**Understanding Asymptotic Notation**

**Big O Notation**:

Big O notation describes the upper bound of an algorithm’s runtime, providing a worst-case scenario for how an algorithm performs as the input size grows.

It helps in analyzing the efficiency of algorithms by comparing their time complexity.

**Best, Average, and Worst-Case Scenarios**:

Best Case: The scenario where the algorithm performs the minimum number of operations.

Average Case: The scenario where the algorithm performs an average number of operations.

Worst Case: The scenario where the algorithm performs the maximum number of operations.

**Implementation**

**Linear Search**:

Linear search checks each element in the array until it finds the target element.

Time Complexity: O(n)

**Binary Search**:

Binary search works on a sorted array by repeatedly dividing the search interval in half.

Time Complexity: O(log n)

**Analysis**

**Time Complexity Comparison**:

* **Linear Search**: O(n) - It checks each element one by one, making it less efficient for large datasets.
* **Binary Search**: O(log n) - It divides the search space in half each time, making it much faster for large datasets.

**Suitability**:

* **Binary Search** is more suitable for the e-commerce platform due to its faster performance with large datasets. However, it requires the array to be sorted, which adds an additional step.