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

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

Big O notation is used to describe the time complexity of an algorithm — that is, how the run time grows as the size of the input increases.  
It helps us analyze and compare the efficiency of algorithms, especially for large datasets.

* **Describe the best, average, and worst-case scenarios for search operations.**
  + Linear Search:
    - Best case: O(1) — target found at the first position
    - Average case: O(n/2) → O(n)
    - Worst case: O(n) — target not found or at last position
  + Binary Search:
    - Best case: O(1) — target found at middle position
    - Average case: O(log n)
    - Worst case: O(log n) — must divide repeatedly until 1 element left
* **Compare the time complexity of linear and binary search algorithms**
  + Linear search works on **unsorted** data — no pre-processing needed.
  + Binary search requires **sorted** data — but is much faster for large datasets: O(log n) vs O(n).
  + For an e-commerce platform, where millions of products may be searched, **binary search is more suitable** — provided the products can be sorted by the search field (ex: productName).
* **Discuss which algorithm is more suitable for your platform and why.**
  + On an e-commerce platform, performance is key.
  + Binary search is more suitable because of its O(log n) time.
  + In practice, e-commerce platforms use **more advanced algorithms** like search trees or indexes — but between linear and binary — binary search is better.