**E-Commerce Platform Search Function - Document**

**1. What is Big O Notation?** Big O notation describes the time or space complexity of an algorithm in terms of input size. It helps estimate performance and scalability.

**2. Best, Average, and Worst Case Scenarios for Search:** - **Linear Search:** - Best: O(1) - Item found at the beginning. - Average: O(n) - Worst: O(n) - Item not found or last in list.

* **Binary Search (on sorted data):**
  + Best: O(1)
  + Average: O(log n)
  + Worst: O(log n)

**3. Product Class Setup:** Each product contains: - productId (int) - productName (String) - category (String)

**4. Data Structures Used:** - Products stored in an array. - Array allows both linear and binary search.

**5. Linear Search Implementation:** - Loops through the array. - Compares product name with target. - Time Complexity: O(n)

**6. Binary Search Implementation:** - Array must be sorted by product name. - Uses divide-and-conquer. - Time Complexity: O(log n)

**7. Sorting Before Binary Search:** - Used Arrays.sort() with a custom Comparator to sort by name. - Sorting Time Complexity: O(n log n)

**8. Comparison of Linear vs Binary Search:** - Linear is simple and works on unsorted data. - Binary is much faster but requires sorting first. - For large datasets, binary search is preferred for performance.

**9. Which is More Suitable?** - **Binary Search** is more suitable when the data is large and mostly read-only. - **Linear Search** is better when data is small or frequently changing.