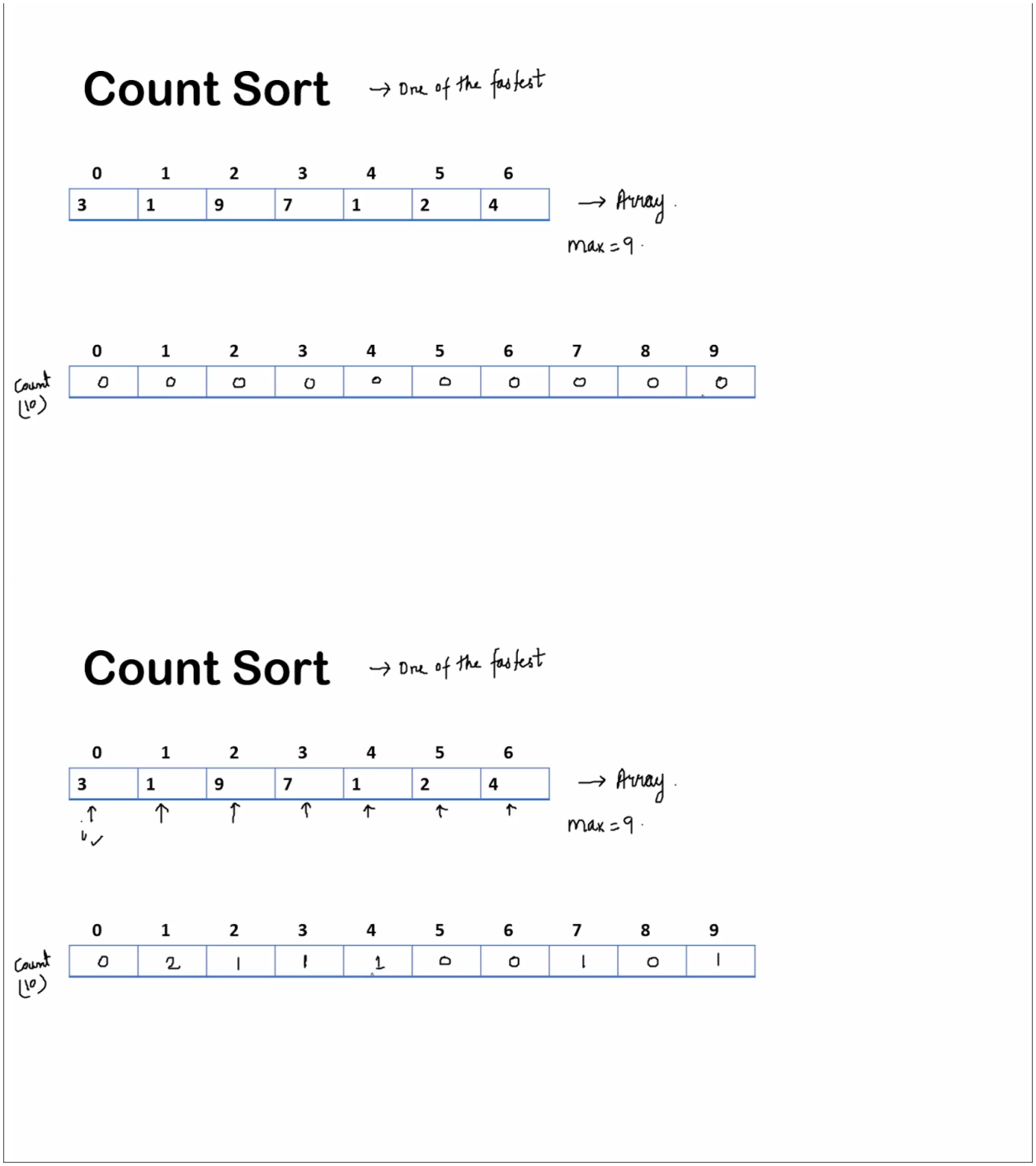
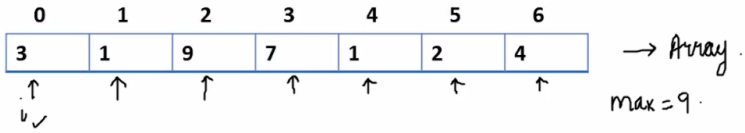
Count sort → One of the fastest methods

