Library Management System

1. Linear Search:

* Linear search is a searching algorithm in which we traverse through entire array of inputs to find the target
* It is similar to brute force
* We exit when we find the target

2. Binary search:

* Binary search is a searching algorithm which works when the array is sorted
* The idea is two make the array into two parts where the first half is lesser than mid value and second half is greater than mid value
* We calculate mid value using (left+right)/2 and use a while loop until left<=right
* We check if mid value is target. If yes we break the loop, else we update left and right based on the condition (array[mid]>target)
* In this way we cut down the number of iterations from N to Log N

3. Linear search vs Binary search:

1. Best case:

* Linear search: O(1), this happens when the target element is present at the very first index
* Binary search: O(1), this happens when the length of array is odd and target element is present at exactly middle

1. Average case:

* Linear search: O(n) , this happens when the element is somewhere in the array except the very first index
* Binary search: O(log n), this happens when the element is somewhere in the array except in the middle

1. Worst case

* Linear search: O(n), this happens when the target element is not present in the array
* Binary search: O(log n), this happens when the target element is not present in the array

4. Use Case:

* Linear Search is simple and flexible, ideal for small datasets or unsorted data.
* Binary Search is significantly faster for large datasets but requires sorted data, which may introduce overhead if data changes frequently.