Prova 1 – Projeto e Análise de Algoritmos (INE410104) – 20set2019

Programa de Pós-Graduação em Ciência da Computação – Universidade Federal de Santa Catarina

Prof. Rafael de Santiago

Estudante: ____

1. Considere os dois algoritmos a seguir que resolvem o mesmo problema. Os algoritmos recebem uma matriz quadrada $n \times n$, composta por números inteiros e devolvem (retornam) a maior soma de valores encontrada nos elementos de "janelas" (ou submatrizes) de 2×2. Para o algoritmo Me1, assuma que a primeira chamada é dada passando os parâmetros $(M, 1, n, 1, n)$. Assuma também que a matriz passada por parâmetro é indexada com valores entre 1 e n (intervalo fechado). Por questões de facilidade, assuma que n é uma potência de 2.

Me1(M, l_i, l_f, c_i, c_f)

se $l_f - l_i + 1 = 2$ **então**
    $s \leftarrow 0$
    para $i \leftarrow l_i$ **até** l_f **faça**
        para $j \leftarrow c_i$ **até** c_f **faça**
            $s \leftarrow s + m_{ij}$
    devolve s
$q \leftarrow l_f - l_i + 1$
$k \leftarrow \frac{q}{2}$
$a \leftarrow$ Me1$(M, l_i, l_i + k - 1, c_i, c_i + k - 1)$
$b \leftarrow$ Me1$(M, l_i, l_i + k - 1, c_i + k, c_f)$
$c \leftarrow$ Me1$(M, l_i + k, l_f, c_i, c_i + k - 1)$
$d \leftarrow$ Me1$(M, l_i + k, l_f, l_i + k - 1, c_i + k, c_f)$
devolve $\max\{a, b, c, d\}$

Me2(M)

1: $ms \leftarrow 0$
2: **para** $dl \leftarrow 1$ **até** $\frac{n}{2}$ **faça**
3:     **para** $dc \leftarrow 1$ **até** $\frac{n}{2}$ **faça**
4:         $s \leftarrow 0$
5:         **para** $i \leftarrow 2(dl - 1) + 1$ **até** $2dl$ **faça**
6:             **para** $j \leftarrow 2(dc - 1) + 1$ **até** $2dc$ **faça**
7:                 $s \leftarrow s + m_{ij}$
8:         **se** $s > ms$ **então**
9:             $ms \leftarrow s$
10: **devolve** ms

Pede-se:

(a) *(2,0pt)* Escreva a relação de recorrência para o algoritmo Me1 referente ao consumo de tempo computacional. Apresente a fórmula fechada e a complexidade, utilizando um dos métodos vistos em aula.

(b) *(0,5pt)* Escreva a função de complexidade de tempo para o algoritmo Me2. Seja o mais preciso possível ao representar o número de atribuições necessárias.

(c) *(1,5pt)* Para o algoritmo Me2, defina uma invariante de laço para as instruções FOR das linhas 2 e 3, relacionando o valor de ms ao maior valor encontrado para uma janela 2×2. Destaque a inicialização, manutenção e término para a invariante.

(d) *(1,0pt)* Compare a eficiência dos algoritmos Me1 e Me2 em relação ao tempo computacional requerido.

2. *(2,0pt)* Determine a **fórmula fechada**, com respectiva **prova por indução**, e **complexidade** de um algoritmo, considerando:

$$T(n) = T(n-2) + n$$

3. Uma das formas de se calcular um determinante para uma matriz A de ordem $n \times n$ é

$$det(A) = \sum_{j=1}^{n} (-1)^{1+j} \cdot a_{1j} \cdot det(A_{-1,-j}),$$

na qual

$$A_{-1,-2} = \begin{pmatrix} a_{21} & a_{23} & \dots & a_{2n} \\ a_{31} & a_{33} & \dots & a_{3n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n3} & \dots & a_{nn} \end{pmatrix}$$

por exemplo.

Pede-se:

(a) *(2,5pt)* Elabore um algoritmo de divisão e conquista que resolva o determinante de uma matriz $n \times n$ utilizando qualquer linguagem de programação, considerando a forma acima.

(b) *(0,5pt)* Determine a recorrência representando o consumo de tempo computacional do algoritmo desenvolvido nessa questão para o cálculo do determinante.

Boa prova!

Formulário:

A. Teorema mestre (Cormen et al) $\quad T(n) = aT(\frac{n}{b}) + f(n) \quad a \geq 1 \quad b > 1$

(1) Se $\quad f(n) \in O(n^{(\log_b a)-\epsilon})$ para $\epsilon > 0$ $\Rightarrow$ $T(n) \in \Theta(n^{\log_b a})$

(2) Se $\quad f(n) \in \Theta(n^{\log_b a})$ $\Rightarrow$ $T(n) \in \Theta(n^{\log_b a} \log n)$

(3) Se $\quad f(n) \in \Omega(n^{(\log_b a)+\epsilon})$ para $\epsilon > 0$
e se $\quad af(\frac{n}{b}) \leq cf(n)$ para $c < 1$
e $\quad n$ suficientemente grande $\Rightarrow$ $T(n) \in \Theta(f(n))$

B. Logaritmos

(1) Produto: $\log_b(x \cdot y) = \log_b x + \log_b y$

(2) Quociente: $\log_b(\frac{x}{y}) = \log_b x - \log_b y$

(3) Potência: $\log_b a^n = n \log_b a$

(4) Mudança de base: $\log_b a = \frac{\log_c a}{\log_c b}$

(5) Mudança de expoente: $a^{\log_b n} = n^{\log_b a}$

(6) Inversa: $b^{\log_b n} = n, \quad n > 0$

(7) Notação: $\log_2 x = \lg x$

C. Radiciação

- $\sqrt[a]{x^b} = x^{(\frac{b}{a})}$

D. Séries

(1) Soma de PA: $\sum_{i=0}^{n-1} (a_1 + i \cdot r) = \frac{n(a_1 + [a_1 + (n-1)r])}{2}$

(2) Soma de PG: $\sum_{i=0}^{n-1} (a_1 \cdot q^i) = \frac{a_1(q^n - 1)}{q-1}$

(3) Soma de PG infinita: $S_\infty = \frac{a_1}{1-q}$, para $|q| < 1$

E. Crescimento de funções (c e k constantes, $k \geq 2$)

- $c < \log n < n < n \log n < n^k < c^n$