**Understand Search Algorithms**

**Linear Search:**

* **Definition:** A simple search algorithm that checks each element in the list until the target element is found or the list ends.
* **Time Complexity:** O(n) in the worst case, where n is the number of elements in the list.
* **Best For:** Small or unsorted datasets.

**Binary Search:**

* **Definition:** An efficient search algorithm that repeatedly divides the sorted list in half to find the target element.
* **Time Complexity:** O(log n) in the worst case, where n is the number of elements in the list.
* **Best For:** Large, sorted datasets.

**Analysis**

**Time Complexity Analysis:**

1. **Linear Search:**
   * **Time Complexity:** O(n) in the worst case, where n is the number of books.
   * **Usage:** Suitable for small or unsorted datasets.
2. **Binary Search:**
   * **Time Complexity:** O(log n) in the worst case, where n is the number of books.
   * **Usage:** Suitable for large, sorted datasets. Requires the list to be sorted.

**When to Use Each Algorithm:**

* **Linear Search:** Use when the dataset is small or unsorted. It is simple and does not require the list to be sorted.
* **Binary Search:** Use when the dataset is large and sorted. It is more efficient than linear search but requires the list to be sorted.