Linear Search:

Linear search, also known as sequential search, is a simple algorithm that checks each element of a list sequentially until the desired element is found or the end of the list is reached.

Start at the beginning of the list. Compare the current element with the target value. If the element is found, return its index. If the end of the list is reached without finding the element, return -1 (or indicate not found)

Time complexity: O(n) in all cases (best, average, and worst)

Binary Search:

Binary search is a much more efficient algorithm for searching sorted lists. It works by repeatedly dividing the search interval in half.

 Ensure the list is sorted.

 Find the middle element of the list.

 Compare the middle element with the target value.

 If the middle element is the target value, return its index.

 If the middle element is greater than the target value, search the left half of the list.

 If the middle element is smaller than the target value, search the right half of the list.

 Repeat steps 2-6 until the target value is found or the search interval is empty.

Time complexity: O (log n) in average and worst cases, O (1) in best case (when the target is the middle element)

**When to Use Which Algorithm**

* **Linear Search:**
  + Unsorted lists
  + Small datasets
  + Simple implementation is a priority
* **Binary Search:**
  + Sorted lists
  + Large datasets
  + Performance is critical