# **Analysis:**

## **Compare the time complexity of linear and binary search algorithms.**

**Time Complexity Comparison**

Linear Search

* Time Complexity: O(n)
* Explanation: Checks each element one by one until the target is found or the end of the array is reached.
* Use Case: Suitable for small or unsorted arrays.

Binary Search

* Time Complexity: O(logn)
* Explanation: Repeatedly divides the sorted array in half to locate the target.
* Use Case: Ideal for large, sorted arrays due to its efficiency.

Summary

* Linear Search: O(n) - Simpler, no sorting required.
* Binary Search: O(logn)- Faster for large datasets, requires sorted array.

## **Discuss which algorithm is more suitable for your platform and why.**

Suitable Search Algorithm for E-commerce Platform

Binary Search is more suitable for the e-commerce platform because:

* Efficiency: Handles large datasets quickly (O(log n) time complexity).
* Scalability: Provides faster search results as the product catalog grows.
* Sorted Data: E-commerce platforms often sort products, making binary search feasible.

Implementation Strategy:

* Initial Sort: Sort product list when loading the catalog.
* Maintain Sorted Order: Keep products sorted for efficient searches.

Conclusion: Binary search is preferred for its efficiency and speed, essential for large, dynamic product catalogs on an e-commerce platform.