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

**Big O notation and why it's important for the analysis of algorithms:**

Mathematically, Big O notation defines the limiting behavior of a function while an argument tends to a particular value or infinity. In the context of algorithms, Big O notation is applied to analyze the efficiency or complexity of an algorithm with respect to its input size.

**Best Case:** The number of operations performed by the algorithm is at its minimum. That means when the target element is at the head of the list, in the case of a linear search. Average Case: The average number of operations the algorithm performs over all possible inputs of a given size.

**Worst Case:** It is the maximum number of operations the algorithm performs for any input of a given size. It is a good measure in that it allows one to know how bad the performance may turn out to be regarding the algorithm.

**Time Complexity Comparison**

* **Linear Search**:
  + **Best Case**: O(1) (when the target is found at the beginning)
  + **Average Case**: O(n)
  + **Worst Case**: O(n) (when the target is at the end or not present)
* **Binary Search**:
  + **Best Case**: O(1) (when the target is found at the middle)
  + **Average Case**: O(log n)
  + **Worst Case**: O(log n) (when the target is not present or at the ends)

**Which algorithm would be more appropriate for your platform, and why?**

**Binary Search** is, in general, more appropriate for the search functionality of the e-commerce platform. This is because it has an average and worst-case time complexity of O(log n), which really matters when the number of products (n) is large.

**Linear Search** has a linear time complexity in the average case, O(n), which degrades linearly with an increase in the number of products, hence slower search performance, especially with big datasets.

**Requirement of a Sorted Array:** Binary search requires that the array should be sorted, and this can be maintained in two ways: by keeping the array sorted all the time in case frequent insertions and deletions are not needed, or by sorting it before the searches are made.