Amrita Kumari (6362289)

WEEK-1
Data Structure and Algorithm
Exercise 2: E-commerce Platform Search Function

Understand Asymptotic Notation

Big O Notation

It describes the upper bound of the running time of an algorithm in terms of the input size. It helps you understand how your algorithm scales.
1. Linear Search: O(n)- Time grows linearly with input.
2. Binary Search: O(log n)- Time increases slowly, following a logarithmic pattern

Best, Average, Worst Cases:

Algorithm	**Best Case**	**Average Case**	**Worst Case**
Linear Search	O(1) (first match)	O(n/2) ≈ O(n)	O(n) (last/no match)
Binary Search	O(1) (middle match)	O(log n)	O(log n)

Analyzing

Time Complexity Comparison

Algorithm	**Time Complexity**	**Space Complexity**
Linear Search	O(n)	O(1)
Binary Search	O(log n)	O(1)

Which is better ?

1. Linear Search:
- No sorting required.
- Works on unsorted data.
- Slower for lager datasets.

2. Binary Search:
- Requires sorted data.
- Much faster on large, sorted data.

For a real-world e-commerce platform with large product data, Binary Search is more efficient—assuming the product list is sorted by productId. Sorting can be done once, and updates can be handled incrementally.