

MAC 5711 - Análise de Algoritmos

Departamento de Ciência da Computação

Segundo semestre de 2018

Lista 2

1. Resolva as recorrências abaixo.

(a) $T(n) = 2T(\lfloor n/2 \rfloor) + \Theta(n^2)$

(b) $T(n) = 8T(\lfloor n/2 \rfloor) + \Theta(n^2)$

(c) $T(n) = 2T(\lfloor n/2 \rfloor) + \Theta(n^3)$

(d) $T(n) = 7T(\lfloor n/3 \rfloor) + \Theta(n^2)$

(e) $T(n) = T(\lfloor 9n/10 \rfloor) + \Theta(n)$

2. Escreva um algoritmo que ordena uma lista de n itens dividindo-a em três sublistas de aproximadamente $n/3$ itens, ordenando cada sublista recursivamente e intercalando as três sublistas ordenadas. Analise seu algoritmo concluindo qual é o seu consumo de tempo.

3. Seja $X[1..n]$ um vetor de inteiros e i e j dois índices distintos de X , ou seja, i e j são inteiros entre 1 e n . Dizemos que o par (i, j) é uma *inversão* de X se $i < j$ e $X[i] > X[j]$. Escreva um algoritmo $O(n \lg n)$ que devolva o número de inversões em um vetor X , onde n é o número de elementos em X .

4. Descreva um algoritmo que, dados inteiros n e k , juntamente com k listas ordenadas que em conjunto tenham n registros, produza uma única lista ordenada contendo todos os registros dessas listas (isto é, faça uma *intercalação*). O seu algoritmo deve ter complexidade $O(n \lg k)$. Note que isto se transforma em $O(n \lg n)$ no caso de n listas de 1 elemento, e em $O(n)$ se só houver duas listas (no total com n elementos).

5. Considere a seqüência de vetores $A_k[1..2^k], A_{k-1}[1..2^{k-1}], \dots, A_1[1..2^1]$, e $A_0[1..2^0]$. Suponha que cada um dos vetores é crescente. Queremos reunir, por meio de sucessivas operações de intercalação (= *merge*), o conteúdo dos vetores A_0, \dots, A_k em um único vetor crescente $B[1..n]$, onde $n = 2^{k+1} - 1$. Escreva um algoritmo que faça isso em $O(n)$ unidades de tempo. Use como subrotina o INTERCALE visto em aula.