

L - 17 Radix Sort

Tuesday, May 19, 2020 8:42 AM

Radix Sort

Sort on digits from right to left

Radix-Sort (Arr, d)

1. for $i = 1$ to d
2. use **stable sort** to sort array Arr on digit i

Example:

329	720	720	145
457	355	329	329
657	145	436	355
839	436	839	436
436	457	145	457
720	657	355	657
355	329	457	720
145	839	657	839

Counting-Sort(A, B, k, d=1)

1. let $C[0..k]$ be a dynamically allocated array of length $k+1$
2. //Initialize the C array with zero
3. for $i = 0$ to k
4. $C[i] = 0$
5. //Count and store the number of each element in A
6. for $j = 1$ to A. length
7. $C[A[j]] = C[A[j]] + 1$
8. //Find the number of elements less than or equal to x
9. for $j = 1$ to k
10. $C[j] = C[j] + C[j - 1]$
11. for $j = A. \text{length}$ downto 1
12. $B[C[A[j]]] = A[j]$
13. $C[A[j]] = C[A[j]] - 1$

$$T(n) = O(n) + O(n) + O(k) + O(k) = 2c_1n + 2c_2k = O(n + k)$$

N=10

K=1000

Time Complexity of Radix Sort

Counting Sort Time Complexity $O(n + k)$

$O(d(n + k))$

20 020 0020

321	650	20	020
57	20	321	057
650	350	436	088
339	321	339	143
436	143	143	321
20	436	650	339
350	57	350	350
143	88	57	126

5 5 4 1
1 5 4 5
1 4 5 5

6 5 5 8
5 6 5 8
5 5 6 8

Stable Sort:

A Sorting algorithm is STABLE if the **order of duplicate elements** in the input is preserved in the sorted output.

Quick Sort	non-Stable
Merge Sort	Stable
Insertion Sort	Stable
Selection Sort	non-Stable

20	436	650	339
350	57	350	350
143	88	57	436
88	339	88	650

Merge Sort	Stable
Insertion Sort	Stable
Selection Sort	non-Stable
Bubble Sort	Stable
Counting Sort	Stable

329	329	329	457
457	457	439	329
439	439	457	439

CAT	DOG	CAR	BAT
DOG	DIG	CAT	CAR
CAR	CAR	MAT	CAT
DIG	CAT	BAT	DIG
MAT	MAT	DIG	DOG
BAT	BAT	DOG	MAT