

Programação Funcional

Análise de algoritmos

Marco A L Barbosa malbarbo.pro.br

Começando

- 1) O que significa analisar um algoritmo?
- 2) Por que a análise de algoritmos é importante?
- 3) O que significa dizer que a complexidade de tempo de um algoritmo é $T(n) = 3n + 7$?
- 4) O que significa dizer que uma função $f(n)$ é $O(g(n))$?
- 5) Em que situações utilizamos a notação Ω ?
- 6) Um algoritmo com complexidade de tempo $O(n)$ é melhor que um algoritmo com complexidade de tempo $O(n^2)$, verdadeiro ou falso?
- 7) O que é uma equação de recorrência?
- 8) O que significa resolver uma equação de recorrência?

Praticando

- 9) Resolva as seguintes equações de recorrência (utilize as fórmulas de somatórios se necessário)
 - a) $T(n) = T(n - 2) + n^2$
 - b) $T(n) = 3T(n - 1) + 1$
 - c) $T(n) = 2T(n/2) + 1$
 - d) $T(n) = 3T(n/2) + n$

Somatórios

$$\sum_{k=1}^n k = \frac{n(n+1)}{2}$$

$$\sum_{k=1}^n k^2 = \frac{n(n+1)(2n+1)}{6}$$

$$\sum_{k=1}^n k^3 = \frac{n^2(n+1)^2}{4}$$

$$\sum_{k=0}^n 2^k = 2^{n+1} - 1$$

$$\sum_{k=0}^n 3^k = \frac{3^{n+1} - 1}{3 - 1} = \frac{3^{n+1} - 1}{2}$$

$$\sum_{k=1}^n \log(k) = \log(n!) \approx n \log(n) - n$$