



IF969 Algoritmos e Estruturas de Dados — 2012.2

Prof. Paulo Fonseca

**PRIMEIRA PROVA
27 de Fevereiro de 2013**

- Esta prova contém 04 (quatro) questões.
- A duração da prova é de 02 (duas) horas.
- A detecção de cópia implicará na atribuição de nota 0 (zero) à prova.

QUESTÃO 1

Os algoritmos A e B requerem *exatamente* $T_A(n) = 10n \lg n + 20$ e $T_B(n) = 2n^2 + 3n$ operações elementares, respectivamente, para um entrada de tamanho n .

- Qual o algoritmo mais eficiente do ponto de vista assintótico? Escreva as ordens de complexidade de A e B em notação assintótica.
- Existe alguma entrada para a qual o algoritmo menos eficiente do ponto de vista assintótico é mais rápido? Se sim, dê um exemplo. Caso contrário, justifique sucintamente.

QUESTÃO 2

Represente o processo de ordenação do array

⟨ M E R G E S O R T ⟩

pelo Algoritmo Mergesort. Para tal, escreva o array parcialmente ordenado ao final da execução de cada chamada do procedimento recursivo. Obs.: Suponha os caracteres ordenados por ordem alfabética.

QUESTÃO 3

A *profundidade* de um nó N de uma árvore binária, $D(N)$, é definida como

$$D(N) = \begin{cases} 0, & \text{se } N = \text{raiz}, \\ 1 + D(\text{Pai}(N)), & \text{caso contrário.} \end{cases}$$

Escreva um algoritmo em pseudo-código que recebe um apontador para a raiz de uma árvore binária e imprime a profundidade dos seus nós em *★pré-ordem★*.

QUESTÃO 4

Seja T uma árvore AVL, inicialmente vazia.

- Represente a inserção em T dos elementos

3, 2, 8, 1, 5, 4, 6, 9, 7,

nesta ordem. Se uma inserção não necessitar rotações, basta acrescentar o elemento à árvore corrente no local adequado (i.e. não precisa re-desenhar toda a árvore). Caso a inserção necessite de rotações, desenhe a árvore logo após a inserção (antes das rotações) e após cada rotação necessária.

- Represente a remoção dos elementos

6, 2,

nesta ordem, da árvore resultante do item anterior. Valem as mesmas observações sobre a representação das rotações.