Explain linear search and binary search algorithms.

Linear Search is a method in which we search through a list sequentially for a specified value, and the method does not require the list to be sorted, rather it starts its search from the first element, going through the list completely. The time complexity of a linear search is O(n), where n is the number of elements in the list. In other words, the algorithm may have to traverse the whole list in the worst-case scenario before it can finally find the target or decide that it does not exist. Linear search is a technique that is difficult for small lists or raw data, both of which may not be easy or required to sort the data.

Binary Search is a much quicker method to find a target value in a sorted list. It uses the approach of cutting the list in half repeatedly and comparing the value with the middle-pier of the list to find the specified value. If the target value is lesser than the middle element, the search will be continued in the lower half; if it is greater, the upper half will be searched. This process is repeated until the target value is found or the subarray which is being searched becomes empty. The time complexity of binary search is O(log n), where n is the number of elements in the list. This means that binary search is the fastest way to find the target as it is logarithmically slower as compared to linear search especially if the data set is large. But on the other side, binary search can only be used for lists that are already sorted, and it limits itself to occasions where data is naturally sorted, or it can be sorted efficiently before searching.Compare the time complexity of linear and binary search.

Time Complexity Comparison

* Linear Search: O(n) – Every element is checked until the target is found.
* Binary Search: O(log n) – The search space is halved with each step.

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

Linear Search is best for small or unsorted datasets where sorting is impractical. It checks each item sequentially, making it simple but less efficient for large lists.

Binary Search is ideal for large, sorted datasets due to its efficient O(log n) time complexity. It quickly narrows down the search by repeatedly dividing the dataset in half, but requires the data to be sorted.