\cancel{C_{op}}\, C(2n)}{\cancel{C_{op}}\, C(n)} = \frac{\frac{1}{2}(2n)^2}{\frac{1}{2} n^2} = 4$$

2- Pior caso: É a situação para a qual o algoritmo executa o maior número de operações para uma entrada de tamanho n.

Eficiência de pior caso: É a eficiência do algoritmo para a entrada de tamanho n que faz com que ele execute o maior número de operações.

* Análise geralmente feita. Forma de estabelecer limites superiores para a execução de um algoritmo.

Eficiência do melhor caso: É a eficiência para a entrada de tamanho n que faz com que o algoritmo execute o menor número de operações possível

* Não é tão utilizada mas pode ser útil conhecer quais são os melhores casos de um algoritmo. Comentar insertion Sort

Eficiência de caso médio: É o desempenho esperado de um algoritmo. Mede o número de operações considerando uma distribuição de probabilidades sobre todas as possíveis entradas.

* Bem mais difícil de calcular mas fornece a visão mais realista do desempenho de um algoritmo.

Exemplo: Sequential Search $(A[0 \ldots n-1], k)$

```
i ← 0
while i < n and A[i] ≠ k do
    i ← i+1
if i < n
    return i
else
    return -1
```

Operação básica: adição (+)

Pior caso: K não está no array A

$$T(n) = \sum_{i=0}^{n-1} 1 = \underbrace{1 + 1 + \cdots + 1}_{?}$$

$$= (((n-1) - 0) + 1) \cdot 1 = n$$

Melhor caso: K está em $A[0]$

$$T(n) = 1$$

Caso médio:

$$T(n) = \sum_{i=0}^{n-1} (\text{comparações para a posição } i \times \text{probabilidade de estar na posição } i)$$

$$= \sum_{i=0}^{n-1} \left(i \times \frac{1}{n}\right)$$

$$= \frac{1}{n} \sum_{i=0}^{n-1} i$$

$$= \frac{1}{n}\left(\frac{n(n+1)}{2}\right) = \frac{n+1}{2}$$

4- a) $\log_2 n$

$\log_2 4n - \log_2 n$

$= (\log_2 4 + \log_2 n) - \log_2 n$

$= 2$ operações

b) $\frac{\sqrt{4n}}{\sqrt{n}} = \frac{2\cancel{\sqrt{n}}}{\cancel{\sqrt{n}}} = 2\times$ mais operações

c) $\frac{4n}{n} = 4$

d) $\frac{(4n)^2}{n^2} = \frac{16n^2}{n^2} = 16$

e) $\frac{(4n)^3}{n^3} = 64$

f) $\frac{2^{4n}}{2^n} = 2^{4n-n} = 2^{3n}$

5 - a) $\frac{\cancel{\frac{1}{3}}(10^3)^3}{\cancel{\frac{1}{3}}(5\cdot10^2)^3} = \frac{(10^3)^3}{5^3\cdot10^6} = \frac{10^9}{5^3\cdot10^6} = \frac{10^3}{5^3} \quad \frac{2^3\cdot5^3}{5^3} = 8$

b) Para um problema de tamanho n gastamos + tempo para executar cm $\frac{1}{3}n^3$ operações.

$t_{old} = C_M \frac{1}{3} n^3$

$t_{new} = 10^{-3} C_m \frac{1}{3} N^3$

$t_{old} = t_{new}$

$\cancel{C_m} \cancel{\tfrac{1}{3}} n^3 = 10^{-3} \cancel{C_m} \cancel{\tfrac{1}{3}} N^3$

$n^3 = 10^{-3} N^3$

$n = \frac{N}{10}$

6 - O - notation

$t(n)$ está em $O(g(n))$, $t(n) \in O(g(n))$, se $t(n)$ for limitada superiormente por algum multiplo constante de $g(n)$ para todos valores grandes de n, ou seja:

