MAC 5711 - Análise de Algoritmos

Departamento de Ciência da Computação

Segundo semestre de 2005

Lista 2

Devolução: 30 de agosto de 2005

---

1. Resolva as recorrências abaixo:

   (a) $T(n) = 8T(n/2) + \Theta(n^2)$
   (b) $T(n) = 2T(n/2) + \Theta(n^3)$
   (c) $T(n) = 7T(n/3) + \Theta(n^2)$
   (d) $T(n) = T(\lfloor 9n/10 \rfloor) + \Theta(n)$
   (e) $T(n) = T(\lfloor n/3 \rfloor) + T(\lfloor 2n/3 \rfloor) + \Theta(n)$
   (f) $T(n) = 2T(n/4) + \sqrt{n}$.

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. Escreva um algoritmo eficiente que busca um valor x em uma matriz $A_{n\times n}$ cujas linhas e colunas estão ordenadas em ordem não-decrescente. Seu algoritmo deve consumir tempo $O(n)$.

4. Escreva um algoritmo de divisão e conquista que determina o maior valor que aparece em um vetor. Escreva uma recorrência que descreve o número de comparações envolvendo elementos do vetor que tal algoritmo faz. Resolva a recorrência e compare o resultado com o número de comparações efetuada pelo algoritmo óbvio de cálculo do máximo de um vetor.

5. Considere o seguinte problema: dados n, uma seqüência de números reais $a_n, \ldots, a_0$ e um número real x, determinar o valor do polinômio $p_n(x) = a_n x^n + \cdots + a_1 x + a_0$. Pode-se resolver esse problema calculando o valor de $q_{n-1}(x) = a_n x^{n-1} + \cdots + a_2 x + a_1$ e, posteriormente, $p_n(x) = x\, q(x) + a_0$. Esse método é chamado de *regra de Horner*. Projete um algoritmo de divisão e conquista para resolver o problema e compare-o ao método derivado da regra de Horner. Quantas adições e quantas multiplicações faz o seu algoritmo?

6. Considere a recorrência

$$T(n) = \begin{cases} 1 & \text{se } n = 0 \text{ ou } n = 1 \\ \max_{0\le k\le n-1}\{T(k) + T(n-1-k)\} + n & \text{se } n \ge 2. \end{cases}$$

   Mostre que $T(n) \ge n^2/2$ para todo $n \ge 0$.

7. Considere a recorrência

$$T(n) = \begin{cases} 1 & \text{se } n = 0 \text{ ou } n = 1 \\ \min_{0\le k\le n-1}\{T(k) + T(n-1-k)\} + n & \text{se } n \ge 2. \end{cases}$$

   Mostre que $T(n) \le (n+1)\lg(n+1) + 1$ para todo $n \ge 0$.