



Universidade Federal de Viçosa – Campus UFV-Florestal  
Ciência da Computação – Projeto e Análise de Algoritmos  
Professor: Daniel Mendes Barbosa  
**Lista de Exercícios 1 – Complexidade de Algoritmos**

- 1) Um algoritmo que tenha complexidade exponencial necessariamente é um algoritmo inútil? Explique.
- 2) Quando um algoritmo possui um pior caso muito caro, ele ainda assim pode ser útil independente do tamanho da entrada de dados?
- 3) Como podemos modificar quase que qualquer algoritmo para ter um bom tempo de execução para o melhor caso? Ou seja, o melhor caso é uma boa medida de complexidade de tempo para um algoritmo?
- 4) O que significa dizer que uma função  $g(n)$  é  $O(f(n))$ ?
- 5) Suponha um algoritmo A e um algoritmo B com funções de complexidade de tempo  $a(n) = n^2 - n + 274$  e  $b(n) = 49n + 49$ , respectivamente. Determine quais são os valores de  $n$  pertencentes ao conjunto dos números naturais para os quais A leva menos tempo para executar do que B. O que esses valores representam numa análise comparativa entre esses dois algoritmos, uma vez que um deles é  $O(n^2)$  e o outro é  $O(n)$ ?
- 6) Sejam  $f(n)$  e  $g(n)$  funções assintoticamente não-negativas. Usando a definição básica da notação  $\Theta$ , prove que  $\max(f(n), g(n)) = \Theta(f(n) + g(n))$ .  
Dica: lembre-se da definição de  $\Theta$  e defina a função  $\max$  como  $h(n)$ , da seguinte forma:

$$h(n) = \begin{cases} f(n) & \text{if } f(n) \geq g(n) , \\ g(n) & \text{if } f(n) < g(n) . \end{cases}$$

- 7) Mostre que para quaisquer constantes reais  $a$  e  $b$ , tal que  $b > 0$ ,

$$(n + a)^b = \Theta(n^b).$$

- 8) Mostre se:

$$(a) \quad 2^{n+1} \stackrel{?}{=} O(2^n).$$

$$(b) \quad 2^{2n} \stackrel{?}{=} O(2^n).$$

- 9) Descreva um algoritmo com complexidade de tempo  $O(n \log n)$  que, dado um conjunto  $S$  de  $n$  inteiros e um outro inteiro  $x$ , determina se existe ou não dois elementos de  $S$  cuja soma seja exatamente  $x$ .