**GATE** ▼

CS Subjects ▼

GBlog

Quizzes ▼

2

Practice

Puzzles

Radix Sort

Count pairs in array whose sum is divisible by K

Nth number whose sum of digit is multiple of 10

Coin Change | BFS Approach

Given two arrays count all pairs whose sum is an odd number

Find maximum xor of k elements in an array

Maximum absolute difference in an array

Program to find the last two digits of x^y

Partition an array of nonnegative integers into two subsets such that average of both the subsets is equal

Count substrings that contain all vowels | SET 2

Count rotations of N which are Odd and Even

New Algorithm to Generate Prime Numbers from 1 to Nth Number

Sum of the series 1<sup>1</sup> + 2<sup>2</sup> + 3<sup>3</sup> + ..... + n<sup>n</sup> using recursion

Find the sum of digits of a number at even and odd places

Count of integers of length N and value less than K such that they contain digits only from the given set

Check for balanced parentheses in an expression | O(1) space | O(N^2) time complexity

Complexity Analysis of Binary Search

Find the minimum number of operations required to make all array elements equal

Create new linked list from two given linked list with greater element at each node

Count number of ways to reach a given score in a Matrix

# **Counting Sort**

Languages ▼

DS ▼

Algo ▼

Counting sort is a sorting technique based on keys between a specific range. It works by counting the number of objects having distinct key values (kind of hashing). Then doing some arithmetic to calculate the position of each object in the output sequence.

Students ▼

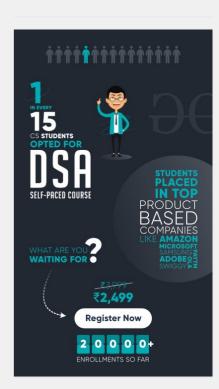
Let us understand it with the help of an example.

```
For simplicity, consider the data in the range 0 to 9.
Input data: 1, 4, 1, 2, 7, 5, 2
 1) Take a count array to store the count of each unique object.
            0 1 2 3 4 5 6 7 8 9
            0 2 2 0 1 1 0 1 0 0
 2) Modify the count array such that each element at each index
 stores the sum of previous counts.
            0 1 2 3 4 5 6 7 8 9
 Count:
            0 2 4 4 5 6 6 7 7
The modified count array indicates the position of each object in
the output sequence.
 3) Output each object from the input sequence followed by
 decreasing its count by 1.
 Process the input data: 1, 4, 1, 2, 7, 5, 2. Position of 1 is 2.
 Put data 1 at index 2 in output. Decrease count by 1 to place
 next data 1 at an index 1 smaller than this index.
```

Recommended: Please solve it on " $\underline{\textit{PRACTICE}}$ " first, before moving on to the solution.

Following is implementation of counting sort.

```
C++
               Java
                         Python
                                      C#
                                             PHP
// Java implementation of Counting Sort
class CountingSort
    void sort(char arr[])
        int n = arr.length;
        // The output character array that will have sorted arr
        char output[] = new char[n];
        // Create a count array to store count of inidividul
        // characters and initialize count array as 0
        int count[] = new int[256];
        for (int i=0; i<256; ++i)</pre>
            count[i] = 0;
        // store count of each character
        for (int i=0; i<n; ++i)</pre>
            ++count[arr[i]];
        // Change count[i] so that count[i] now contains actual
        // position of this character in output array
        for (int i=1; i<=255; ++i)</pre>
            count[i] += count[i-1];
        // Build the output character array
        // To make it stable we are operating in reverse order.
        for (int i = n-1; i>=0; i--)
            output[count[arr[i]]-1] = arr[i];
            --count[arr[i]];
```



# Most popular in Mathematical

Count of integers in a range which have even number of odd digits and odd number of even digits

Count of integers that divide all the elements of the given array

Print all the permutation of length L using the elements of an array | Iterative

Program to find the Nth Prime Number

Count number of ways to get Odd Sum

XOR of all the elements in the given range [L, R]

