**ALGORITHM:-**

* ACCEPT TWO ARRAYS FROM USER.
* SORT THE ARRAYS TO GET INTERSECTION
* COUNT THE FREQUENCY OF SIMILAR ARRAYS
* IF SIMILAR VALUES FOUND PRINT THE VALUES AS OUTPUT.

**PROGRAM IN C++:-**

#include <bits/stdc++.h>

using namespace std;

// Function to find intersection of arrays

void intersectionofarrays(int p[], int g[], int o, int k)

{

int i = 0, j = 0;

while (i < o && j < k) {

if (p[i] > g[j]) {

j++;

}

else if (g[j] > p[i]) {

i++;

}

else {

// when both are equal

cout << p[i] << " ";

i++;

j++;

}

}

}

int main()

{

int n1,n2;

cout<<"Enter Size of array 1: "<<endl;

cin>> n1;

int p[n1] ;//taking array input as per example,1st array

cout<<"Enter Size of array 2: "<<endl;

cin>> n2;

int g[n2]; //2nd array

cout<<"Enter Elements of array 1 : "<<endl; // accept array 1 from user

for(int i=0;i<n1;i++){

cin>>p[i];

}

cout<<"Enter Elements of array 2 : "<<endl; // accept array 1 from user

for(int j=0;j<n2;j++){

cin>>g[j];

}

// sorting the elements

sort(p, p + n1);

sort(g, g + n2);

// calling the function

intersectionofarrays(p, g, n1, n2);

}

**OUTPUT:-**