



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

# Equilibrium index of an array

Difficulty Level : Easy • Last Updated : 27 May, 2021

Equilibrium index of an array is an index such that the sum of elements at lower indexes is equal to the sum of elements at higher indexes. For example, in an array A:

**Example :**

**Input:**  $A[] = \{-7, 1, 5, 2, -4, 3, 0\}$

**Output:** 3

3 is an equilibrium index, because:

$$A[0] + A[1] + A[2] = A[4] + A[5] + A[6]$$

**Input:**  $A[] = \{1, 2, 3\}$

**Output:** -1

Write a function `int equilibrium(int[] arr, int n)`; that given a sequence `arr[]` of size `n`, returns an equilibrium index (if any) or -1 if no equilibrium indexes exist.

Recommended: Please solve it on "**PRACTICE**" first, before moving on to the solution.



## Method 1 (Simple but inefficient)

Use two loops. Outer loop iterates through all the element and inner loop finds out





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

## C++

```
// C++ program to find equilibrium
// index of an array
#include <bits/stdc++.h>
using namespace std;

int equilibrium(int arr[], int n)
{
    int i, j;
    int leftsum, rightsum;

    /* Check for indexes one by one until
    an equilibrium index is found */
    for (i = 0; i < n; ++i)
    {
        /* get left sum */
        leftsum = 0;
        for (j = 0; j < i; j++)
            leftsum += arr[j];

        /* get right sum */
        rightsum = 0;
        for (j = i + 1; j < n; j++)
            rightsum += arr[j];

        /* if leftsum and rightsum
        are same, then we are done */
        if (leftsum == rightsum)
            return i;
    }

    /* return -1 if no equilibrium
    index is found */
    return -1;
}

// Driver code
int main()
{
    int arr[] = { -7, 1, 5, 2, -4, 3, 0 };
    int arr_size = sizeof(arr) / sizeof(arr[0]);
    cout << equilibrium(arr, arr_size);
}
```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSPELLADS

## C

```
// C program to find equilibrium
// index of an array

#include <stdio.h>

int equilibrium(int arr[], int n)
{
    int i, j;
    int leftsum, rightsum;

    /* Check for indexes one by one until
       an equilibrium index is found */
    for (i = 0; i < n; ++i) {

        /* get left sum */
        leftsum = 0;
        for (j = 0; j < i; j++)
            leftsum += arr[j];

        /* get right sum */
        rightsum = 0;
        for (j = i + 1; j < n; j++)
            rightsum += arr[j];

        /* if leftsum and rightsum are same,
           then we are done */
        if (leftsum == rightsum)
            return i;
    }

    /* return -1 if no equilibrium index is found */
    return -1;
}

// Driver code
int main()
{
    int arr[] = { -7, 1, 5, 2, -4, 3, 0 };
    int arr_size = sizeof(arr) / sizeof(arr[0]);
    printf("%d", equilibrium(arr, arr_size));

    getchar();
    return 0;
}
```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSPELLADS

```
// Java program to find equilibrium
// index of an array

class EquilibriumIndex {
    int equilibrium(int arr[], int n)
    {
        int i, j;
        int leftsum, rightsum;

        /* Check for indexes one by one until
           an equilibrium index is found */
        for (i = 0; i < n; ++i) {

            /* get left sum */
            leftsum = 0;
            for (j = 0; j < i; j++)
                leftsum += arr[j];

            /* get right sum */
            rightsum = 0;
            for (j = i + 1; j < n; j++)
                rightsum += arr[j];

            /* if leftsum and rightsum are same,
               then we are done */
            if (leftsum == rightsum)
                return i;
        }

        /* return -1 if no equilibrium index is found */
        return -1;
    }
}

// Driver code
public static void main(String[] args)
{
    EquilibriumIndex equi = new EquilibriumIndex();
    int arr[] = { -7, 1, 5, 2, -4, 3, 0 };
    int arr_size = arr.length;
    System.out.println(equi.equilibrium(arr, arr_size));
}
```

// This code has been contributed by Mayank Jaiswal



Python3





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```
def equilibrium(arr):
    leftsum = 0
    rightsum = 0
    n = len(arr)

