MAC5711 - Análise de Algoritmos

Departamento de Ciência da Computação

Segundo semestre de 2018

Lista 1

1. Lembre-se que $\lg n$ denota o logaritmo na base 2 de n. Usando a definição de notação O, prove que

   (a) 3^n não é $O(2^n)$

   (b) $n \lg^5 n$ é $O(n^2)$ mas n^2 não é $O(n \lg^5 n)$

   (c) $\log_{10} n$ é $O(\lg n)$

   (d) $n^2 + 10n + 20 \;=\; O(n^2)$

   (e) $\lceil n/3 \rceil \;=\; O(n)$

   (f) $\lg n \;=\; O(\log_{10} n)$

   (g) $n \;=\; O(2^n)$

   (h) $n/1000$ não é $O(1)$

   (i) $n^2/2$ não é $O(n)$

2. Prove ou dê um contra-exemplo para as afirmações abaixo:

   (a) $\lg \sqrt{n} \;=\; O(\lg n)$

   (b) Se $f(n) = O(g(n))$ e $g(n) = O(h(n))$ então $f(n) = O(h(n))$.

   (c) Se $f(n) = O(g(n))$ e $g(n) = \Theta(h(n))$ então $f(n) = \Theta(h(n))$.

   (d) Suponha que $\lg(g(n)) > 0$ e que $f(n) \geq 1$ para todo n suficientemente grande. Neste caso, se $f(n) = O(g(n))$ então $\lg(f(n)) = O(\lg(g(n)))$.

   (e) Se $f(n) = O(g(n))$ então $2^{f(n)} = O(2^{g(n)})$.

3. Prove que

   (a) $\sum_{i=1}^{n} i^k$ é $\Theta(n^{k+1})$

   (b) $\sum_{i=1}^{n} \frac{i}{2^i} \leq 2$.

4. CLRS 3.1-1, 3-1, 3-2, 3-3, 3-4

   Vale a pena fazer tudo isso, e até recomendo que seja em grupo. É bom *aprender* as afirmativas da 3-1 e da 3-4 (não precisa ter as demonstrações na ponta da língua). A 3-2 e a 3-3 são questões abertas; um bom jeito de fazer é chutar cada caso, para ver se está sendo formada uma intuição – mas é importante conferir com rigor se a intuição está certa.