**Day-43 of the #101 days of coding challenge------**

**Problem:-** Write a C++ program to find the two repeating elements in a given array of integers.

Code:-

#include <iostream>

using namespace std;

void findTwoRepeatnumber(int \*arr, int n)

{

int i, j, count = 0;

for(i = 0; i<n; i++)

{

count = 0;

for(j = i; j<n; j++)

{

if(arr[i] == arr[j])

{

count++;

if(count == 2)

{

cout<<arr[i];

break;

}

}

}

cout<<" ";

}

}

int main() {

int n;

cout<<"Enter the size of the array"<<endl;

cin>>n;

int arr[n];

cout<<"Enter the elements"<<endl;

for(int i = 0; i<n; i++)

{

cin>>arr[i];

}

cout<<"Solved"<<endl;

findTwoRepeatnumber(arr,n);

return 0;

}

Output:-

A white background with black text

Description automatically generated

2nd Method:-

Replace the logic only:-

void findTwoRepeatnumber(int \*arr, int n)

{

int i, j, count = 0;

for(i = 0; i<n; i++)

{

for(j = i+1; j<n; j++)

{

if(arr[i] == arr[j])

cout<<arr[i]<<" ";

}

}

}

Problem2:- Write a C++ program to find the missing element from two given arrays of integers except one element.

Code:-

#include <iostream>

using namespace std;

void findmisingElementTwoarray(int \*arrFirst, int \*arrSecond, int n, int n1)

{

int i, j, count = 0;

// only we need to check till first array

for(i = 0; i<n; i++)

{

if (arrFirst[i] != arrSecond[i])

cout<<arrFirst[i]<<" ";

}

}

int main() {

int n, n1;

cout<<"Enter the size of the array two arrays"<<endl;

cin>>n;

int arrFirst[n], arrSecond[n1];

cout<<"Enter the elements of first array"<<endl;

for(int i = 0; i<n; i++)

{

cin>>arrFirst[i];

}

cout<<"Enter the elements of second array"<<endl;

for(int i = 0; i<n; i++)

{

cin>>arrSecond[i];

}

cout<<"Solved"<<endl;

findmisingElementTwoarray(arrFirst, arrSecond,n, n1);

return 0;

}

Output:-

A white background with black text

Description automatically generated

Problem:- Write a C++ program to find and print all common elements in three sorted arrays of integers.

Code:-

#include <iostream>

using namespace std;

void findmisingElementThreesortedArray(int \*arrFirst, int \*arrSecond, int \*arrThird, int n1, int n2, int n3)

{

int i, count = 0;

// finding the minimum length of the array

int minLength;

if (n1 < n2 && n1 < n3)

minLength = n1;

else if (n2 < n1 && n2 < n3)

minLength = n2;

else

minLength = n3;

// only we need to check up to the minimum length

for (i = 0; i < minLength; i++)

{

if (arrFirst[i] == arrSecond[i] && arrSecond[i] == arrThird[i])

{

cout << arrFirst[i] << " ";

count++;

}

}

if (count == 0)

cout << "No matching element is found" << endl;

}

int main() {

int n1, n2, n3;

cout << "Enter the size of the three arrays" << endl;

cin >> n1 >> n2 >> n3;

int arrFirst[n1], arrSecond[n2], arrThird[n3];

cout << "Enter the elements of the first array" << endl;

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

{

cin >> arrFirst[i];

}

cout << "Enter the elements of the second array" << endl;

for (int i = 0; i < n2; i++)

{

cin >> arrSecond[i];

}

cout << "Enter the elements of the third array" << endl;

for (int i = 0; i < n3; i++)

{

cin >> arrThird[i];

}

cout << "Solved" << endl;

findmisingElementThreesortedArray(arrFirst, arrSecond, arrThird, n1, n2, n3);

return 0;

}

Output:-

