

# Análise de Algoritmos

## Prof. Marcelo Keese Albertini

### Faculdade de Computação - Universidade Federal de Uberlândia

### Lista de exercícios 0: Introdução

1. Informalmente, o que é um procedimento? O que é um algoritmo? Qual é a diferença entre procedimento e algoritmo? Dê um exemplo de procedimento que eventualmente resulta em resposta incorreto mas ainda assim é útil.
2. Para cada período de tempo disponível  $T_{max} \in \{1 \text{ segundo}, 1 \text{ minuto}, 1 \text{ hora}, 1 \text{ dia}, 1 \text{ ano}, 1 \text{ século}\}$ , determine o maior tamanho da instância de um problema que um algoritmo que executa em  $f(n) \in \{\lg n, \sqrt{n}, n, n \lg n, n^2, 2^n, n!\}$  microssegundos pode resolver.
3. Descreva o modelo computacional *Random Access Machine*. Porque ele é usado?
4. Escreva um algoritmo de ordenação Bubblesort. Faça a análise de pior e melhor caso do número de comparações e trocas para esse algoritmo.
5. Escreva o Insertion sort. Em quais situações o desempenho do Bubblesort pode ser considerado equivalente ao do Insertion sort?
6. Mostre que, no melhor caso, o Insertion sort tem tempo de execução em  $\Omega(n)$  mas não em  $\omega(n)$ .
7. Mostre que o Insertion sort não tem tempo de execução em  $\Theta(n^2)$ .
8. Mostre que o Insertion sort não tem tempo de execução em  $o(n^2)$ .
9. Analise o algoritmo Counting Sort considerando operações de acesso a arrays. Para quais tamanho de instâncias  $N$  e quais casos (pior ou melhor) o algoritmo Bubblesort pode usar menos operações de acesso a vetores?
10. Expresse a função  $n^3/1000 - 100n^2 - 100n + 3$  em notação  $\Theta$ .
11. É verdade que se  $f(n) = O(g(n))$  então  $g(n) = O(f(n))$ ?
12. É verdade que se  $f(n) = O(g(n))$  então  $g(n) = \Omega(f(n))$ ?
13. É verdade que se  $f(n) = O(g(n))$  então  $g(n) = o(f(n))$ ?
14. É verdade que se  $f(n) = O(g(n))$  então  $g(n) = \omega(f(n))$ ?
15. Mostre que  $f(n) = n \lg n + O(n)$  implica em  $f(n) = \Theta(n \log n)$ .
16. Indique com “sim” ou “não”, na tabela a seguir, quais pares de expressões  $(A, B)$  em que  $A$  é  $o, O, \Omega, \omega, \theta$  de  $B$ . Considere que  $k \geq 1, \epsilon > 0$  e  $c > 1$  são constantes.

	A	B	$o$	$O$	$\Omega$	$\omega$	$\Theta$
a.	$\lg n$	$n^\epsilon$					
b.	$n^k$	$c^n$					
c.	$\sqrt{n}$	$n^{\sin n}$					
d.	$2^n$	$2^{n/2}$					
e.	$n^{\lg c}$	$c^{\lg n}$					
f.	$\lg(n!)$	$\lg(n^n)$					

17. Prove ou conteste as seguintes relações sobre a notação little-oh:

- $n^2 = o(n^3)$
- $3n = o(10n)$
- $\lg n = o(n)$

- $n \lg n = \Omega(n)$
- $n \lg n = \Omega(n^2)$
- $\lg(x + y) = O(\lg(x * y))$
- $\max(x, y) = O(x + y)$
- $\lg n^2 = o(\lg n^3)$
- se  $0 < p < q$ ,  $n^p = o(n^q)$
- se  $b > 1$  constante,  $b^n = o(n!)$
- $\sqrt{n} = o(n)$
- $n\sqrt{n} = o(n^2)$
- $\sqrt{n} \lg n = o(n \lg n)$
- $\lg n = o(\sqrt{n})$
- $n^{10} = o(2^n)$
- $10^{1.5n} = o(n!)$
- $3 = \lg n$
- $2 = (9/10)^n$
- $\frac{n^3}{10} = o(n^3)$
- $2^n = o(2^{n+1})$
- $6(\ln n)^3 = o(n)$
- $n^2 \lg n = o(n^3)$
- $\lg(n^3) = o(1.5)$
- $3^n 2\sqrt{n} = o(n^3)$
- $\lg n = o(\lg(2n))$
- $3^n = o(3^{2n})$
- $n^4 \sqrt{n} = o(n!)$
- $\sqrt{\lg n} = o(n^{2/5})$
- $n! = o((n + 1)!)$
- $5n^2 = o(n\sqrt{n})$
- $e^n = o(\pi^n)$
- $\lg(e^n) = o(\lg(\pi^n))$
- $10 \lg n = o(2^n)$
- $\sqrt{n \lg n} = o(n)$
- $\ln(n!) = o(n \ln(n))$

**18.** Ordene usando a notação  $o(\cdot)$ :

$$\sqrt{n} \lg n \quad \lg(n^{10}) \quad \sqrt{n!} \quad \frac{1}{\lg n} \quad n^2 \lg n \quad n^{5/2} \quad [\lg n]^2 \quad [\lg 10]^n$$

19. Prove ou conteste as seguintes relações:

- $n^2 \sim n^2/2$
- $n^2 \sim n^2 + 10n$
- $2^{n+1} = O(2^n)$
- $2^{2n} = O(2^n)$
- $\log(n!) = \Theta(n \log n)$
- $O(\log_{\sqrt{N}} N) = O(1)$
- $\sqrt{N(N-1)(N-2)} = o(N^2)$
- $(n+a)^b = \Theta(n^b)$  para constantes  $a, b \in \mathbb{R}^+$
- $o(g(n)) \cap \omega(g(n)) = \emptyset$

20. Em quais condições é verdade que se  $f(n) = O(g(n))$  então  $g(n) = O(f(n))$ ?

21. Ordene as seguintes funções por ordem de crescimento, ou seja, encontre uma lista ordenada  $g_1, g_2, g_3, \dots, g_{32}$  das funções satisfazendo  $g_1 = \Omega(g_2), g_2 = \Omega(g_3), \dots, g_{31} = \Omega(g_{32})$ .

Divida essa lista em subgrupos tal que  $f(n)$  e  $g(n)$  pertencem aos mesmo grupo se e somente se  $f(n) = \Theta(g(n))$ .

$\lg(\lg^* n)$	$2^{\lg^* n}$	$(\sqrt{2})^{\lg n}$	$n^2$	$n!$	$(\lg n)!$	$(3/2)^n$	$n^3$
$\lg^2 n$	$\lg(n!)$	$2^{2^n}$	$n^{1/\lg n}$	$\ln \ln n$	$\lg^* n$	$n2^n$	$n^{\lg \lg n}$
$\ln n$	1	$2^{\lg n}$	$e^n$	$4^{\lg n}$	$(n+1)!$	$\sqrt{\lg n}$	$\lg^*(\lg n)$
$2^{\sqrt{2} \lg n}$	$n$	$2^n$	$n \lg n$	$2^{2^{n+1}}$	$2^{n2^n}$	$(\lg n)^{\lg n}$	$n/\sqrt{n}$

22. Dê um exemplo do porquê a afirmação “O tempo de execução do algoritmo A é pelo menos  $O(n^2)$ ” é absurda.