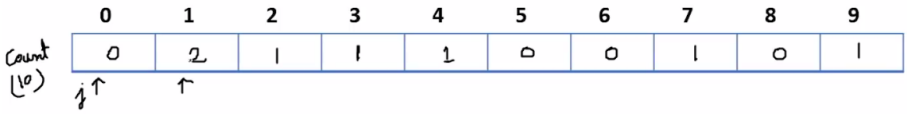


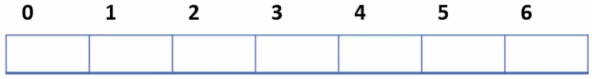
Count Sort → One of the fastest



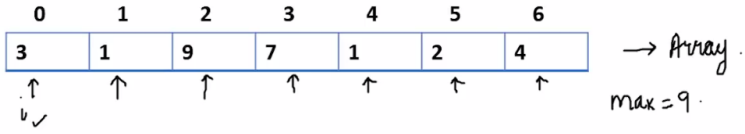
→ Array.

max = 9.



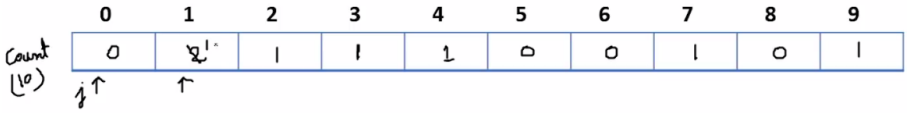


Count Sort → One of the fastest

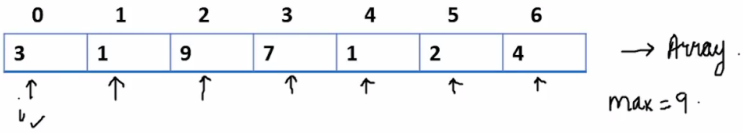


→ Array.

max = 9.

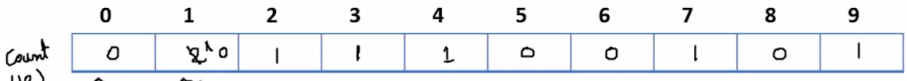


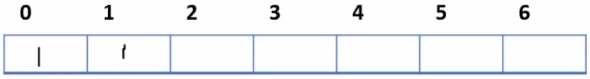
Count Sort → One of the fastest



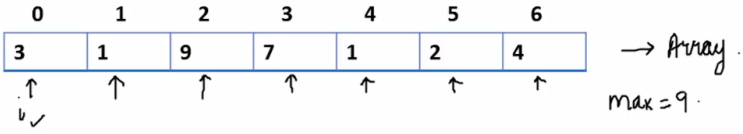
→ Array.

max = 9.



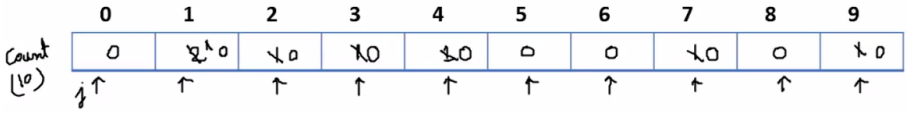


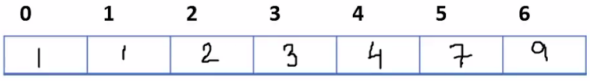
Count Sort → One of the fastest



→ Array.

max = 9.





→ Sorted Array

Count Sort → One of the fastest

Size of Array = n = 6

Given Array

0	1	2	3	4	5	6
3	1	9	7	1	2	4

→ Array.

max = 9 → m

Count (10)

0	1	2	3	4	5	6	7	8	9
0	2	4	4	5	0	0	4	0	4

✓

0	1	2	3	4	5	6
1	1	2	3	4	7	9

→ Sorted Array

Analysis

1. Demerit → It takes extra spaces
2. Time complexity = O(m + n) $\sim O(n)$ → as time complexity is discussed for large n
    a. where m = size of new count array
    b. n = size of given array
    c. if both ‘m’ and ‘n’ are comparable then time complexity is O(m + n)