

Universidade Federal de São Carlos – Departamento de Computação
Projeto e Análise de Algoritmos
Prof. Alexandre L. M. Levada

Atividade Avaliativa 1 – Análise Assintótica

1. Defina o que é a notação Big-O. Faça um gráfico ilustrativo para exemplificar sua definição.
2. Defina o que é a notação Ω . Faça um gráfico ilustrativo para exemplificar sua definição.
3. Defina o que é a notação Θ . Faça um gráfico ilustrativo para exemplificar sua definição.
4. Defina a função T(n) que conta quantas operações serão executadas pelo script Python a seguir. Calcule as notações Big-O, Ω e Θ . Explique como você obteve sua resposta.

```
def Algo_C(n):  
    a = 100  
    j = n  
    while j > 0:  
        k = 0  
        while k < j:  
            a = a + 10  
            k = k + 1  
        j = j - 1  
    return a
```

5. Considere o script em Python a seguir, em que a função fn1() possui complexidade O(1), a função fn2() possui complexidade O(n) e a função fn3() possui complexidade O(n²). Qual é a complexidade do algoritmo? Explique seu raciocínio.

```
for i in range(n):  
    fn1(i)  
    for j in range(n):  
        fn2(j)  
        for k in range(n):  
            fn3(k)
```

6. Mostre que se c é um número real positivo, então:

$$g(n) = \sum_{k=0}^n c^k$$

é:

- a) $\Theta(n)$, se $c = 1$
- b) $\Theta(c^n)$, se $c > 1$
- c) $\Theta(1)$, se $c < 1$

O que isso significa na prática? Explique.

7. Mostre que para quaisquer constantes reais $a, b > 0$

$$(n+a)^b \text{ é } \Theta(n^b)$$

8. Mostre que $f(n)=n!$ é $O(n^n)$.

9. Mostre que $f(n)=\log n!$ é $O(n \log n)$.

10. Para cada uma das funções a seguir, indique se $f=O(g)$, $f=\Omega(g)$ ou $f=\Theta(g)$, justificando cada uma das respostas:

	$f(n)$	$g(n)$
a)	$n^{1/2}$	$n^{2/3}$
b)	$10 \log n$	$\log n^2$
c)	\sqrt{n}	$(\log n)^3$
d)	$n^{0.01}$	$\log n$

"You will never speak to anyone more than you speak to yourself in your head. Be kind to yourself."
-- Author Unknown