https://course.acciojob.com/idle?question=1cd14fb1-00a9-4f26-a55b-4aba95c08f5a

MEDIUM

Max Score: 40 Points

Relative Sorting

Ramesh is given two arrays, arr1 and arr2. He wants to sort arr1 in such a way that the relative order of arr2 is maintained in arr1. For elements that are not present in arr2, he wants to add the these elements at the end of the array in sorted fashion.

Can you help Ramesh achieve this task?

Input Format

First line contains two integers `n` and `m` denoting size of arrays `arr1` and `arr2` respectively.

Second line contains `n` space separated integers denoting the array `arr1`.

Third line contains `m` space separated integers denoting the array `arr2`.

Output Format

Print the relatively sorted `arr1`

Example 1

Input

`11 4`

`2 1 2 5 7 1 9 3 6 8 8`

`2 1 8 3`

Output

```
2 2 1 1 8 8 3 5 6 7 9
```

Explanation

arr1 is sorted according to arr2 elements, hence 2 comes before 1, 1 before 8 and 8 before 3. Elements not in arr2 are sorted and appended at the end of arr1.

Example 2

Input

```
6 2
```

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

```
3 1
```

Output

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

Explanation

arr1 is sorted according to arr2 elements, hence 3 comes before 1. Elements not in arr2 are sorted and appended at the end of arr1.

Constraints

1<= n <= 10^5

0 <= arr1[i], arr2[i] <= 10^9

Topic Tags

Hashing

Sorting

My code

```
// in java
import java.util.*;
import java.lang.*;
import java.io.*;


public class Main {
static void relativeSorting(int arr1[], int n, int arr2[], int m)
    {
    // Write your code here, print output
        HashMap<Integer,Integer>hm=new HashMap<>();
        Arrays.sort(arr1);
        for(int i=0;i<n;i++)
        hm.put(arr1[i],hm.getOrDefault(arr1[i],0)+1);
        for(int i=0;i<m;i++)
            {
                int a=0;
                if(hm.containsKey(arr2[i]))
                {
                    a=hm.get(arr2[i]);
                    for(int j=0;j<a;j++)
                        {
                            System.out.print(arr2[i]+" ");
                        }
                }
            }
        HashMap<Integer,Integer>hm2=new HashMap<>();
        for(int i=0;i<m;i++)
            {
                hm2.put(arr2[i],1);
            }
        for(int i=0;i<n;i++)
                    {
                        if(!hm2.containsKey(arr1[i]))
                            System.out.print(arr1[i]+" ");
                    }

    }
  public static void main(String[] args) throws Throwable {
    Scanner sc = new Scanner(System.in);
    int n = sc.nextInt();
    int m = sc.nextInt();
    int arr1[]=new int[n] ;
    int arr2[]=new int[m] ;
    for(int i=0;i<n;i++){
      arr1[i]=sc.nextInt();
    }
    for(int i=0;i<m;i++){
      arr2[i]=sc.nextInt();
    }

    relativeSorting(arr1, n, arr2 , m);
}
}
```