    # Check for indexes one by one
    # until an equilibrium index is found
    for i in range(n):
        leftsum = 0
        rightsum = 0

        # get left sum
        for j in range(i):
            leftsum += arr[j]

        # get right sum
        for j in range(i + 1, n):
            rightsum += arr[j]

        # if leftsum and rightsum are same,
        # then we are done
        if leftsum == rightsum:
            return i

    # return -1 if no equilibrium index is found
    return -1

# driver code
arr = [-7, 1, 5, 2, -4, 3, 0]
print (equilibrium(arr))

# This code is contributed by Abhishek Sharama
```

## C#

```
// C# program to find equilibrium
// index of an array

using System;

class GFG {
    static int equilibrium(int[] arr, int n)
    {
        int i, j;
        int leftsum, rightsum;
```



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```
// initialize left sum
// for current index i
leftsum = 0;

// initialize right sum
// for current index i
rightsum = 0;

/* get left sum */
for (j = 0; j < i; j++)
    leftsum += arr[j];

/* get right sum */
for (j = i + 1; j < n; j++)
    rightsum += arr[j];

/* if leftsum and rightsum are
   same, then we are done */
if (leftsum == rightsum)
    return i;
}

/* return -1 if no equilibrium
   index is found */
return -1;
}

// driver code
public static void Main()
{
    int[] arr = { -7, 1, 5, 2, -4, 3, 0 };
    int arr_size = arr.Length;

    Console.WriteLine(equilibrium(arr, arr_size));
}

// This code is contributed by Sam007
```

## PHP



```
// PHP program to find equilibrium
// index of an array
```



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSPELLADS

```
$rightsum;
```

```
/* Check for indexes one by one until
an equilibrium index is found */
```

```
for ($i = 0; $i < $n; ++$i)
{
    /* get left sum */
    $leftsum = 0;
    for ($j = 0; $j < $i; $j++)
        $leftsum += $arr[$j];

    /* get right sum */
    $rightsum = 0;
    for ($j = $i + 1; $j < $n; $j++)
        $rightsum += $arr[$j];

    /* if leftsum and rightsum
    are same, then we are done */
    if ($leftsum == $rightsum)
        return $i;
}
```

```
/* return -1 if no equilibrium
index is found */
```

```
return -1;
```

```
}
```

```
// Driver code
```

```
$arr = array( -7, 1, 5, 2, -4, 3, 0 );
```

```
$arr_size = sizeof($arr);
```

```
echo equilibrium($arr, $arr_size);
```

```
// This code is contributed
```

```
// by akt_mit
```

```
?>
```

## Javascript

```
<script>
```

```
// JavaScript Program to find equilibrium
```

```
// index of an array
```

```
function equilibrium(arr, n)
```

```
{
```

```
    var i, j;
```

```
    var leftsum, rightsum;
```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```
{  
  
    /*get left sum*/  
    leftsum = 0;  
    for(let j = 0; j < i; j++)  
        leftsum += arr[j];  
  
    /*get right sum*/  
    rightsum = 0;  
    for(let j = i + 1; j < n; j++)  
        rightsum += arr[j];  
  
    /*if leftsum and rightsum are same,  
    then we are done*/  
    if(leftsum == rightsum)  
        return i;  
}  
  
/* return -1 if no equilibrium index is found*/  
return -1;  
}  
  
// Driver code  
  
var arr = new Array(-7,1,5,2,-4,3,0);  
n = arr.length;  
document.write(equilibrium(arr,n));  
  
// This code is contributed by simranarora5sos  
</script>
```

## Output

3

**Time Complexity:**  $O(n^2)$

## Method 2 (Tricky and Efficient)

The idea is to get the total sum of the array first. Then iterate through the array and keep updating the left sum which is initialized as zero. In the loop, we can get the right sum by subtracting the elements one by one. Thanks to Sambasiva for suggesting this solution and providing code for this.







Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

3) Iterate through the array and for each index *i*, do following.

a) Update *sum* to get the right sum.

