**Library Management System**

Understand Search Algorithms

Linear Search

* Description: A simple search algorithm that checks each element in the list sequentially until the target element is found or the end of the list is reached.
* Time Complexity:
  + Best Case: O(1) if the element is at the beginning.
  + Average Case: O(n), where n is the number of elements.
  + Worst Case: O(n), where n is the number of elements.

Binary Search

* Description: A more efficient search algorithm that works on sorted lists. It repeatedly divides the search interval in half. If the value of the search key is less than the item in the middle of the interval, the algorithm narrows the interval to the lower half; otherwise, it narrows it to the upper half.
* Time Complexity:
  + Best Case: O(1) if the middle element is the target.
  + Average Case: O(log n), where n is the number of elements.
  + Worst Case: O(log n), where n is the number of elements.

**Implementation**

1. Linear Search: Searches through the array sequentially.
   * Time Complexity: O(n) for average and worst-case scenarios.
   * Implementation: linearSearchByTitle method iterates through the array to find the book by title.
2. Binary Search: Requires the list to be sorted. It works by dividing the list in half repeatedly.
   * Time Complexity: O(log n) for average and worst-case scenarios.
   * Implementation: binarySearchByTitle method performs binary search on the sorted array.

**Analysis**

* Linear Search:
  + Time Complexity: O(n), making it less efficient for large datasets.
  + Usage: Suitable for small or unsorted lists where sorting is not practical.
* Binary Search:
  + Time Complexity: O(log n), making it much more efficient for large sorted datasets.
  + Usage: Best used when the dataset is large and can be sorted. Sorting has a time complexity of O(n log n), so binary search is more efficient when multiple searches are needed.

When to Use Each Algorithm

* Linear Search: Use when the dataset is small or unsorted, or when sorting the dataset is impractical.
* Binary Search: Use when the dataset is large and can be sorted. It provides significant performance improvements for repeated searches in sorted datasets.