

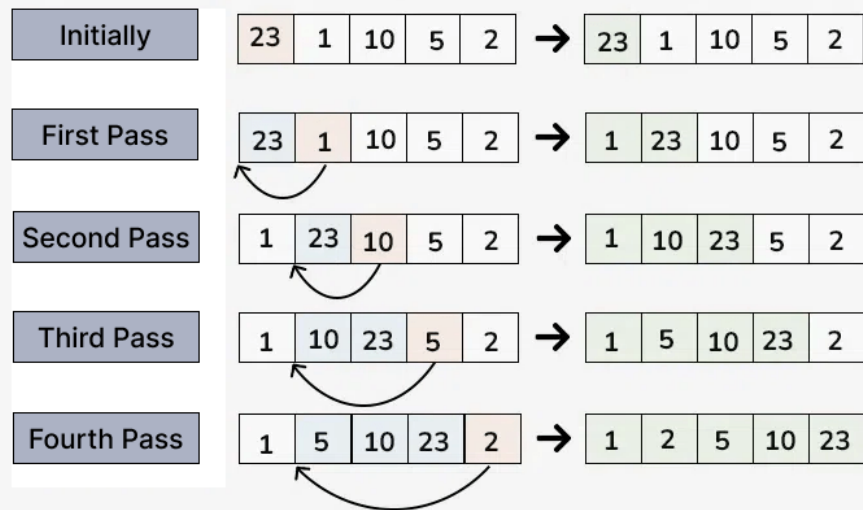
Insertion Sort



Insertion sort is a sorting algorithm that places an unsorted element at its suitable place in each iteration. It uses one part of the array to hold the sorted values, and the other part of the array to hold values that are not sorted yet. The algorithm takes one value at a time from the unsorted part of the array and puts it into the right place in the sorted part of the array, until the array is sorted.

How it works:

1. Take the first value from the unsorted part of the array.
2. Move the value into the correct place in the sorted part of the array.
3. Go through the unsorted part of the array again as many times as there are values.



Insertion Sort



Pseudocode

```

Version 1 : insertionSort(array)
  mark first element as sorted
  for each unsorted element X
    'extract' the element X
    for j <- lastSortedIndex down to 0
      if current element j > X
        move sorted element to the right by 1
    break loop and insert X here
  end insertionSort

```

```

Version 2 : Insertion-Sort(A)
for j = 2 to A.length
  key = A[j]
  i = j - 1
  while i > 0 and A[i] > key
    A[i + 1] = A[i]
    i = i - 1
  A[i + 1] = key

```

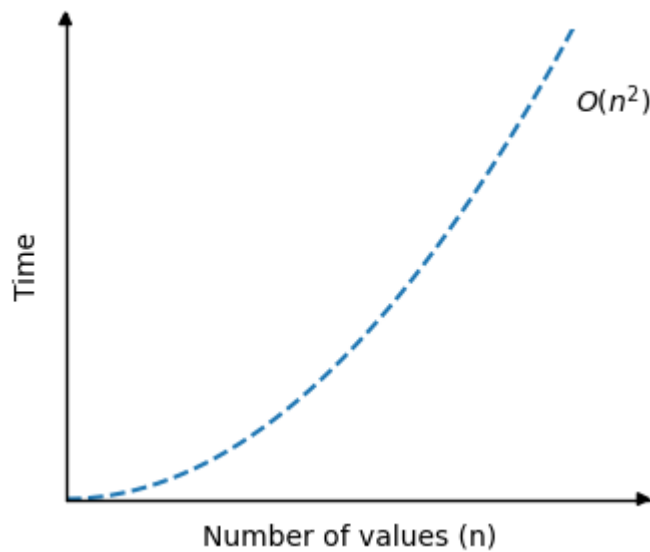
Analysis

Time Complexity	
Best	$O(n)$
Worst	$O(n^2)$
Average	$O(n^2)$
Space Complexity	$O(1)$
Stability	Yes

→ **Worst Case Complexity:** $O(n^2)$ Suppose, an array is in ascending order, and you want to sort it in descending order. In this case, worst case complexity occurs. Each element has to be compared with each of the other elements so, for every n th element, $(n-1)$ number of comparisons are made. Thus, the total number of comparisons = $n*(n-1) \sim n^2$

→ **Best Case Complexity:** $O(n)$ When the array is already sorted, the outer loop runs for n number of times whereas the inner loop does not run at all. So, there are only n number of comparisons. Thus, complexity is linear.

→ **Average Case Complexity:** $O(n^2)$ It occurs when the elements of an array are in jumbled order (neither ascending nor descending).



Advantages of Insertion Sort:

- Simple and easy to implement.
- **Stable** sorting algorithm.
- Efficient for small lists and nearly sorted lists.
- Space-efficient as it is an in-place algorithm.
- Adoptive. the number of inversion is directly proportional to number of swaps. For example, no swapping happens for a sorted array and it takes $O(n)$ time only.

Disadvantages of Insertion Sort:

- Inefficient for large lists.
- Not as efficient as other sorting algorithms (e.g., merge sort, quick sort) for most cases.

Applications of Insertion Sort:

Insertion sort is commonly used in situations where:

- The list is small or nearly sorted.
- Simplicity and stability are important.

- Used as a subroutine in Bucket Sort
- Can be useful when array is already almost sorted




Java implementation can be found under Implementation_Java folder




References

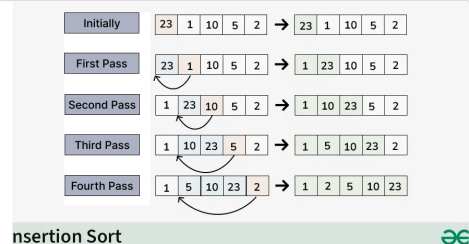
W3Schools.com

W3Schools offers free online tutorials, references and exercises in all the major languages of the web. Covering popular subjects like HTML, CSS, JavaScript, Python, SQL,  https://www.w3schools.com/dsa/dsa_algo_insertionsort.php




Insertion Sort Algorithm - GeeksforGeeks

Insertion sort is a simple, in-place sorting algorithm that iteratively places each element from an unsorted list into its correct position within a sorted portion of the list, making it  <https://www.geeksforgeeks.org/insertion-sort-algorithm/>



Insertion Sort Algorithm

Insertion Sort Algorithm - Insertion sort is a very simple method to sort numbers in an ascending or descending order. This method follows the incremental method. It can be  https://www.tutorialspoint.com/data_structures_algorithms/insertion_sort_algorithm.htm



Insertion Sort (With Code in Python/C++/Java/C)

Insertion Sort is a sorting algorithm that places the input element at its suitable place in each pass. It works in the same way as we sort cards while playing cards game. In this tutorial, you will understand the working of insertion sort with working code in C, C++, Java, and Python.

 <https://www.programiz.com/dsa/insertion-sort>

✍️ Author → Serhat Kumas

<https://www.linkedin.com/in/serhatkumas/>

SerhatKumas - Overview

Computer engineering student who loves coding in different fields instead of focusing on a one spesific area. -

SerhatKumas

 <https://github.com/SerhatKumas>

