# Algorithm: Linear/Sequential Search

## Resources & Acknowledgements

The following materials were utilized to create this document:

<https://en.wikipedia.org/wiki/Linear_search>

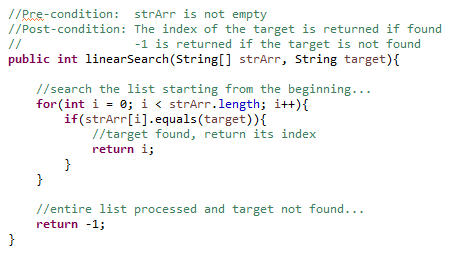
<https://en.wikipedia.org/wiki/Big_O_notation>

## Description

Linear search or sequential search is an algorithm which is utilized for finding a target value in a list. The algorithm sequentially checks each element of a list for a target value until a match is found or all elements have been searched.

Performance will be discussed later, but it is important to note this algorithm belongs in the class of “brute-force” algorithms. It is not an efficient method to search for values, and it is usually reserved for lists that are unordered.

## Java Example



## Performance

Normally, programmers focus on worst-case scenario performance for reliability reasons. However, we can describe an algorithm’s performance for different scenarios – worst-case, best-case, and average-case. We focus on the number of comparisons the algorithm is performing to describe its performance.

### Worst-Case

The target is not in the list, so we examine each item in the list.

***O(n)***

### Best-Case

The target is the first item in the list. We perform one comparison and find the target.

***O(1)***

### Average-Case

If the values are randomly distributed in the list, we should find the target after searching half the values on average. Notice, the coefficient of N is dropped when we convert to Big-O notation.

***N/2 => O(n)***