Find the minimum number of elements that should be removed to make an array good

Find the smallest positive number missing from an unsorted array | Set 3

Find sub-arrays from given two arrays such that they have equal sum

Count of all possible values of X such that A % X = B

Find a triplet in an array whose sum is closest to a given number

Card Shuffle Problem | TCS Digital Advanced Coding Ouestion

Append a digit in the end to make the number equal to the length of the remaining string

Find XOR of numbers from the range [L, R]

Print path from root to all nodes in a Complete Binary Tree

```
Classroom program in NOIDA
Starting from 21st September 2019
  Machine Learning
             OUNDATIO
WITH PYTHON
                    ₹10,999
The course was excellent. All basic topics of ML are covered.
```

```
// Copy the output array to arr, so that arr now
        // contains sorted characters
        for (int i = 0; i<n; ++i)</pre>
             arr[i] = output[i];
    }
    // Driver method
    public static void main(String args[])
        CountingSort ob = new CountingSort();
        char arr[] = {'g', 'e', 'e', 'k', 's', 'f', 'o', 'r', 'g', 'e', 'e', 'k', 's'
                      };
        ob.sort(arr);
        System.out.print("Sorted character array is ");
        for (int i=0; i<arr.length; ++i)</pre>
             System.out.print(arr[i]);
    }
/*This code is contributed by Rajat Mishra */
```

# Output:

```
Sorted character array is eeeefggkkorss
```

Time Complexity: O(n+k) where n is the number of elements in input array and k is the range of input.

Auxiliary Space: O(n+k)

The problem with the previous counting sort was that we could not sort the elements if we have negative numbers in it. Because there are no negative array indices. So what we do is, we find the minimum element and we will store count of that minimum element at zero index

Java





```
// Counting sort which takes negative numbers as well
import java.util.*;
class GFG
    static void countSort(int[] arr)
        int max = Arrays.stream(arr).max().getAsInt();
        int min = Arrays.stream(arr).min().getAsInt();
        int range = max - min + 1;
        int count[] = new int[range];
        int output[] = new int[arr.length];
        for (int i = 0; i < arr.length; i++)</pre>
        {
            count[arr[i] - min]++;
        }
        for (int i = 1; i < count.length; i++)</pre>
        {
            count[i] += count[i - 1];
        }
        for (int i = arr.length - 1; i >= 0; i--)
            output[count[arr[i] - min] - 1] = arr[i];
            count[arr[i] - min]--;
        }
        for (int i = 0; i < arr.length; i++)</pre>
            arr[i] = output[i];
        }
    }
    static void printArray(int[] arr)
        for (int i = 0; i < arr.length; i++)</pre>
            System.out.print(arr[i] + " ");
        System.out.println("");
    }
```

# Most visited in Sorting

Comparison among Bubble Sort, Selection Sort and Insertion Sort

Amazon Internship Interview Experience

Count the triplets such that A[i] < B[j] < C[k]

Greatest contiguous sub-array of size K

Sort an array without changing position of negative numbers



```
// Driver code
public static void main(String[] args)
{
    int[] arr = {-5, -10, 0, -3, 8, 5, -1, 10};
    countSort(arr);
    printArray(arr);
}

// This code is contributed by princiRaj1992
```

# Output:

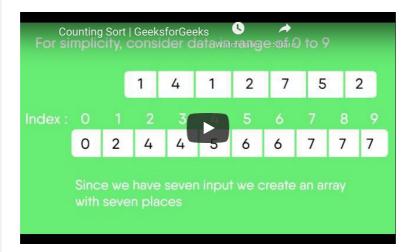
```
-10 -5 -3 -1 0 5 8 10
```

### Points to be noted:

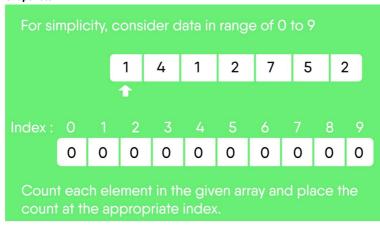
- 1. Counting sort is efficient if the range of input data is not significantly greater than the number of objects to be sorted. Consider the situation where the input sequence is between range 1 to 10K and the data is 10, 5, 10K, 5K.
- $\textbf{2. It is not a comparison based sorting. It running time complexity is <math>O(n)$  with space proportional to the range of data.}
- 3. It is often used as a sub-routine to another sorting algorithm like radix sort.
- **4.** Counting sort uses a partial hashing to count the occurrence of the data object in O(1).
- 5. Counting sort can be extended to work for negative inputs also.