```
sum = sum - arr[i]
```

```
// sum is now right sum
```

b) If *leftsum* is equal to *sum*, then return current index.

```
// update leftsum for next iteration.
```

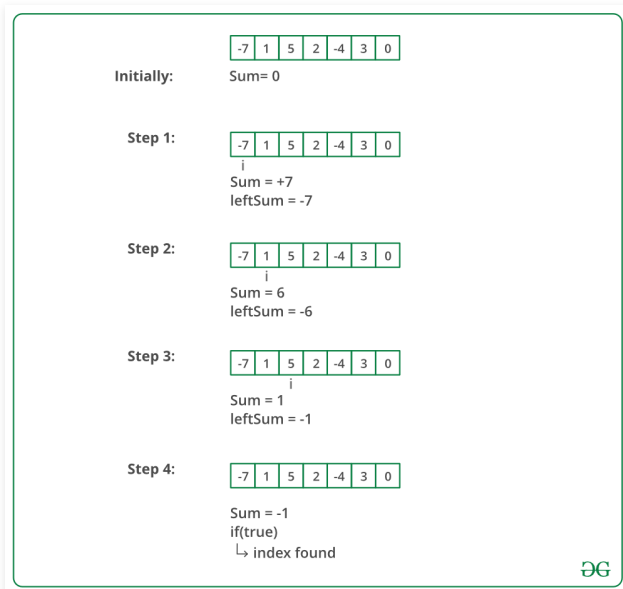
c) *leftsum* = *leftsum* + *arr[i]*

4) return -1

```
// If we come out of loop without returning then
```

```
// there is no equilibrium index
```

The image below shows the dry run of the above approach:



Below is the implementation of the above approach:

## C++

```
// C++ program to find equilibrium
// index of an array
#include <bits/stdc++.h>
using namespace std;

int equilibrium(int arr[], int n)
{
    int sum = 0; // initialize sum of whole array
    int leftsum = 0; // initialize leftsum
```



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```

for (int i = 0; i < n; ++i)
{
    sum -= arr[i]; // sum is now right sum for index i

    if (leftsum == sum)
        return i;

    leftsum += arr[i];
}

/* If no equilibrium index found, then return 0 */
return -1;
}

// Driver code
int main()
{
    int arr[] = { -7, 1, 5, 2, -4, 3, 0 };
    int arr_size = sizeof(arr) / sizeof(arr[0]);
    cout << "First equilibrium index is " << equilibrium(arr, arr_size);
    return 0;
}

// This is code is contributed by rathbhupendra

```

## C

```

// C program to find equilibrium
// index of an array

#include <stdio.h>

int equilibrium(int arr[], int n)
{
    int sum = 0; // initialize sum of whole array
    int leftsum = 0; // initialize leftsum

    /* Find sum of the whole array */
    for (int i = 0; i < n; ++i)
        sum += arr[i];

    for (int i = 0; i < n; ++i) {
        sum -= arr[i]; // sum is now right sum for index i

        if (leftsum == sum)

```



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSPELLADS

```

    /* If no equilibrium index found, then return 0 */
    return -1;
}

// Driver code
int main()
{
    int arr[] = { -7, 1, 5, 2, -4, 3, 0 };
    int arr_size = sizeof(arr) / sizeof(arr[0]);
    printf("First equilibrium index is %d",
           equilibrium(arr, arr_size));

    getchar();
    return 0;
}

```

## Java

```

// Java program to find equilibrium
// index of an array

class EquilibriumIndex {
    int equilibrium(int arr[], int n)
    {
        int sum = 0; // initialize sum of whole array
        int leftsum = 0; // initialize leftsum

        /* Find sum of the whole array */
        for (int i = 0; i < n; ++i)
            sum += arr[i];

        for (int i = 0; i < n; ++i) {
            sum -= arr[i]; // sum is now right sum for index i

            if (leftsum == sum)
                return i;

            leftsum += arr[i];
        }

        /* If no equilibrium index found, then return 0 */
        return -1;
    }
}

// Driver code

```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```
int arr_size = arr.length;
System.out.println("First equilibrium index is " +
    equi.equilibrium(arr, arr_size));
}
}

