**Exercise 6: Library Management System**

**Linear Search:**

Linear search is a simple searching algorithm that sequentially checks each element in a list until the desired element is found or all elements have been checked. It works well for unsorted data.

* **Algorithm**:
  + Start from the beginning of the list.
  + Compare each element with the target element.
  + If found, return its position (index); otherwise, continue to the next element.
  + If the end of the list is reached without finding the element, return a specified value (e.g., -1).
* **Time Complexity**:
  + **Best case**: O(1) (when the element is found at the first position).
  + **Worst and Average case**: O(n) (when the element is at the last position or not present at all).

**Binary Search:**

Binary search is an efficient algorithm for finding an item from a sorted list of items. It works by repeatedly dividing in half the portion of the list that could contain the item, until it narrows down the possible locations to just one.

* **Algorithm**:
  + Compare the target value with the middle element of the sorted list.
  + If they are equal, return the middle element.
  + If the target value is less than the middle element, narrow the search to the left half.
  + If the target value is greater than the middle element, narrow the search to the right half.
  + Repeat until the element is found or the search space is empty.
* **Time Complexity**:
  + **Best case**: O(1) (when the middle element is the target).
  + **Worst and Average case**: O(log n) (due to halving the search space in each step).

**Time Complexity Comparison**

* **Linear Search**:
  + **Time Complexity**: O(n)
  + **Use Case**: Use linear search when the list is small or unsorted. It's simple to implement and works efficiently with unsorted data.
* **Binary Search**:
  + **Time Complexity**: O(log n)
  + **Use Case**: Use binary search when the list is sorted. It's significantly faster than linear search for large datasets, especially useful when repeated searches are anticipated.

**When to Use Each Algorithm**

* **Linear Search**:
  + Use when the dataset is small or unsorted.
  + Suitable for applications where the list size is expected to remain small or when the dataset changes frequently.
* **Binary Search**:
  + Use when the dataset is sorted.
  + Ideal for scenarios where the list size is large and sorted, as it provides efficient search operations with a time complexity of O(log n).