$t(n) \leq c\,g(n)$ para todo $n \geq n_0$

$c > 0$, $n_0 \geq 0$

Ω - notation

$t(n) \geq c\,g(n)$ para todo $n \geq n_0$

$c > 0$, $n_0 \geq 0$

Θ - notation

$c_2 g(n) \leq t(n) \leq c_1 g(n)$ para todo $n \geq n_0$

$c_1, c_2 > 0$, $n_0 \geq 0$

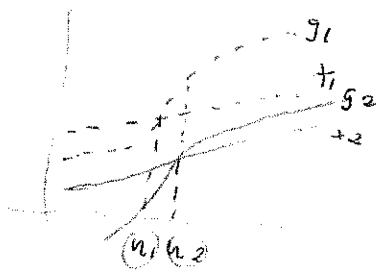
7 -

Teorema: Se $t_1(n) \in O(g_1(n))$ e $t_2(n) \in O(g_2(n))$

então $t_1(n) + t_2(n) \in O(\max\{g_1(n), g_2(n)\})$

$t_1(n) \leq c\, g_1(n) \quad n \geq n_1$

$t_2(n) \leq b\, g_2(n) \quad n \geq n_2$



Propriedade:

$a_1, b_1, a_2, b_2 \in \mathbb{R}$

Se

$a_1 \leq b_1$ e $a_2 \leq b_2 \Rightarrow a_1 + a_2 < 2\max\{b_1, b_2\}$

$\Rightarrow t_1(n) + t_2(n) \leq a\, g_1(n) + b\, g_2(n), \quad n \geq \max(n_1, n_2)$

$d = \max(a, b)$

$\Rightarrow t_1(n) + t_2(n) \leq d\, g_1(n) + d\, g_2(n)$

$g_3 = \max(g_1(n), g_2(n))$

$\Rightarrow t_1(n) + t_2(n) \leq 2d\, g_3(n), \quad n \geq \max(n_1, n_2)$

$\Rightarrow t_1(n) + t_2(n) \in O(g_3(n)), \quad c = 2d$

$= 2\max(a, b)$

$= t_1(n) + t_2(n) \in O(\max(g_1(n), g_2(n))) \quad n_0 = \max(n_1, n_2)$

8.-

a) $$\lim_{n\to\infty} \frac{\frac{1}{2}n(n-1)}{n^2} = \frac{1}{2}\lim_{n\to\infty}\frac{n^2-n}{n^2} = \frac{1}{2}\lim_{n\to\infty}\left(1-\frac{1}{n}\right) = \frac{1}{2}$$

b) $$\lim_{n\to\infty}\frac{\log_2 n}{\sqrt{n}} = \lim_{n\to\infty}\frac{(\log_2 n)'}{(\sqrt{n})'} = \lim_{n\to\infty}\frac{(\log_2 e)\frac{1}{n}}{\frac{1}{2\sqrt{n}}}$$

$$= 2\log_2 e \lim_{n\to\infty}\left(\frac{\sqrt{n}}{n}\cdot\frac{\sqrt{n}}{\sqrt{n}}\right) = 2\log_2 e\lim_{n\to\infty}\left(\frac{\cancel{n}}{\cancel{n}\sqrt{n}}\right) = 0$$

c) $$\lim_{n\to\infty}\frac{n!}{2^n} = \lim_{n\to\infty}\frac{\sqrt{2\pi n}\left(\frac{n}{e}\right)^n}{2^n}$$

$$= \lim_{n\to\infty}\sqrt{2\pi n}\,\frac{n^n}{2^n e^n} =$$

$$= \lim_{n\to\infty}\sqrt{2\pi n}\left(\frac{n}{2e}\right)^n = \infty$$

$$\lim_{n\to\infty}\frac{f(n)}{g(n)} = \begin{cases} 0, & f(n) \text{ tem ordem de crescimento menor que } g(n) \\ c, & \text{mesma ordem de crescimento} \\ \infty, & f(n) \text{ tem ordem de crescimento maior.} \end{cases}$$

10 -

$$\lim_{n\to\infty} \frac{P(n)}{n^k} = \lim_{n\to\infty} \left(\frac{a_k n^k + a_{k-1} n^{k-1} + a_{k-2} n^{k-2} + \dots + a_0}{n^k} \right)$$

$$= \lim_{n\to\infty} \left(a_k + \cancel{\frac{a_{k-1}}{n}} + \cancel{\frac{a_{k-2}}{n^2}} + \dots + \cancel{\frac{a_0}{n^k}} \right) = a_k$$

Para o que serve esse resultado.

11 -

$$\lim_{n\to\infty} \left(\frac{a_1^n}{a_2^n} \right) = \lim_{n\to\infty} \left(\frac{a_1}{a_2} \right)^n$$

- Se $a_1 > a_2 \to \infty$
- Se $a_1 < a_2 \to 0$
- Se $a_1 = a_2 \to 1$

$2^n \in O(3^n)$

$(10n)^2 \in \Theta(n^2)$

$(5n)^2 \in \Theta(n^2)$