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

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

**Ans->** A mathematical representation that describes the upper limit of an algorithm's running time. It helps in analyzing the efficiency of algorithms by providing a way to express the worst-case scenario in terms of input size.

1. **Describe the best, average, and worst-case scenarios for search operations.**

**Ans->** **Best Case**: The minimum time required for an algorithm to complete. For a search operation, the best case is O(1) when the element is found at the first position.

**Average Case**: The expected time required for an algorithm to complete, averaged over all possible inputs. For linear search, it's O(n/2), which simplifies to O(n).

**Worst Case**: The maximum time required for an algorithm to complete. For linear search, it's O(n), and for binary search, it's O(log n).

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

**Ans-> Linear Search**: O(n) - Scans each element until the target is found or the end is reached.

**Binary Search**: O(log n) - Divides the array into halves, reducing the search space by half each time.

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

**Ans-> Linear Search**: Suitable for small datasets or unsorted arrays.

**Binary Search**: More suitable for large datasets due to its logarithmic time complexity but requires the array to be sorted.

**OUTPUT OF LINEAR AND BINARY SEARCHING OF PRODUCTS –**

