

Algorithms arranged in data structures & algorithms

Sorting is to arrange data in a specific format. In computer science, sorting algorithms determine how to arrange data in a certain order. Sorted in order here is arranged in numeric or alphabetical order as in the dictionary.

Sorting is to arrange data in a specific format. In computer science, sorting algorithms determine how to arrange data in a certain order. Sorted in order here is arranged in numeric or alphabetical order as in the dictionary.

The importance of arranging data is: finding data can be optimal if the data is arranged in a certain order (increase or decrease). Arrangements are also used to represent data in a more readable format.

In-place and Not-in-place arrangement algorithms

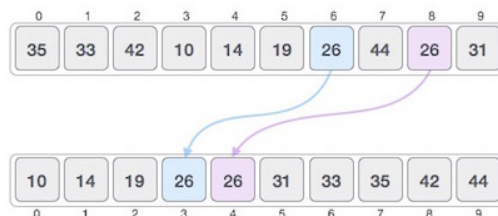
Arrangement algorithms may need to add some extra memory for comparison and temporary memory to store some data elements.

Algorithms that do not require any extra memory and the arrangement is carried out in the previously declared memory section (eg in an array for example) is called in-place sorting. An example of this sort of algorithm is the bubble sorting algorithm (bubble sorting).

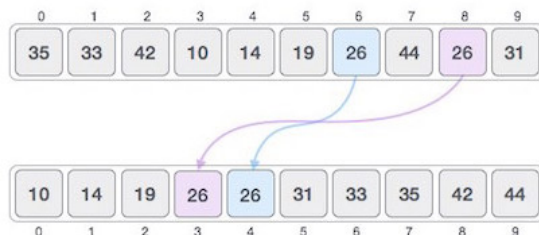
But in some sort algorithms, the program needs to add the amount of memory that can be greater than or equal to the number of elements being sorted. These algorithms are called not-in-place sorting. For example, this sort of algorithm is merge sort.

Algorithm of fixed arrangement and comparison arrangement

A sorting algorithm is called a fixed arrangement if, after conducting the alignment, the relative position between equal elements is not changed.



An algorithm is called comparative sort if in the process of performing the algorithm we proceed to compare the keys and swap elements for each other. That is, the relative position of equal elements is changed.



Adaptive and Non-Adaptive Alignment algorithms

An algorithm is considered as an adaptive, if it takes advantage of the sorted elements in the list that has been sorted. That is, while sorting if the initial list has some sorted elements, the adaptive form algorithm will record these elements and will try not to change their order.

In contrast to the above algorithm, the non-adaptive algorithm does not record previously arranged elements. This type of algorithm will try to rearrange each element in the original list.

Important concepts in sorting algorithms

Below is a brief introduction to some of the concepts that appear while discussing sorting algorithms:

Order increases

A range of values is considered as in increasing order if the following element is larger than the preceding element. Examples: 1, 3, 5, 6, 9.

Order decreased

A range of values is treated as in descending order if the following element is smaller than the preceding element. Examples: 9, 6, 5, 3, 1.

Order does not increase

A range of values is considered as in the order that does not increase if the following element is smaller than or equal to the preceding element. Examples: 9, 6, 5, 5, 1. This type of order appears when in a sequence that contains the same values (in the example is 5).

Order does not decrease

A range of values that are considered as in order does not decrease if the following element is greater than or equal to the preceding element. Examples: 1, 5, 5, 6, 9. This type of order appears when in a sequence containing the same values (in the example is 5).

According to Tutorialspoint

Previous article: Hash Table data structure

Next lesson: Bubble Sort algorithm (Bubble Sort)

You finished reading the article "**Algorithms arranged in data structures & algorithms**" edited by the [TipsMake](#) team. We hope this article has provided you with many useful tech tips and tricks. You can search for similar articles on tips and guides. Thank you for reading and for following us regularly.
