# **Exercise 2: E-commerce Platform Search Function**

# **1. Explain Big O Notation and how it helps in analyzing algorithms**

## **Big O notation is a mathematical way to describe the efficiency of an algorithm in terms of time or space as the input size grows. It helps us understand how well an algorithm performs, especially for large inputs.**

## **For example, if an algorithm is O(n), it means the time it takes grows linearly with input size n. Big O ignores constants and focuses only on the growth rate, which is useful for comparing different algorithms objectively.**

## **It helps developers choose the most optimal algorithm by analyzing:**

## **Time complexity (how fast it runs)**

## **Space complexity (how much memory it uses)**

## **2. Desribe the best, average, and worst-case scenarios for search operations**

|  |  |
| --- | --- |
| **Best Case -** | **The item is found at the first position (minimum time).** |
| **Average - Case** | **The item is somewhere in the middle or randomly placed.** |
| **Worst Case -** | **The item is at the last position or not present at all (maximum time).** |

## **For example, in linear search:**

## **Best: O(1)**

## **Average: O(n/2) → O(n)**

## **Worst: O(n)**

## **In binary search (on sorted data):**

## **Best: O(1)**

## **Average: O(log n)**

## **Worst: O(log n)**

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

## **For an e-commerce platform, performance and scalability are critical. Since the product list can grow large and users expect fast search results, binary search is more suitable because of its O(log n) performance.**

## **However, binary search requires the list to be sorted. In real-world applications, data is often indexed and sorted in the backend (e.g., using databases or search engines), so binary or even faster search techniques like hash indexing or tries are used.**

## **If sorting is not feasible or the dataset is small, linear search might be acceptable. But for optimized performance, binary search is the preferred method when working with sorted data.**