

1. Where is linear searching used? a) When the list has only a few elements b) When performing a single search in an unordered list c) Used all the time d) When the list has only a few elements and When performing a single search in an unordered list
 Answer: d Explanation: It is practical to implement linear search in the situations mentioned in When the list has only a few elements and When performing a single search in an unordered list, but for larger elements the complexity becomes larger and it makes sense to sort the list and employ binary search or hashing.

3. What is the best case for linear search? a) $O(n \log n)$ b) $O(\log n)$ c) $O(n)$ d) $O(1)$
 Answer: a Explanation: Unordered term refers to the given array, that is, the elements need not be ordered. To search for an element in such an array, we need to loop through the elements until the desired element is found.

4. What is the worst case for linear search? a) $O(n \log n)$ b) $O(\log n)$ c) $O(n)$ d) $O(1)$
 Answer: d Explanation: The element is at the head of the array, hence $O(1)$.

6. What is the best case and worst case complexity of ordered linear search? a) $O(n \log n)$, $O(\log n)$ b) $O(\log n)$, $O(n \log n)$ c) $O(n)$, $O(1)$ d) $O(1)$, $O(n)$
 Answer: c Explanation: Worst case is when the desired element is at the tail of the array or not present at all, in this case you have to traverse till the end of the array, hence the complexity is $O(n)$.

10. Which of the following is a disadvantage of linear search? a) Requires more space b) Greater time complexities compared to other searching algorithms c) Not easy to understand d) Not easy to implement
 Answer: b Explanation: The term ordered refers to the items in the array being sorted (here we assume ascending order). So traverse through the array until the element, if at any time the value at i exceeds key value, it means the element is not present in the array. This provides a slightly better efficiency than unordered linear search.