

Universidade Federal de Roraima
Departamento de Ciência da Computação
Análise de Algoritmos

Rodrigo dos Santos Tavares

Lista 1

Questão 1:

(a) $n + (\log n) = \Theta(n)$

$$0 \leq c_1 \times n \leq n + (\log n) \leq c_2 \times n$$

Para $n = 2$:

$$0 \leq 2c_1 \leq 2 + 1 \leq 2c_2$$

$$0 \leq 2c_1 \leq 3 \leq 2c_2$$

Para $c_1 = 1$ e $c_2 = 2$:

$$0 \leq 2 \leq 3 \leq 4$$

Sim, é verdadeira.

(b) $(n + 1)^2 = O(2n^2)$

$$0 \leq n^2 < n^3 \times c$$

Para $n = 2$ e $c = 1$:

$$0 \leq 4 < 8$$

(c) $(n + 1)^2 = O(2n^2)$

$$0 \leq (n + 1)^2 \leq 2n^2 \times c$$

Para $n = 3$:

$$0 \leq 16 \leq 18c$$

Para qualquer $c \geq 1$, a afirmação acima é verdadeira, portanto:

$$(n + 1)^2 \text{ é } O(2n^2)$$

(d) $n - 300 = \Theta(300n)$

$$0 \leq c_1 \times 300n \leq n - 300 \leq c_2 \times 300n$$

Para $n = 400$:

$$0 \leq 120.000c_1 \leq 100 \leq 120.000c_2$$

Questão 2:

O problema de ordenação consiste em, tendo como entrada uma lista de dados ordenáveis, resultar em uma lista de dados ordenados de forma crescente ou decrescente. Isto sugere que será necessário percorrer todos os valores da lista pelo menos uma vez.

Por exemplo, a lista (1,5,13,6,4) pode ser ordenada de maneira crescente como (1,4,5,6,13).

Os limites inferior e superior são, respectivamente, $\Omega(n)$ e $O(n^2)$.

Questão 3:

(a)

$$\sum_{l=1}^{10000} \sum_{i=1}^{n-5} \sum_{j=i+2}^{n/2} \sum_{k=1}^n = 10000n^2 - 50000n$$

(b)

$$T(n) = \begin{cases} 1, & n = 1 \\ 3T(n/3) + \frac{5n}{3} - 2 \end{cases}$$

$$\begin{aligned} T(n) &= 3^1 T(n/3^1) + 1 \times \frac{5n}{3} - 2 \times 3^0 = \\ T(n) &= 3^2 T(n/3^2) + 2 \times \frac{5n}{3} - 2 \times 3^1 - 2 \times 3^0 = \\ T(n) &= 3^3 T(n/3^3) + 3 \times \frac{5n}{3} - 2 \times 3^2 - 2 \times 3^1 - 2 \times 3^0 \end{aligned}$$

$$3^k T(n/3^k) + k \times \frac{5n}{3} - \sum_{i=0}^{k-1} 2 \times 3^i$$

$$k = \log_3 n$$

$$\begin{aligned} n + \log_3 n \times \frac{5n}{3} - 2 \log_3 n \times n = \\ n + n \log_3 n \times \frac{5}{3} - 2n \log_3 n \end{aligned}$$

$$T(n) \text{ é } O(n \log n)$$

(c)

$$\sum_{i=1}^{n-2} \sum_{j=i+1}^n \sum_{k=1}^j = \frac{n^3}{3} - \frac{3n^2}{2} + \frac{17n}{2} - 1$$

(d)

$$\begin{aligned} T(n) &= \begin{cases} 1, & n = 0 \\ 2T(n-1) + 1 \end{cases} \\ \dots \\ 2^k T(n-k) + \sum_{i=0}^{n-1} 2^i \end{aligned}$$

$$k = n$$

$$2^{n+1} - 1$$

Questão 6:

Para ordenar uma lista em ordem crescente com o algoritmo Quick Sort:

1. Escolhe-se um item a ser o pivô;
2. Posiciona-se os itens de forma que todos à esquerda do pivô sejam menores que ele, e todos à direita sejam maiores;
3. Executa-se o algoritmo recursivamente na lista de elementos à esquerda do pivô e na lista de elementos à direita.

Questão 7:

- A raiz deve ser preta;
- Um nó vermelho deve ter como filhos apenas nós pretos;
- Para qualquer nó da árvore, todos os caminhos até as folhas devem ter a mesma quantidade de nós pretos.

