**Understand Asymptotic Notation**

**Big O Notation:**

* Big O notation describes the upper bound of an algorithm's running time as the input size n grows.
* It helps in evaluating the efficiency and scalability of algorithms without relying on hardware or environment.

**Case Scenarios for Search Operations:**

|  |  |
| --- | --- |
| **Best Case:** | The desired item is found in the first comparison. |
| **Average:** | The item is located somewhere in the middle. |
| **Worst Case:** | The item is found at the last index or not found. |

**Time Complexity Comparison:**

| **Algorithm** | **Best Case** | **Average Case** | **Worst Case** |
| --- | --- | --- | --- |
| **Linear Search** | O(1) | O(n) | O(n) |
| **Binary Search** | O(1) | O(log n) | O(log n) |

**Conclusion:**

Binary Search is more suitable because it is much faster for large product datasets due to O(log n) time given that the index is always sorted. Even in the worst case, it performs well. Common tasks like finding a product by ID benefit from binary search.