

Conteúdo Principal:

- Análise e Complexidade - Análise Empírica e Matemática

1. O que significa dizer que uma função $g(n)$ é $O(f(n))$?
2. O que significa dizer que uma função $g(n)$ é $\Theta(f(n))$ (theta de n)?
3. O que significa dizer que uma função $g(n)$ é $\Omega(f(n))$ (ômega de n)?
4. Dois algoritmos A e B possuem complexidades n^5 e 2^n respectivamente. Você utilizaria o algoritmo B ao invés do A, em qual caso? Explique.
5. Pesquise sobre quais problemas costumam ser exponenciais. Comente sobre dois deles.
6. Indique a ordem de complexidade, no pior caso, das seguintes funções de custo:
 - a. $f(n) = 2n + 10$
 - b. $f(n) = 1/2n * (n + 1)$
 - c. $f(n) = 1/2 n^2$
 - d. $f(n) = 1/2 n^4 - 3n^2 + 5n + 7$
 - e. $f(n) = 7n + 3 \log_2(n) + 20$
 - f. $f(n) = n! + 5n^2 + 10$
 - g. $f(n) = 3 * 5000^n + 1000$
 - h. $f(n) = 10^{10}$
7. Calcule a complexidade, no pior caso, do seguinte fragmento de código:

```
int i, j, k;
for (i = 0; i < N; i++) {
    for (j=0; j < N; j++) {
        R[i][j] = 0;
        for (k=0; k < N; k++)
            R[i][j] += A[i][k] * B[k][j];
    }
}
```

8. Escreva um algoritmo que receba valores em um vetor e imprima “ORDENADO” se o vetor estiver em ordem crescente. Considerando o código desenvolvido, indique a função de custo de pior caso e sua ordem de complexidade O . Lembre de anexar o código do algoritmo.
9. Escreva um algoritmo que procure por um dado número em vetor ordenado. Considerando o código desenvolvido, indique a função de custo de pior caso e sua ordem de complexidade O e Ω .