# Exercise:

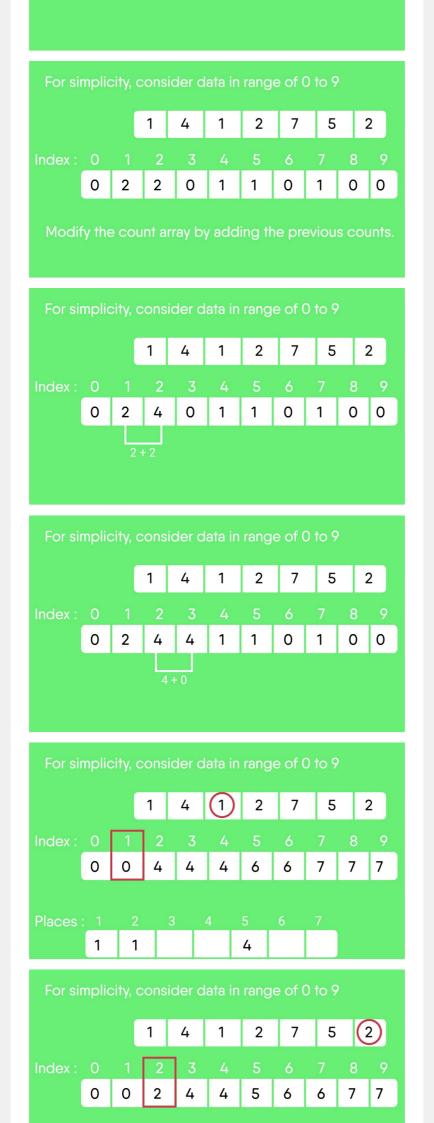
- 1. Modify above code to sort the input data in the range from M to N.
- 2. Is counting sort stable and online?
- 3. Thoughts on parallelizing the counting sort algorithm.

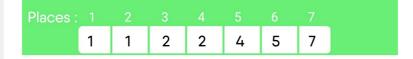


# Snapshots:









# **Quiz on Counting Sort**

# **Coding Practice for Sorting**

Other Sorting Algorithms on GeeksforGeeks/GeeksQuiz

Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Heap Sort, QuickSort, Radix Sort, Counting Sort, Bucket Sort, ShellSort, Comb Sort, PegionHole Sorting

This article is compiled by Aashish Barnwal. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

# **Recommended Posts:**

C Program for Counting Sort

Java Program for Counting Sort

Median and Mode using Counting Sort

Sort an array of 0s, 1s and 2s (Simple Counting)

Comparison among Bubble Sort, Selection Sort and Insertion Sort

Why Quick Sort preferred for Arrays and Merge Sort for Linked Lists?

Counting k-mers via Suffix Array

Counting cross lines in an array

Counting numbers of n digits that are monotone

Bucket Sort To Sort an Array with Negative Numbers

Program to sort an array of strings using Selection Sort

Insertion sort to sort even and odd positioned elements in different orders

Rencontres Number (Counting partial derangements)

Counting numbers whose difference from reverse is a product of  ${\bf k}$ 

Counting even decimal value substrings in a binary string

Improved By: Mithun Kumar, spattk, krikti, sagarudasi2, princiraj1992, more



Previous

Next

I< HeapSort

Comparator function of qsort() in C >I

Writing code in comment? Please use ide.geeksforgeeks.org, generate link and share the link here.

Load Comments

5th Floor, A-118, Sector-136, Noida, Uttar Pradesh - 201305 feedback@geeksforgeeks.org Careers Privacy Policy Contact Us Data Structures Languages CS Subjects Video Tutorials Company-wise Topic-wise How to begin? Write Interview Experience Internships Videos

