**Exercise 6: Library Management System**

Q. Explain linear search and binary search algorithms.

A.**Linear search** is a method for searching for an element in a collection of elements. In linear search, each element of the collection is visited one by one in a sequential fashion to find the desired element. Linear search is also known as **sequential search**.

**Binary search** efficiently finds the target element in a sorted list by repeatedly dividing the search interval in half.

Q. Compare the time complexity of linear and binary search.

A. LINEAR SEARCH:

The**time complexity** of the **Linear Search algorithm** is**O(n)**, where **n** is the number of elements in the array.

**Best Case Time Complexity of Linear Search Algorithm: O(1)**

**Average Case Time Complexity of Linear Search Algorithm: O(n)**

**Worst Case Time Complexity of Linear Search Algorithm: O(n)**

BINARY SEARCH:

**Best Case**: O(1) (target is at the middle position)

**Average Case**: O(log n) (searching through half of the list each time)

**Worst Case**: O(log n) (searching through half of the list each time)

Q. Discuss when to use each algorithm based on the data set size and order.

**Linear Search**: **Use When**:

* + The data set is small, making the difference in performance negligible.
  + The data is unsorted or cannot be sorted easily.
  + Simplicity is preferred and performance is not a critical concern.

**Binary Search**: **Use When**:

* + The data set is large, and efficient search performance is needed.
  + The data is sorted or can be sorted (sorting is done once, and search operations are frequently performed).
  + Faster search performance is required, and the overhead of maintaining a sorted list is acceptable.