// This code has been contributed by Mayank Jaiswal
```

## Python3

```
# Python program to find the equilibrium
# index of an array

# function to find the equilibrium index
def equilibrium(arr):

    # finding the sum of whole array
    total_sum = sum(arr)
    leftsum = 0
    for i, num in enumerate(arr):

        # total_sum is now right sum
        # for index i
        total_sum -= num

        if leftsum == total_sum:
            return i
        leftsum += num

    # If no equilibrium index found,
    # then return -1
    return -1

# Driver code
arr = [-7, 1, 5, 2, -4, 3, 0]
print ('First equilibrium index is ',
    equilibrium(arr))

# This code is contributed by Abhishek Sharma
```



```
// C# program to find the equilibrium
```



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSPELLADS

```
class GFG {
    static int equilibrium(int[] arr, int n)
    {
        // initialize sum of whole array
        int sum = 0;

        // initialize leftsum
        int leftsum = 0;

        /* Find sum of the whole array */
        for (int i = 0; i < n; ++i)
            sum += arr[i];

        for (int i = 0; i < n; ++i) {

            // sum is now right sum
            // for index i
            sum -= arr[i];

            if (leftsum == sum)
                return i;

            leftsum += arr[i];
        }

        /* If no equilibrium index found,
        then return 0 */
        return -1;
    }

    // Driver code
    public static void Main()
    {
        int[] arr = { -7, 1, 5, 2, -4, 3, 0 };
        int arr_size = arr.Length;

        Console.WriteLine("First equilibrium index is " +
            equilibrium(arr, arr_size));
    }
}
// This code is contributed by Sam007
```



```
<?php
// PHP program to find equilibrium
```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSPELLADS

```
$sum = 0; // initialize sum of
        // whole array
$leftsum = 0; // initialize leftsum

/* Find sum of the whole array */
for ($i = 0; $i < $n; ++$i)
    $sum += $arr[$i];

for ($i = 0; $i < $n; ++$i)
{
    // sum is now right sum
    // for index i
    $sum -= $arr[$i];

    if ($leftsum == $sum)
        return $i;

    $leftsum += $arr[$i];
}

/* If no equilibrium index
found, then return 0 */
return -1;
}

// Driver code
$arr = array( -7, 1, 5, 2, -4, 3, 0 );
$arr_size = sizeof($arr);
echo "First equilibrium index is ",
    equilibrium($arr, $arr_size);

// This code is contributed by ajit
?>
```

## JavaScript

```
<script>
// program to find equilibrium
// index of an array
function equilibrium(arr, n)
{
    sum = 0; // initialize sum of whole array
    leftsum = 0; // initialize leftsum

    /* Find sum of the whole array */
    for (let i = 0; i < n; ++i)
```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```
sum -= arr[i]; // sum is now right sum for index i

    if (leftsum == sum)
        return i;

    leftsum += arr[i];
}

/* If no equilibrium index found, then return 0 */
return -1;
}

// Driver code

arr =new Array(-7, 1, 5, 2, -4, 3, 0);
n=arr.length;
document.write("First equilibrium index is " + equilibrium(arr, n));

// This code is contributed by simranarora5sos
</script>
```

## Output

First equilibrium index is 3

### Output:

First equilibrium index is 3

**Time Complexity:**  $O(n)$

### Method 3 :

This is a quite simple and straightforward method. The idea is to take the prefix sum of the array twice. Once from the front end of array and another from the back end of array.

After taking both prefix sums run a loop and check for some  $i$  if both the prefix sum from one array is equal to prefix sum from the second array then that point can be considered as the Equilibrium point.



