

Análise de algoritmos

Notação Big O

- A notação Big O é uma ferramenta fundamental na análise de algoritmos e na compreensão do seu desempenho em termos de tempo de execução e uso de recursos. Algumas das propriedades principais da notação Big O são:

1) Definição:

A notação Big O, denotada como $O(f(n))$, descreve o limite superior assintótico do comportamento de uma função em termos de outra função. Em termos mais simples, ela fornece uma maneira de comparar a taxa de crescimento de duas funções à medida que o tamanho da entrada (n) cresce

2) Limite Superior:

A notação Big O descreve um limite superior assintótico para o tempo de execução de um algoritmo. Isso significa que para uma determinada função $f(n)$, $O(g(n))$ representa o conjunto de funções que crescem tão rapidamente quanto ou mais lentamente que $g(n)$.

3) Simplificação:

A notação Big O é usada para simplificar a análise de desempenho de algoritmos, especialmente em casos onde o tempo de execução exato é difícil de determinar. Ela se concentra na taxa de crescimento dominante da função à medida que o tamanho da entrada aumenta

4) Ignora Constantes:

A notação Big O ignora constantes multiplicativas, focando apenas na parte dominante do crescimento da função. Por exemplo, se um algoritmo tem um tempo de execução de $5n^2 + 3n + 2$, a notação Big O seria $O(n^2)$, pois a contribuição dominante para o tempo de execução é n^2 . Ou seja, para calcular o tempo de execução, considera-se apenas o termo dominante

5) Simples comparação:

A notação Big O permite uma comparação simples entre algoritmos, ajudando a identificar qual algoritmo é mais eficiente em termos de tempo de execução para um determinado conjunto de dados de entrada.

6) Tipos Comuns:

Alguns dos tipos comuns de notação Big O incluem $O(1)$ para tempo de execução constante, $O(\log n)$ para tempo de execução logarítmico, $O(n)$ para tempo de execução linear $O(n^2)$ para tempo de execução quadrático, e assim por diante

7) Notação Assintótica:

A notação Big O é uma notação assintótica, o que significa que ela descreve o comportamento de uma função à medida que o tamanho da entrada se aproxima do infinito. Isso pode não ser representativo do comportamento real do algoritmo para entradas de tamanho pequeno ou moderado

8) Worst-case analysis: A notação Big O é comumente usada para análise de pior caso de algoritmos, embora também possa ser usada para análises de melhor caso e médio caso, mas com notações diferentes (Ω – ômega e Θ – theta)

Pior caso, Melhor caso e Caso médio

A análise de algoritmos considera três cenários possíveis, os quais relacionam a entrada com o número de iterações do algoritmo

- O pior caso acontece quando o algoritmo executa o número máximo de iterações possível
- No melhor caso o algoritmo executa o número mínimo de iterações possível
- O caso médio representa o cenário esperado quando as entradas possuem determinada distribuição de probabilidade de ocorrência

OBSERVAÇÃO: A fórmula para o caso médio coincide com a definição básica probabilística de valor esperado

- Embora o caso médio seja mais próximo da realidade, sua análise é mais técnica e depende de conceitos elaborados de matemática e probabilidade
- Além disso, o caso médio “tende” a ser idêntico ao pior caso no contexto da complexidade assintótica
- A notação mais utilizada é a notação Big O, seguida pela notação Big Θ