

[Dashb...](#) / [My cou...](#) / [CS23331-DAA-202...](#) / [Competitive Progra...](#) / [4-Print Intersection of 2 sorted arrays- \$O\(m+n\)\$ Time Complexity, \$O\(1\)\$ S...](#)

Started on	Tuesday, 12 November 2024, 9:05 AM
State	Finished
Completed on	Tuesday, 12 November 2024, 9:05 AM
Time taken	27 secs
Marks	1.00/1.00
Grade	30.00 out of 30.00 (100%)

Question 1

Correct

Mark 1.00 out of 1.00

Find the intersection of two sorted arrays.

OR in other words,

Given 2 sorted arrays, find all the elements which occur in both the arrays.

Input Format

· The first line contains T, the number of test cases. Following T lines contain:

1. Line 1 contains N1, followed by N1 integers of the first array
2. Line 2 contains N2, followed by N2 integers of the second array

Output Format

The intersection of the arrays in a single line

Example

Input:

```
1
3 10 17 57
6 2 7 10 15 57 246
```

Output:

```
10 57
```

Input:

```
1
6 1 2 3 4 5 6
2 1 6
```

Output:

```
1 6
```

For example:

Input	Result
1 3 10 17 57 6 2 7 10 15 57 246	10 57

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>
2 void findIntersection(int arr1[], int n1, int arr2[], int n2) {
3     int i = 0, j = 0;
4     int first = 1;
5     while (i < n1 && j < n2) {
6         if (arr1[i] < arr2[j]) {
7             i++;
8         } else if (arr1[i] > arr2[j]) {
9             j++;
10        } else {
11            if (first) {
12                printf("%d", arr1[i]);
13                first = 0;
14            } else {
15                printf(" %d", arr1[i]);
16            }
17        }
18    }
19 }
```

```
16         i++;
17         j++;
18     }
19 }
20 }
21 int main() {
22     int T;
23     scanf("%d", &T);
24
25     while (T--) {
26         int n1;
27         scanf("%d", &n1);
28         int arr1[n1];
29
30         for (int i = 0; i < n1; i++) {
31             scanf("%d", &arr1[i]);
32         }
33         int n2;
34         scanf("%d", &n2);
35         int arr2[n2];
36
37         for (int i = 0; i < n2; i++) {
38             scanf("%d", &arr2[i]);
39         }
40         findIntersection(arr1, n1, arr2, n2);
41         printf("\n");
42     }
43     return 0;
44 }
```

	Input	Expected	Got	
✓	1 3 10 17 57 6 2 7 10 15 57 246	10 57	10 57	✓
✓	1 6 1 2 3 4 5 6 2 1 6	1 6	1 6	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◀ 3-Print Intersection of 2 sorted arrays-O(m*n)Time Complexity,O(1) Space Complexity

Jump to...

5-Pair with Difference-O(n^2)Time Complexity,O(1) Space Complexity ▶