# Big O notation

Big O notation is a mathematical way to describe the time complexity and space complexity as the input size grows. It helps developers compare algorithms and predict performance at scale.

this is the way BIG O notations are Analyzing the Algorithms:

1. Identify the dominant operation (the part that runs the most).

2. Count how many times it executes relative to input size (n).

3. Drop constants and lower-order terms (e.g., O(2n + 3) → O(n)).

**Search Operation Scenarios:**

The element is found immediately (first element checked)

Linear search: O(1)

Binary search: O(1)

Average-case: The element is found somewhere in the middle

Linear search: O(n)

Binary search: O(log n)

Worst-case: The element isn't present or is at the end

Linear search: O(n)

Binary search: O(log n)