



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## Experiment1.4

Student Name: Virat Samdarshi

Branch: B.E(CSE)

Semester: Fifth

Subject Name: AP LAB 1

UID: 22BCS12648

Section/Group: IOT\_627-B

Date of Performance: 08/08/24

Subject Code: 22CSP-314

1. **Aim:** Given two arrays of integers, find which elements in the second array are missing from the first array.
2. **Objective:** Identify and return the elements that are present in the second array but absent in the first array.
3. **Implementation/Code :**

```
1  #include <bits/stdc++.h>
2
3  using namespace std;
4
5  string ltrim(const string &);
6  string rtrim(const string &);
7  vector<string> split(const string &);
8
9  vector<int> missingNumbers(vector<int> arr, vector<int> brr) {
10     vector<int>ans;
11     unordered_map<int,int>map;
12     for(int i=0;i<brr.size();i++){
13         map[brr[i]]++;
14     }
15     for(int i=0;i<arr.size();i++){
16         map[arr[i]]--;
17     }
18     for (auto &i : map) {
19         if (i.second > 0) {
20             ans.push_back(i.first);
21         }
22     }
23     sort(ans.begin(), ans.end());
24     return ans;
25 }
26
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## 4. Output :-

```
Compilation Successful :)
Click the Submit Code button to run your code against all the test cases.

Input (stdin)                                     Download
1 5
2 1 2 3 4 5
3 8
4 1 3 2 4 5 7 6 8

Your Output (stdout)
1 6 7 8
```

## 5. Time Complexity: $O(n + m + m \log m)$

## PROBLEM 2

1. **Aim:** Sorting is useful as the first step in many different tasks. The most common task is to make finding things easier, but there are other uses as well. In this case, it will make it easier to determine which pair or pairs of elements have the smallest absolute difference between them.

2. **Objective:** The objective is to sort the elements to simplify the process of identifying the pair or pairs with the smallest absolute difference between them.

3. **Implementation/Code :-**

```
1  #include <bits/stdc++.h>
2
3  using namespace std;
4
5  string ltrim(const string &);
6  string rtrim(const string &);
7  vector<string> split(const string &);
8
9
10 vector<int> closestNumbers(vector<int> arr) {
11     sort(arr.begin(), arr.end());
12     int difference = INT_MAX;
13     vector<int> result;
14     for(int i = 1; i < arr.size(); i++){
15         if(arr[i] - arr[i-1] < difference){
16             difference = arr[i] - arr[i-1];
17             result = {arr[i-1], arr[i]};
18         }
19         else if(arr[i] - arr[i-1] == difference){
20             result.insert(result.end(), {arr[i-1], arr[i]});
21         }
22     }
23     return result;
24 }
25
26 }
```



# DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

## 4. Output :-

```
Compilation Successful :)
Click the Submit Code button to run your code against all the test cases.

Input (stdin)                                     Download
1 5
2 1 5 2 4 6

Your Output (stdout)
1 1 2 4 5 5 6
```

## 5. Time Complexity: $O(n \log n)$