



Dash



All



Articles



Videos



Problems



Quiz

<< Prev

Next >>

Sort an Array with two types of elements

Given an array of 0s and 1s in random order. Segregate 0s on left side and 1s on right side of the array. Traverse array only once.

Examples:

Input : arr[] = [0, 1, 0, 1, 0, 0, 1, 1, 1, 0]

Output : arr[] = [0, 0, 0, 0, 0, 1, 1, 1, 1, 1]

Input : arr[] = [1, 1, 1, 0, 1, 0, 0, 1, 1, 1]

Output : arr[] = [0, 0, 0, 1, 1, 1, 1, 1, 1, 1]

Step 1 : Here we can take two pointers type0 (for element 0) starting from beginning (index = 0) and type1 (for element 1) starting from end index.

Step 2: We intend to put 1 to the right side of the array. Once we have done this then 0 will definitely towards left side of array to achieve this we do following.

We compare elements at index type0

1) if this is 1 then this should be moved to right side so we need to swap this with index type1 once swapped we are sure that element at index type1 is '1' so we need to decrement index type1

2) else it will be 0 then we need to simple increment index type0

C++14

Java



Dash



All



Articles



Videos



Problems



Quiz

30% Off
Courses

Tutorials

Jobs

Practice

Contests



```
// Java program to sort an array with two types
// of values in one traversal.
public class GFG {
    /* Method for segregation 0 and 1
    given input array */
    class segregation {
        static void segregate0and1(int arr[], int n)
        {
            int type0 = 0;
            int type1 = n - 1;

            while (type0 < type1) {
                if (arr[type0] == 1) {

                    // swap type0 and type1
                    arr[type0] = arr[type0] + arr[type1];
                    arr[type1] = arr[type0] - arr[type1];
                    arr[type0] = arr[type0] - arr[type1];
                    type1--;
                }
                else {
                    type0++;
                }
            }
        }
    }

    // Driver program
```



Dash



All



Articles



Videos



Problems



Quiz



```
int arr[]
    = { 1, 1, 1, 0, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0 };

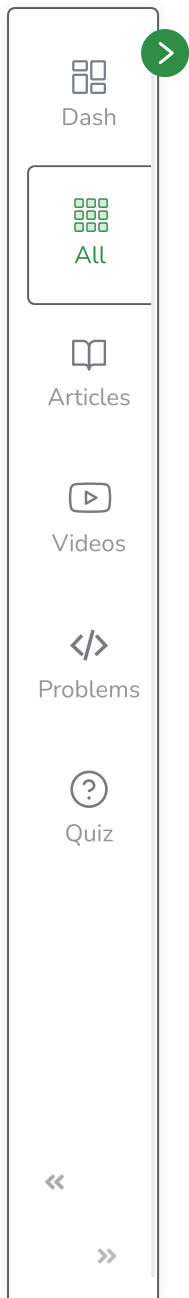
segregate0and1(arr, arr.length);
for (int a : arr)
    System.out.print(a + " ");
}
```

Output:

0 0 0 1 1 1 1 1 1 1 1 1

Time Complexity : $O(n)$ **Auxiliary Space:** $O(1)$

Marked as Read

 Report An Issue

If you are facing any issue on this page. Please let us know.