C++





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSPELLADS

```
int equilibrium(int a[], int n)
{
    if (n == 1)
        return (0);
    int forward[n] = { 0 };
    int rev[n] = { 0 };

    // Taking the prefixsum from front end array
    for (int i = 0; i < n; i++) {
        if (i) {
            forward[i] = forward[i - 1] + a[i];
        }
        else {
            forward[i] = a[i];
        }
    }

    // Taking the prefixsum from back end of array
    for (int i = n - 1; i > 0; i--) {
        if (i <= n - 2) {
            rev[i] = rev[i + 1] + a[i];
        }
        else {
            rev[i] = a[i];
        }
    }

    // Checking if forward prefix sum
    // is equal to rev prefix
    // sum
    for (int i = 0; i < n; i++) {
        if (forward[i] == rev[i]) {
            return i;
        }
    }
    return -1;

    // If You want all the points
    // of equilibrium create
    // vector and push all equilibrium
    // points in it and
    // return the vector
}
```

Driver code

```
int main()
{
    int arr[] = { -7, 1, 5, 2, -4, 3, 0
```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

}

## Java

```
// Java program to find equilibrium
// index of an array
class GFG{

static int equilibrium(int a[], int n)
{
    if (n == 1)
        return (0);

    int[] front = new int[n];
    int[] back = new int[n];

    // Taking the prefixsum from front end array
    for (int i = 0; i < n; i++)
    {
        if (i != 0)
        {
            front[i] = front[i - 1] + a[i];
        }
        else
        {
            front[i] = a[i];
        }
    }

    // Taking the prefixsum from back end of array
    for (int i = n - 1; i > 0; i--)
    {
        if (i <= n - 2)
        {
            back[i] = back[i + 1] + a[i];
        }
        else
        {
            back[i] = a[i];
        }
    }

    // Checking for equilibrium index by
    //compairing front and back sums
    for(int i = 0; i < n; i++)
    {
```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```
}

// If no equilibrium index found, then return -1
return -1;
}

// Driver code
public static void main(String[] args)
{
    int arr[] = { -7, 1, 5, 2, -4, 3, 0 };
    int arr_size = arr.length;

    System.out.println("First Point of equilibrium " +
        "is at index " +
        equilibrium(arr, arr_size));
}
}

// This code is contributed by Lovish Aggarwal
```

## Python3

```
# Python program to find the equilibrium
# index of an array

# Function to find the equilibrium index
def equilibrium(arr):
    left_sum = []
    right_sum = []

    # Iterate from 0 to len(arr)
    for i in range(len(arr)):

        # If i is not 0
        if(i):
            left_sum.append(left_sum[i-1]+arr[i])
            right_sum.append(right_sum[i-1]+arr[len(arr)-1-i])
        else:
            left_sum.append(arr[i])
            right_sum.append(arr[len(arr)-1])

    # Iterate from 0 to len(arr)
    for i in range(len(arr)):
        if(left_sum[i] == right_sum[len(arr) - 1 - i]):
            return(i)
```





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```
# Driver code
arr = [-7, 1, 5, 2, -4, 3, 0]
print('First equilibrium index is ',
      equilibrium(arr))
```

# This code is contributed by Lokesh Sharma

## C#

```
// C# program to find equilibrium
// index of an array
using System;

class GFG{

static int equilibrium(int[] a, int n)
{
    if (n == 1)
        return (0);

    int[] front = new int[n];
    int[] back = new int[n];

    // Taking the prefixsum from front end array
    for(int i = 0; i < n; i++)
    {
        if (i != 0)
        {
            front[i] = front[i - 1] + a[i];
        }
        else
        {
            front[i] = a[i];
        }
    }

    // Taking the prefixsum from back end of array
    for(int i = n - 1; i > 0; i--)
    {
        if (i <= n - 2)
        {
            back[i] = back[i + 1] + a[i];
        }
        else
        {
            back[i] = a[i];
        }
    }
}
```



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```
// comparing front and back sums
for(int i = 0; i < n; i++)
{
    if (front[i] == back[i])
    {
        return i;
    }
}

// If no equilibrium index found, then return -1
return -1;
}

// Driver code
public static void Main(string[] args)
{
    int[] arr = { -7, 1, 5, 2, -4, 3, 0 };
    int arr_size = arr.Length;

    Console.WriteLine("First Point of equilibrium " +
        "is at index " +
        equilibrium(arr, arr_size));
}
}

// This code is contributed by ukasp
```

## Javascript

```
<script>
// Program to find equilibrium index of an array
function equilibrium(a, n)
{
    if (n == 1)
        return (0);
    var forward = new Array(0);
    var rev = new Array(0);

