

# Classwork

## Swap alternate elements in an array in C++

```
#include<iostream>
using namespace std;

void alternateSwap(int arr[],int arraySize){
    for(int i=0;i<arraySize;i=i+2){
        if((i+1)<(arraySize)){
            swap(arr[i],arr[i+1]);
        }
    }

    cout<<"Array, After alternate swaping "<<endl;
    for(int i=0;i<arraySize;i++){
        cout<<arr[i]<<" ";
    }
}

int main(){
    int arr[1000];
    int arraySize;
    cout<<"Enter size of array "<<endl;
    cin>>arraySize;

    cout<<"Enter elements in an array"<<endl;
    for(int i=0;i<arraySize;i++){
        cin>>arr[i];
    }

    cout<<"Array, Before alternate swaping "<<endl;
    for(int i=0;i<arraySize;i++){
        cout<<arr[i]<<" ";
    }
    cout<<endl;

    alternateSwap(arr,arraySize);
}
```

## Swap alternate elements in an array in Java

```
package com.help.code;
import java.util.*;
public class SwapAlternate {

    public static void swapArrayAlternatively(int arr[],int
arraySize) {
        for(int i=0;i<arraySize;i=i+2) {
            if((i+1)<arraySize) {
                int temp=arr[i];
                arr[i]=arr[i+1];
                arr[i+1]=temp;
            }
        }

        System.out.println("Array, After alternate swaping ");
        for(int i=0;i<arraySize;i++) {
            System.out.print(arr[i]+" ");
        }

        }

    public static void main(String[] args) {
        // TODO Auto-generated method stub
        Scanner sc=new Scanner(System.in);
        int[] arr=new int[1000];
        System.out.println("Enter size of array ");
        int arraySize=sc.nextInt();

        System.out.println("Enter elements in an array");
        for(int i=0;i<arraySize;i++) {
            arr[i]=sc.nextInt();
        }

        System.out.println("Array, Before alternate swaping
");
        for(int i=0;i<arraySize;i++) {
            System.out.print(arr[i]+" ");
        }
        System.out.println();
        swapArrayAlternatively(arr,arraySize);
    }
}
```

```
}
```

## Find Unique [Coding Ninjas]

Question Link:-

[https://www.codingninjas.com/studio/problems/find-unique\\_625159?source=youtube&campaign=love\\_babbar\\_codestudio1&utm\\_source=youtube&utm\\_medium=affiliate&utm\\_campaign=love\\_babbar\\_codestudio1&leftPanelTab=0](https://www.codingninjas.com/studio/problems/find-unique_625159?source=youtube&campaign=love_babbar_codestudio1&utm_source=youtube&utm_medium=affiliate&utm_campaign=love_babbar_codestudio1&leftPanelTab=0)



### C++ code

```
int findUnique(int *arr, int size)
{
    //Write your code here
    int ans=0;
    for(int i=0;i<size;i++){
        ans=ans^arr[i];
    }
    return ans;
}
```

## Duplicate In Array [Coding Ninjas]

Question Link:-

[https://www.codingninjas.com/studio/problems/duplicate-in-array\\_893397?source=youtube&campaign=love\\_babbar\\_codestudio1&utm\\_source=youtube&utm\\_medium=affiliate&utm\\_campaign=love\\_babbar\\_codestudio1&leftPanelTab=0](https://www.codingninjas.com/studio/problems/duplicate-in-array_893397?source=youtube&campaign=love_babbar_codestudio1&utm_source=youtube&utm_medium=affiliate&utm_campaign=love_babbar_codestudio1&leftPanelTab=0)

Current Submission Report

**Correct Answer** Test Cases EXP: 11/11 Penalty 40/40 0% Java

few secs ago

Runtime graph Memory graph

You performed better than 35.19% Runtime 890 ms

Well done  
You are in the highest level. Keep coding

```

1 import java.util.ArrayList;
2
3 public class Solution {
4
5     public static int findDuplicate(ArrayList<Integer> arr) {
6         // Write your code here.
7         int ans=0;
8         for(int i=0;i<arr.size();i++){
9             ans=ans^arr.get(i);
10        }
11
12        for(int i=1;i<arr.size();i++){
13            ans=ans^i;
14        }
15        return ans;
16    }
17 }

