**EXERCISE 6: Library Management System**

**Understanding Search Algorithms**

**1. Linear Search:**

* **Description:** Linear search sequentially checks each element in the list until the desired element is found or the list is exhausted.
* **Time Complexity:**
  + **Best Case:** O(1) (if the element is the first one checked)
  + **Average Case:** O(n)
  + **Worst Case:** O(n) (if the element is not present or is the last one checked)
* **Space Complexity:** O(1) (does not require additional space beyond the input list)

**2. Binary Search:**

* **Description:** Binary search works on sorted lists. It repeatedly divides the list in half and compares the target value with the middle element, narrowing down the search range based on the comparison.
* **Time Complexity:**
  + **Best Case:** O(1) (if the middle element is the target)
  + **Average Case:** O(log n)
  + **Worst Case:** O(log n) (if the element is not present, it still involves dividing the list in half each time)
* **Space Complexity:** O(1) (iterative implementation) or O(log n) (recursive implementation due to stack space)

**Analysis**

**Time Complexity Comparison:**

* **Linear Search:**
  + **Best Case:** O(1)
  + **Average Case:** O(n)
  + **Worst Case:** O(n)
* **Binary Search:**
  + **Best Case:** O(1)
  + **Average Case:** O(log n)
  + **Worst Case:** O(log n)