

Learn the difference between linear search and binary search

Binary search and linear search are two frequently used search methods for finding an item in a list or array. Both algorithms have benefits and drawbacks, and knowing the [difference between linear search and binary search](#) can help you decide which is best for your particular use case.

Linear Search

The most straightforward search algorithm is linear search, often called sequential search. Once the desired item is located or the full list has been searched, it includes iterating through each element of the list or array. Learning about the definition of Linear search is important before knowing the **difference between linear search and binary search**.

Whenever a match is found or the list's end is reached, the algorithm repeats the process starting at the beginning of the list and comparing each element with the target item. The algorithm returns the item's index if it is located. The algorithm returns a "not found" or "null" value if the item cannot be located. When there are n elements in a list or array, the time - varying complexity of a linear search is $O(n)$. This implies that longer lists or arrays require the algorithm to look through them more slowly.

One benefit of linear search is that it may be applied to any kind of list or array, whether it is sorted or unsorted, this makes a major **difference between linear search and binary search**. Furthermore, because linear search uses so little memory, it is a suitable option for tiny lists or arrays.

Binary Search

On the other hand, binary search, a more advanced search technique, needs the list or array to be sorted in either ascending or descending order. Just like the linear search definition, the binary search definition is also important regarding the **difference between linear search and binary search**. Up until the target item is located or the interval is empty, it works by repeatedly splitting the search interval in half and removing half of the remaining elements at each step.

The middle member of the list or array is the first thing the algorithm compares to when determining the target item. The algorithm returns the index of the middle element if the target item is identical to the middle element. The search interval is reduced to the lower half of the list or array if the target item is less than the middle element. The search interval is shortened to the top half of the list or array if the target item is bigger than the middle element. Until the target object is located or the search period is empty, the process continues. Keep reading to know the **difference between linear search and binary search**.

Differences between Linear Search and Binary Search

Linear search refers to how they look for items in a list or array. Binary search reduces the search interval in half at each step while linear search systematically examines each element in the list or array.

The way the two algorithms handle sorted and unsorted lists or arrays is another **difference between linear search and binary search**. Whether a list or array is sorted or unsorted, linear search can be applied to either type of data structure. On the other hand, ascending or descending order must be used when sorting a list or array for binary search.

The two algorithms also differ in terms of temporal complexity. Because linear search has an $O(n)$ time complexity, it takes longer to search through longer lists or arrays. However, Binary search is faster at traversing longer lists or arrays due to its $O(\log n)$ time complexity.