Given two arrays of integers, find which elements in the second array are missing from the first array.

Example

$arr = [7, 2, 5, 3, 5, 3]$

$brr = [7, 2, 5, 4, 6, 3, 5, 3]$

The brr array is the orginal list. The numbers missing are $[4, 6]$.

Notes

- If a number occurs multiple times in the lists, you must ensure that the frequency of that number in both lists is the same. If that is not the case, then it is also a missing number.
- Return the missing numbers sorted ascending.
- Only include a missing number once, even if it is missing multiple times.
- The difference between the maximum and minimum numbers in the original list is less than or equal to 100.

Function Description

Complete the missingNumbers function in the editor below. It should return a sorted array of missing numbers.

missingNumbers has the following parameter(s):

- int arr[n]: the array with missing numbers
- int brr[m]: the original array of numbers

Returns

- int[]: an array of integers

Input Format

There will be four lines of input:

n - the size of the first list, arr

The next line contains n space-separated integers $arr[i]$

m - the size of the second list, brr

The next line contains m space-separated integers $brr[i]$

Constraints

- $1 \leq n, m \leq 2 \times 10^5$
- $n \leq m$
- $1 \leq brr[i] \leq 10^4$
- $max(brr) - min(brr) \leq 100$

Sample Input

```
10
203 204 205 206 207 208 203 204 205 206
13
203 204 204 205 206 207 205 208 203 206 205 206 204
```

Sample Output

```
204 205 206
```

Explanation

204 is present in both arrays. Its frequency in arr is 2, while its frequency in brr is 3. Similarly, 205 and 206 occur twice in arr, but three times in brr. The rest of the numbers have the same frequencies in both lists.

```python
#!/bin/python3

import math
import os
import random
import re
import sys

#
# Complete the 'missingNumbers' function below.
#
# The function is expected to return an INTEGER_ARRAY.
# The function accepts following parameters:
#  1. INTEGER_ARRAY arr
#  2. INTEGER_ARRAY brr
#

def missingNumbers(arr, brr):
    # Write your code here
    result = []
    map_brr = {}

    # build a frequency map for all the numbers in the original array
    for number in brr:
        map_brr[number] = map_brr.get(number, 0) + 1

    # update the frequency map as follows
    # frequency(element x in brr) - frequency(element x in arr) NOTE: if element x in arr
    # else can't change, don't change
    for number in arr:
        map_brr[number] -= 1

    # if frequency of x in the map equals zero
    # that means frequency of x is same in both arrays
    # if not, means it's missing
    # add all the missing elements to the resultant array
    for key, value in map_brr.items():
        if value != 0:
            result.append(key)

    # sort the resultant array before returning
    result.sort()
    return result
```

Line: 34 Col: 55

Upload Code as File Test against custom input Run Code Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends?

Next Challenge

Test case 0

Test case 1

Test case 2

Test case 3

Compiler Message

Success

Input (stdin) Download

```
10
203 204 205 206 207 208 203 204 205 206
```