

Write JAVA programs to solve the following problems and analyse the time and space complexity. Is your algorithm/approach the optimum?

1. Given an array `arr[]` of positive integers of size `N`. Reverse every sub-array group of size `K`.
Note: If at any instance, there are no more subarrays of size greater than or equal to `K`, then reverse the last subarray (irrespective of its size). You shouldn't return any array, modify the given array in-place.
2. Given two arrays `A` and `B` of equal size `N`, the task is to find if given arrays are equal or not. Two arrays are said to be equal if both of them contain same set of elements, arrangements (or permutation) of elements may be different though.
Note : If there are repetitions, then counts of repeated elements must also be same for two array to be equal.
3. Given an array of `N` integers. Find the first element that occurs at least `K` number of times.
4. Given an array `arr` of distinct elements of size `N`, the task is to rearrange the elements of the array in a zig-zag fashion so that the converted array should be in the below form:
 $arr[0] < arr[1] > arr[2] < arr[3] > arr[4] < \dots < arr[n-2] < arr[n-1] > arr[n]$.
NOTE: If your transformation is correct, the output will be 1 else the output will be 0.
5. Given an unsorted array `A` of size `N` that contains only non-negative integers, find a continuous sub-array that adds to a given number `S` and return the left and right index(1-based indexing) of that subarray.
In case of multiple subarrays, return the subarray indexes which come first on moving from left to right.
Note:- You have to return an ArrayList consisting of two elements left and right. In case no such subarray exists return an array consisting of element -1.
6. Given an array of size `N-1` such that it only contains distinct integers in the range of 1 to `N`. Find the missing element.
7. Given an array of size `N` containing only 0s, 1s, and 2s; sort the array in ascending order.
8. Given an array `A` of positive integers. Your task is to find the leaders in the array. An element of array is leader if it is greater than or equal to all the elements to its right side. The rightmost element is always a leader.
9. Given an array `A` of `n` positive numbers. The task is to find the first Equilibrium Point in an array.
Equilibrium Point in an array is a position such that the sum of elements before it is equal to the sum of elements after it.
Note: Return the index of Equilibrium point. (1-based index)
10. The cost of stock on each day is given in an array `A[]` of size `N`. Find all the segments of days on which you buy and sell the stock so that in between those days your profit is maximum.
Note: Since there can be multiple solutions, the driver code will print 1 if your answer is correct, otherwise, it will return 0. In case there's no profit the driver code will print the string "No Profit" for a correct solution.

11. Given an array having both positive and negative integers. The task is to compute the length of the largest subarray with sum 0.
12. Given two arrays: $a1[0..n-1]$ of size n and $a2[0..m-1]$ of size m . Task is to check whether $a2[]$ is a subset of $a1[]$ or not. Both the arrays can be sorted or unsorted.
13. Given an array of N strings, find the longest common prefix among all strings present in the array.
14. Given two unsorted arrays A of size N and B of size M of distinct elements, the task is to find all pairs from both arrays whose sum is equal to X .
Note: All pairs should be printed in increasing order of u . For eg. for two pairs $(u1, v1)$ and $(u2, v2)$, if $u1 < u2$ then $(u1, v1)$ should be printed first else second.
15. You are given two arrays, A and B , of equal size N . The task is to find the minimum value of $A[0] * B[0] + A[1] * B[1] + \dots + A[N-1] * B[N-1]$, where shuffling of elements of arrays A and B is allowed.
16. Given an array $Arr[]$ of N integers. Find the contiguous sub-array (containing at least one number) which has the maximum sum and return its sum.
17. Given an array of N integers $arr[]$ where each element represents the maximum length of the jump that can be made forward from that element. This means if $arr[i] = x$, then we can jump any distance y such that $y \leq x$.
Find the minimum number of jumps to reach the end of the array (starting from the first element). If an element is 0, then you cannot move through that element.
Note: Return -1 if you can't reach the end of the array.
18. Given an array $arr[]$ and an integer K where K is smaller than size of array, the task is to find the K th smallest element in the given array. It is given that all array elements are distinct.
Note :- l and r denotes the starting and ending index of the array.
19. Given an array $arr[]$ denoting heights of N towers and a positive integer K . For each tower, you must perform exactly one of the following operations exactly once.
Increase the height of the tower by K
Decrease the height of the tower by K
Find out the minimum possible difference between the height of the shortest and tallest towers after you have modified each tower.
Note: It is compulsory to increase or decrease the height by K for each tower. After the operation, the resultant array should not contain any negative integers.