

Radix Sort is a non-comparative sorting algorithm that sorts elements by processing them digit by digit.

It is based on the idea of sorting numbers by their individual digits, from the least significant digit (LSD) to the most significant digit (MSD).

Radix Sort is particularly useful for sorting integers or strings with fixed-length representations.

Here's how the Radix Sort algorithm works:

- Determine the maximum number of digits among all the elements in the input list. This will determine the number of passes required for the sorting process.
- Starting from the least significant digit (LSD), perform the following steps for each digit position:
 - a. Group the elements based on the value of the current digit (0-9) by using counting sort or any stable sorting algorithm.
- Repeat step 2 for each subsequent digit position, moving towards the most significant digit (MSD).
- After processing all the digits, the list will be sorted.

Example:

Consider the following unsorted list of integers: [170, 45, 75, 90, 802, 24, 2, 66].

Step 1: Determine the maximum number of digits

- The maximum number of digits in the list is 3.

Step 2: Sorting by digit position (LSD to MSD)

- Start with the least significant digit (the units place):
 - Group the elements based on the value of the current digit:
 - [170, 90, 802] (digit 0)
 - [2] (digit 2)
 - [24] (digit 4)
 - [45, 75] (digit 5)
 - [66] (digit 6)
- Move to the next digit position (the tens place):
 - Group the elements based on the value of the current digit:
 - [802, 2, 24, 45, 66, 170, 75, 90] (digit 0)
- Move to the next and final digit position (the hundreds place):
 - Group the elements based on the value of the current digit:
 - [2, 24, 45, 66, 75, 90, 170, 802] (digit 0)

The list is now sorted in ascending order: [2, 24, 45, 66, 75, 90, 170, 802].

Radix Sort has a time complexity of $O(d * (n + k))$, where 'd' is the maximum number of digits, 'n' is the number of elements, and 'k' is the range of digits (10 in the case of decimal numbers).

It is a stable sorting algorithm and can handle large lists efficiently.