    // Taking the prefixsum from front end array
    for (let i = 0; i < n; i++) {
        if (i) {
            forward[i] = forward[i - 1] + a[i];
        }
        else {
            forward[i] = a[i];
        }
    }
}
```



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

```

    if (i <= n - 2) {
        rev[i] = rev[i + 1] + a[i];
    }
    else {
        rev[i] = a[i];
    }
}

// Checking if forward prefix sum
// is equal to rev prefix
// sum
for (let i = 0; i < n; i++) {
    if (forward[i] == rev[i]) {
        return i;
    }
}
return -1;

// If You want all the points
// of equilibrium create
// vector and push all equilibrium
// points in it and
// return the vector
}

// Driver code
arr = new Array(-7, 1, 5, 2, -4, 3, 0);
n = arr.length;
document.write("First Point of equilibrium is at index "
    + equilibrium(arr, n) + "\n");

// This code is contributed by simranarora5sos
</script>

```

## Output

First Point of equilibrium is at index 3

**Time Complexity:**  $O(N)$

**Space Complexity:**  $O(N)$

As pointed out by Sameer, we can remove the return statement and add a print statement to print all equilibrium indexes instead of returning only one.



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

Please write comments if you find the above codes/algorithms incorrect, or find better ways to solve the same problem.

Attention reader! Don't stop learning now. Get hold of all the important DSA concepts with the [DSA Self Paced Course](#) at a student-friendly price and become industry ready. To complete your preparation from learning a language to DS Algo and many more, please refer [Complete Interview Preparation Course](#).

In case you wish to attend **live classes** with experts, please refer [DSA Live Classes for Working Professionals](#) and [Competitive Programming Live for Students](#).

Like 123

Previous

Next

ADVERTISEMENT BY ADRECOVER





Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

- 01** Rearrange array such that even index elements are smaller and odd index elements are greater

08, Oct 17
- 02** Sort Array such that smallest is at 0th index and next smallest it at last index and so on

08, Jul 20
- 03** Farthest index that can be reached from the Kth index of given array by given operations

02, Dec 20
- 04** Queries to calculate sum of array elements present at every Yth index starting from the index X

13, Jan 21
- 05** Minimum count of indices to be skipped for every index of Array to keep sum till that index at most T

24, Jun 21
- 06** Queries for number of distinct elements from a given index till last index in an array

17, Jun 19
- 07** Length of longest subarray for each index in Array where element at that index is largest

28, Jun 21
- 08** Even numbers at even index and odd numbers at odd index

28, Oct 17

## Article Contributed By :



GeeksforGeeks

## Vote for difficulty

Current difficulty : [Easy](#)

☒ Easy
 ☐ Normal
 ☐ Medium
 ☐ Hard
 ☐ Expert

▲



Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

**Article Tags :** [Adobe](#), [Amazon](#), [Hike](#), [prefix-sum](#), [Arrays](#)

**Practice Tags :** [Amazon](#), [Hike](#), [Adobe](#), [prefix-sum](#), [Arrays](#)

Improve Article

Report Issue

Writing code in comment? Please use [ide.geeksforgeeks.org](https://ide.geeksforgeeks.org), generate link and share the link here.

Load Comments

ADVERTISEMENT BY ADRECOVER



5th Floor, A-118,  
Sector-136, Noida, Uttar Pradesh - 201305

[feedback@geeksforgeeks.org](mailto:feedback@geeksforgeeks.org)

## Company

[About Us](#)

[Careers](#)

[Privacy Policy](#)

[Contact Us](#)

[Copyright Policy](#)

## Learn

[Algorithms](#)

[Data Structures](#)

[Languages](#)

[CS Subjects](#)

[Video Tutorials](#)







Innovieren Sie auf einer Plattform für ein verantwortungsvolles Machine Learning

JETZT MEHR ERFAHREN

HIDE AD • AD VIA BUYSSELLADS

TITLE

WHO WRITING

CSS

Write Interview Experience

JavaScript

Internships

Bootstrap

Videos

@geeksforgeeks , Some rights reserved

