

MAC 338 – Análise de Algoritmos

2ª Lista de Exercícios 1.0 – 17/3/2015 – Entrega 7/4/2015

Obs.: esta é a versão 1.0. Aguarde uma nova versão com pelo menos uma questão adicional.

1. Resolva as recorrências abaixo, estimando o crescimento de $T(n)$ tentando ser o mais preciso possível (ou seja, não vale dar $O(n^3)$ se for $O(n^2)$). Suponha, em cada caso, que $T(n)$ é limitado para valores pequenos de n . É suficiente que sua solução valha para uma sequência infinita, crescente, de inteiros n .

(a) $T(n) = 2T(n/2) + n^2$

(b) $T(n) = T(7n/10) + n$

(c) $T(n) = 16T(n/4) + n^2$

(d) $T(n) = 7T(n/2) + n^2$

(e) $T(n) = 2T(n/4) + \sqrt{n}$

(f) $T(n) = 4T(n/3) + n \lg n$

(g) $T(n) = 4T(n/2) + n^2\sqrt{n}$

(h) $T(n) = \sqrt{n}T(\sqrt{n}) + n$

2. Seja A um vetor do *TipoOrd* com n elementos (indexado a partir de 0 ou de 1, conforme seu gosto), e suponha que os elementos sejam todos distintos. Um par (i, j) de índices é uma *inversão* em A se $i < j$ e $A[i] > A[j]$, ou seja se os elementos correspondentes estão fora de ordem entre si. Pela definição, existe um algoritmo trivial para contar o número de inversões de um dado A em tempo $\Theta(n^2)$. Descreva um algoritmo que faça essa contagem em tempo $o(n^2)$. Demonstre de forma convincente que seu algoritmo é correto, e estime rigorosamente o tempo.