



# **Estruturas de Dados Lineares**

## Introdução à Análise de Algoritmos

### Respostas

# Exercícios

1. Calcule a função  $T(n)$  para o algoritmo abaixo assumindo que é uma matriz quadrada. Apresente também a classe de complexidade.

```
def somar(a: np.matrix) -> int:
    somar: int = 0
    for i, _ in enumerate(a):
        for j, _ in enumerate(a[i]):
            somar += a[i][j]

    return somar
```

```
1
n + 1
n . (n + 1) => n² + n
n . n => n²
1
```

$$T(n) = 2n^2 + 2n + 3$$
$$\text{Classe} = O(n^2)$$

## Exercícios – cont.

2. Calcule a função  $T(n)$  para o algoritmo abaixo assumindo que é uma matriz quadrada. Apresente também a classe de complexidade.

```
def imprimir(a: np.matrix) -> None:
    i: int = 0
    while i < len(a):
        j: int = 0
        while j < len(a[i]):
            print(a[i][j], end=' ')
            j += 1
        print()
        i += 1
```

1  
n + 1  
n  
n . (n + 1) => n<sup>2</sup> + n  
n . n => n<sup>2</sup>  
n . n => n<sup>2</sup>  
n  
n

$$T(n) = 3n^2 + 5n + 2$$
$$\text{Classe} = O(n^2)$$

## Exercícios – cont.

3. Dois algoritmos A e B possuem complexidade  $n^5$  e  $2^n$ , respectivamente. Você utilizaria o algoritmo B ao invés do A em qual caso? Desenvolva um gráfico com análise dos dois algoritmos e justifique a sua resposta.

A quando  $n = 1$  ou  $n > 22$

B quando  $n = [2, 22]$

4. Podemos dizer que  $n = O(n^2)$ ? Justifique a sua resposta.

Resposta 1: o comportamento assintótico quadrático é superior (pior desempenho) que o linear, portanto, uma função linear é limitada superiormente ( $BigO$ ) por uma função quadrática. Basta verificar o gráfico de comportamentos assintóticos.

Resposta 2: usando a fórmula do  $BigO$ , temos:

$$0 \leq f(n) \leq c \cdot g(n)$$

$$0 \leq n \leq n^2 \text{ (simplificar)}$$

$0 \leq 1 \leq n$  (para todo  $n > 1$ , a função quadrática é superior, ou seja, ela tem o desempenho pior. Logo, a expressão acima é verdadeira.

## Exercícios – cont.

5. Assuma que cada expressão abaixo represente a função  $T(n)$  de um algoritmo que resolve um problema de tamanho “ $n$ ”. Portanto, defina a classe para cada expressão.

$T(n)$	$O(\dots)$
$10n + 0,01n^2$	$O(n^2)$
$2n + n^{1,25} + n^{0,5}$	$O(n^{1,25}) = \text{subquadrático}$
$10n + n\log_2(n)$	$O(n\log(n)) = \text{linearítmico}$
$10n\log_3(n) + n^3 + 10n$	$O(n^3)$
$\sqrt{n} + \log_2(n)$	$O(\sqrt{n}) = \text{sublinear (ou raiz)}$