Lista 2 - MAC0338 Análise de Algoritmos

Daniel Angelo Esteves Lawand 10297693

17 de setembro de 2021

Exercício 1

b)

Para realizar a recorrência de $T(n) = 8T(\lfloor \frac{n}{2} \rfloor) + \Theta(n^2)$, tomemos $n = 2^k$, $\forall k \geq 0$. Assim, temos:

$$T(n) = 8 \cdot T\left(\frac{n}{2}\right) + c_2 \cdot n^2$$

$$= 8 \cdot \left(8 \cdot T\left(\frac{n}{2^2}\right) + c_2 \cdot \left(\frac{n}{2}\right)^2\right) + c_2 \cdot n^2$$

$$= (2^3)^2 \cdot \left(8 \cdot T\left(\frac{n}{2^3}\right) + c_2 \cdot \left(\frac{n}{2^2}\right)^2\right) + 2 \cdot c_2 \cdot n^2 + c_2 \cdot n^2$$

$$= (2^3)^3 \cdot T\left(\frac{n}{2^3}\right) + 4 \cdot c_2 \cdot n^2 + 2 \cdot c_2 \cdot n^2 + c_2 \cdot n^2$$

$$= (2^3)^k \cdot T\left(\frac{n}{2^k}\right) + \sum_{i=0}^{k-1} 2^i \cdot c_2 \cdot n^2$$

$$= (2^3)^k \cdot T\left(\frac{n}{2^k}\right) + (2^k - 1) \cdot c_2 \cdot n^2$$

Como $n = 2^k$,

$$T(n) = n^3 \cdot T(1) + (n-1) \cdot c_2 \cdot n^2 \iff$$

$$\iff T(n) = n^3 \cdot c_1 + (n-1) \cdot c_2 \cdot n^2 \iff$$

$$\iff T(n) = n^3 \cdot c_1 + (n^3 - n^2) \cdot c_2$$

Vamos conferir que T(n) = $\begin{cases} c_1, \ n = 1 \\ 8 \cdot T\left(\frac{n}{2}\right) + c_2 \cdot n^2, \ n > 1 \end{cases}$ vale $T(n) = n^3 \cdot c_1 + (n^3 - n^2) \cdot c_2$

Indução em $k = 0 \Rightarrow n = 1$

$$T(1) = 1^3 \cdot c_1 + (1^3 - 1^2) \cdot c_2 = c_1 + 0 \cdot c_2 = c_1$$

Para $k > 1$ e valendo para $T(2^k)$, temos:

$$T(2^{k+1}) = 2^3 \cdot T\left(2^k\right) + c_2 \cdot (2^{k+1})^2$$

$$= 2^3 \cdot \left(2^{3k} \cdot c_1 + c_2 \cdot \left(2^{3k} - 2^{2k}\right)\right) + c_2 \cdot (2^{k+1})^2$$

$$= 2^{3k+3} \cdot c_1 + c_2 \cdot \left(2^{3k+3} - 2^{2k+3}\right) + c_2 \cdot 2^{2k+2}$$

$$= 2^{3k+3} \cdot c_1 + c_2 \cdot \left(2^{3k+3} - 2^{2k+3} + 2^{2k+2}\right)$$

$$= 2^{3k+3} \cdot c_1 + c_2 \cdot \left(2^{3k+3} - 2^{2k+2} \cdot (2-1)\right)$$

$$= 2^{3k+3} \cdot c_1 + c_2 \cdot \left(2^{3k+3} - 2^{2k+2}\right)$$

$$= n^3 \cdot c_1 + c_2 \cdot \left(n^3 - n^2\right)$$

$\square$

Portanto, como $T(n) = n^3 \cdot c_1 + c_2 \cdot \left(n^3 - n^2\right)$, podemos dizer que $T(n)$ é $\Theta(n^3)$

e)

Para realizar a recorrência de $T(n) = T(\lfloor \frac{9n}{10} \rfloor) + \Theta(n)$, tomemos $n = (10/9)^k$, $\forall k \geq 0$. Assim, temos:

$$T(n) = T\left(\frac{9n}{10}\right) + c_2 \cdot n$$

$$= \left(T\left(\frac{9^2 n}{10^2}\right) + c_2 \cdot \frac{9n}{10}\right) + c_2 \cdot n$$

$$= \left(T\left(\frac{9^3 n}{10^3}\right) + c_2 \cdot \frac{9^2 n}{10^2}\right) + c_2 \cdot \frac{9n}{10} + c_2 \cdot n$$

$$= T\left(\frac{9^3 n}{10^3}\right) + c_2 \cdot \frac{9^2 n}{10^2} + c_2 \cdot \frac{9n}{10} + c_2 \cdot n$$

$$= T\left(\frac{9^k n}{10^k}\right) + \sum_{i=0}^{k-1} \frac{9^i}{10^i} \cdot c_2 \cdot n$$

$$= T\left(\frac{9^k n}{10^k}\right) + \frac{1 - (\frac{9}{10})^k}{1 - \frac{9}{10}}$$

$$= T\left(\frac{9^k n}{10^k}\right) + 10 \cdot \left(1 - \left(\frac{9}{10}\right)^k\right)$$

Como $n = (10/9)^k \Rightarrow k = \log_{\frac{10}{9}} n$,

$$T(n) = T(1) + c_2 \cdot n \cdot 10 \cdot \left(1 - \frac{1}{n}\right)$$

$$= c_1 + c_2 \cdot 10 \cdot (n-1)$$

Vamos conferir que T(n) = $\begin{cases} c_1,\ n = 1 \\ T\left(\frac{9n}{10}\right) + c_2 \cdot n,\ n > 1 \end{cases}$ vale $T(n) = c_1 + c_2 \cdot 10 \cdot (n-1)$

