

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 find_intersection(int arr1[], int n1, int arr2[], int n2) {
3     int i = 0, j = 0;
4     while (i < n1 && j < n2) {
5         if (arr1[i] == arr2[j]) {
6             printf("%d ", arr1[i]);
7             i++;
8             j++;
9         } else if (arr1[i] < arr2[j]) {
10            i++;
11        } else {
12            j++;
13        }
14    }
15    printf("\n");
```

```
16 }
17 int main() {
18     int T;
19     scanf("%d", &T);
20     while (T--) {
21         int n1, n2;
22         scanf("%d", &n1);
23         int arr1[n1];
24         for (int i = 0; i < n1; i++) {
25             scanf("%d", &arr1[i]);
26         }
27         scanf("%d", &n2);
28         int arr2[n2];
29         for (int i = 0; i < n2; i++) {
30             scanf("%d", &arr2[i]);
31         }
32         find_intersection(arr1, n1, arr2, n2);
33     }
34     return 0;
35 }
36 }
```

	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 ▶