

Get 90% Refund

Courses Tutorials Practice Jobs

Search...

Problem Editorial Submissions Comments

Output Window

Compilation Results Custom Input Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓

Suggest Feedback

Test Cases Passed 1121 / 1121

Attempts : Correct / Total 1 / 1

Accuracy : 100%

Points Scored 4 / 4

Your Total Score: 7 ↑

Time Taken 0.69

Solve Next

Smallest Positive Missing Valid Pair Sum Optimal Array

Stay Ahead With:

Build 21 Projects in 21 Days

Custom Input Compile & Run Submit

Java (21) Start Timer

```
import java.util.Arrays;
class Solution {
    public static int kthSmallest(int[] arr, int k) {
        Arrays.sort(arr);
        return arr[k - 1];
    }
}
```

XAMPP - Search X | XAMPP download | S X | XAMPP Download Su X | What's new in Micros X | chat gpt - Search X | ChatGPT X | Reverse an Array | Pre X

https://www.geeksforgeeks.org/problems/reverse-an-array/1

Get 90% Refund Courses Tutorials Practice Jobs

Problem Editorial Submissions Comments

Output Window

Compilation Results Custom Input

Compilation Completed

Case 1

Input: arr[] =
1 2 3 4

Your Output:
[4, 3, 2, 1]

Expected Output:
[4, 3, 2, 1]

Java (21) Start Timer

```
1 * class Solution {  
2 *     public void reverseArray(int arr[]) {  
3 *         int start = 0;  
4 *         int end = arr.length - 1;  
5 *  
6 *         while (start < end) {  
7 *             // swap  
8 *             int temp = arr[start];  
9 *             arr[start] = arr[end];  
10 *            arr[end] = temp;  
11 *  
12 *            start++;  
13 *            end--;  
14 *        }  
15 *    }  
16 * }  
17 *  
18 *
```

Custom Input Ctrl + C

Compile & Run Submit

 Search...

Get 90% Refund

Courses

Tutorials

Practice

Jobs



Problem

Editorial

Submissions

Comments

Output Window



Compilation Results

Custom Input

Compilation Completed

Case 1

Input:

arr[] =

1 2 3 4

Your Output:

[4, 3, 2, 1]

Expected Output:

[4, 3, 2, 1]

Java (21)

```
1 * class Solution {  
2 *     public void reverseArray(int arr[]) {  
3 *         int start = 0;  
4 *         int end = arr.length - 1;  
5 *         while (start < end) {  
6 *             // swap  
7 *             int temp = arr[start];  
8 *             arr[start] = arr[end];  
9 *             arr[end] = temp;  
10 *            start++;  
11 *            end--;  
12 *        }  
13 *    }  
14 * }
```

Start Timer



Custom Input

Compile & Run

Submit

Get 90% Refund

Courses Tutorials Practice Jobs

Problem Editorial Submissions Comments

Output Window

Compilation Results Custom Input Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓

Suggest Feedback

Test Cases Passed
1115 / 1115

Attempts : Correct / Total
1 / 1

Accuracy : 100%

Points Scored ⓘ
2 / 2

Your Total Score: **2** ⬆

Time Taken
0.74

Solve Next

Mountain Subarray Problem Java ArrayList Operation

Even and odd elements at even and odd positions

Stay Ahead With:

<https://www.geeksforgeeks.org>

Java (21) Start Timer

```
1 class Solution {  
2     public void reverseArray(int arr[]) {  
3         int start = 0;  
4         int end = arr.length - 1;  
5  
6         while (start < end) {  
7             // swap  
8             int temp = arr[start];  
9             arr[start] = arr[end];  
10            arr[end] = temp;  
11  
12            start++;  
13            end--;  
14        }  
15    }  
16}  
17  
18}
```

Custom Input Compile & Run Submit

Get 90% Refund

Courses Tutorials Practice Jobs

Search...

Problem Editorial Submissions Comments

Output Window

Compilation Results Custom Input Y.O.G.I. (AI Bot)

Problem Solved Successfully ✓

Suggest Feedback

Test Cases Passed 1111 / 1111

Attempts : Correct / Total 1 / 1

Accuracy : 100%

Points Scored 1 / 1

Your Total Score: 3 ↑

Time Taken 0.3

Solve Next

Type of array Largest in Array First and Second Smallests

Stay Ahead With:

Build 21 Projects in 21 Days

Custom Input Compile & Run Submit

Java (21) Start Timer

```
import java.util.ArrayList;
class Solution {
    public static ArrayList<Integer> getMinMax(int[] arr) {
        int min = Integer.MAX_VALUE;
        int max = Integer.MIN_VALUE;

        for (int i = 0; i < arr.length; i++) {
            if (arr[i] < min)
                min = arr[i];
            if (arr[i] > max)
                max = arr[i];
        }

        ArrayList<Integer> result = new ArrayList<>();
        result.add(min);
        result.add(max);

        return result;
    }
}
```