Indução em $k = 0 \Rightarrow n = 1$

$$T(1) = c_1 + c_2 \cdot 10 \cdot (1-1) = c_1 + c_2 \cdot 0 = c_1$$

Para $k > 1$ e valendo para $T\left(\frac{10^k}{9^k}\right)$, temos:

$$T\left(\frac{10^{k+1}}{9^{k+1}}\right) = T\left(\frac{10^{k+1}}{9^{k+1}}\frac{9}{10}\right) + c_2 \cdot \frac{10^{k+1}}{9^{k+1}}$$

$$= T\left(\frac{10^k}{9^k}\right) + c_2 \cdot \frac{10^{k+1}}{9^{k+1}}$$

$$= \left(c_1 + c_2 \cdot 10 \cdot \left(\frac{10^k}{9^k} - 1\right)\right) + c_2 \cdot \frac{10^{k+1}}{9^{k+1}}$$

$$= c_1 + c_2 \cdot \left(\frac{10^{k+1}}{9^k} - \frac{9^k 10}{9^k}\right) + c_2 \cdot \frac{10^{k+1}}{9^{k+1}}$$

$$= c_1 + c_2 \cdot \left(\frac{10^{k+1}}{9^k} - \frac{9^k 10}{9^k} + \frac{10^{k+1}}{9^{k+1}}\right)$$

$$= c_1 + c_2 \cdot \left(\frac{10^{k+1} \cdot 9}{9^{k+1}} - \frac{9^{k+1} \cdot 10}{9^{k+1}} + \frac{10^{k+1}}{9^{k+1}}\right)$$

$$= c_1 + c_2 \cdot \left(\frac{10^{k+1} \cdot 10}{9^{k+1}} - 10 \cdot \frac{9^{k+1}}{9^{k+1}}\right)$$

$$= c_1 + c_2 \cdot 10 \cdot \left(\frac{10^{k+1}}{9^{k+1}} - \frac{9^{k+1}}{9^{k+1}}\right)$$

$$= c_1 + c_2 \cdot 10 \cdot (n-1)$$

□

Portanto, como $T(n) = c_1 + c_2 \cdot 10 \cdot (n-1)$, podemos dizer que $T(n)$ é $\Theta(n)$

Exercício 3

Fazendo a análise do algoritmo, podemos perceber que segue o mesmo gasto de tempo que o MergeSort. Portanto, como vimos em aula, o tempo do mergesort segue a seguinte função:

$$T(n) = \begin{cases} c_1,\ n = 1 \\ T(\lfloor \frac{n}{2} \rfloor) + T(\lceil \frac{n}{2} \rceil) + \Theta(n),\ n > 1 \end{cases}$$

O que equivale dizer:

$T(n) = \begin{cases} c_1, \ n = 1 \\ 2 \cdot T(\frac{n}{2}) + c_2 \cdot n, \ n > 1 \end{cases}$

Realizando essa recorrência:

$$T(n) = 2 \cdot T\left(\frac{n}{2}\right) + c_2 \cdot n$$

$$= 2 \cdot \left(2 \cdot T\left(\frac{n}{2^2}\right) + c_2 \cdot \frac{n}{2}\right) + c_2 \cdot n$$

$$= 2^2 \cdot \left(2 \cdot T\left(\frac{n}{2^3}\right) + c_2 \cdot \frac{n}{2^2}\right) + c_2 \cdot \frac{n}{2} + c_2 \cdot n$$

$$= 2^3 \cdot T\left(\frac{n}{2^3}\right) + 3 \cdot c_2 \cdot n$$

$$= 2^k \cdot T\left(\frac{n}{2^k}\right) + k \cdot c_2 \cdot n$$

Seja $n = 2^k \Rightarrow k = \log_2 n$,

$$T(n) = n \cdot T(1) + \log_2 n \cdot c_2 \cdot n \iff$$

$$\iff T(n) = n \cdot c_1 + \log_2 n \cdot c_2 \cdot n$$

Vamos conferir que $T(n) = \begin{cases} c_1, \ n = 1 \\ 2 \cdot T(\frac{n}{2}) + c_2 \cdot n, \ n > 1 \end{cases}$ vale $T(n) = n \cdot c_1 + \log_2 n \cdot c_2 \cdot n$

Indução em $k = 0 \Rightarrow n = 1$

$$T(1) = 1 \cdot c_1 + \log_2 1 \cdot c_2 \cdot 1 = c_1 + 0 = c_1$$

Para $k > 1$ e valendo para $T\left(2^k\right)$, temos:

$$T(2^{k+1}) = 2 \cdot T\left(\frac{2^{k+1}}{2}\right) + c_2 \cdot n$$

$$= 2 \cdot T\left(2^k\right) + c_2 \cdot 2^{k+1}$$

$$= 2 \cdot \left(2^k \cdot c_1 + \log_2 2^k \cdot c_2 \cdot 2^k\right) + c_2 \cdot 2^{k+1}$$

$$= \left(2^{k+1} \cdot c_1 + k \cdot c_2 \cdot 2^{k+1}\right) + c_2 \cdot 2^{k+1}$$

$$= 2^{k+1} \cdot c_1 + c_2 \cdot 2^{k+1} \cdot (k+1)$$

$$= n \cdot c_1 + c_2 \cdot n \cdot \log_2 n$$

$\square$

Portanto, como $T(n) = n \cdot c_1 + c_2 \cdot n \cdot \log_2 n$, podemos dizer que $T(n)$ é $\Theta(n \log_2 n)$