Dashboard / My courses / CS23331-DAA-2023-CSE / Competitive Programming
/ 3-Print Intersection of 2 sorted arrays-O(m*n)Time Complexity,O(1) Space Complexity

Started on	Friday, 25 October 2024, 1:50 PM
State	Finished
Completed on	Friday, 25 October 2024, 2:20 PM
Time taken	29 mins 58 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
	10 57

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

Answer: (penalty regime: 0 %)

```c
1  #include<stdio.h>
2  void find_intersection(int arr1[], int n1, int arr2[], int n2)
3  {
4      int i = 0, j = 0;
5      int found = 0;
6
7      while (i < n1 && j < n2) {
8          if (arr1[i] < arr2[j]) {
9              i++;
10         } else if (arr1[i] > arr2[j]) {
11             j++;
12         } else {
13             if (found) {
14                 printf(" ");
15             }
16             printf("%d", arr1[i]);
17             found = 1;
18             i++;
19             j++;
20         }
21     }
22
23     if (!found) {
24         printf("\n");
25     } else {
26         printf("\n");
27     }
28 }
29
30 int main() {
31     int T;
32     scanf("%d", &T);
33
34     while (T--) {
35         int n1, n2;
36         scanf("%d", &n1);
37         int arr1[n1];
38         for (int i = 0; i < n1; i++) {
39             scanf("%d", &arr1[i]);
40         }
41
42         scanf("%d", &n2);
43         int arr2[n2];
44         for (int i = 0; i < n2; i++) {
45             scanf("%d", &arr2[i]);
46         }
47
48         find_intersection(arr1, n1, arr2, n2);
49     }
50
51     return 0;
52 }
```

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.

◀ 2-Finding Duplicates-O(n) Time Complexity,O(1) Space Complexity

Jump to...

4-Print Intersection of 2 sorted arrays-O(m+n)Time Complexity,O(1) Space Complexity ▶