```

< Prev Next > View hints Last saved on 8:08:03 PM Run Submit

## Java Code

```

import java.util.ArrayList;

public class Solution {

    public static int findDuplicate(ArrayList<Integer> arr) {
        // Write your code here.
        int ans=0;
        for(int i=0;i<arr.size();i++){
            ans=ans^arr.get(i);
        }

        for(int i=1;i<arr.size();i++){
            ans=ans^i;
        }
        return ans;
    }
}

```

## Intersection Of Two Sorted Arrays [Coding Ninjas]

Question Link:-

[https://www.codingninjas.com/studio/problems/intersection-of-2-arrays\\_1082149?source=youtube&campaign=love\\_babbar\\_codestudio1&utm\\_source=youtube&utm\\_medium=affiliate&utm\\_campaign=love\\_babbar\\_codestudio1&leftPanelTab=0](https://www.codingninjas.com/studio/problems/intersection-of-2-arrays_1082149?source=youtube&campaign=love_babbar_codestudio1&utm_source=youtube&utm_medium=affiliate&utm_campaign=love_babbar_codestudio1&leftPanelTab=0)

[Learn](#)
[Contests & Events](#)
[Interview prep](#)
[Practice](#)
[Resources](#)

Problem of the day

Your EXP 276

Topics

Problem

Submissions

Solution

Discuss

C++ (g++ 5.4)

Autocomplete

Current Submission

Report

Correct Answer

Test Cases EXP: 9/9

Penalty 0%

C++

few secs ago

40/40

Runtime graph

Memory graph

You performed better than

71.69%

Runtime

721 ms

Did you find these test cases useful?

👍

👎

1

vector<int> findArrayIntersection(vector<int> &arr1, int n, vector<int> &arr2, int m)

2

{

3

// Write your code here.

4

vector<int> arr;

5

int i=0;

6

int j=0;

7

while(i<n && j<m){

8

if(arr1[i]==arr2[j]){

9

arr.push\_back(arr1[i]);

10

i++;

11

j++;

12

}else if(arr1[i]<arr2[j]){

13

i++;

14

}else{

15

j++;

16

}

17

}

18

return arr;

19

}

Console

## C++ Code

```
vector<int> findArrayIntersection(vector<int> &arr1, int n, vector<int>
&arr2, int m)
{
    // Write your code here.
    vector<int> arr;
    int i=0;
    int j=0;
    while(i<n && j<m){
        if(arr1[i]==arr2[j]){
            arr.push_back(arr1[i]);
            i++;
            j++;
        }else if(arr1[i]<arr2[j]){
            i++;
        }else{
            j++;
        }
    }
    return arr;
}
```

## Pair Sum [Coding Ninjas]

Question Link:-

[https://www.codingninjas.com/studio/problems/pair-sum\\_697295?leftPanelTab=0](https://www.codingninjas.com/studio/problems/pair-sum_697295?leftPanelTab=0)

codingninjas /studio

Learn Contests & Events Interview prep Practice Resources

Problem of the day Your EXP 316

Topics Problem Submissions Solution Discuss

Current Submission [Report](#)

Correct Answer Test Cases EXP: 8/8 Penalty 0% Java  
few secs ago 40/40

Runtime graph Memory graph

You performed better than 23.29% Runtime 1418 ms

Best

1.064 1.163 1.262 1.361 1.460  
Runtime (ms)

Did you find these test cases useful? [👍](#) [👎](#)

```
1 import java.io.*;
2 import java.util.* ;
3
4 public class Solution{
5     public static List<int[]> pairSum(int[] arr, int s) {
6         // Write your code here.
7         List<int[]> result= new ArrayList<>();
8         Arrays.sort(arr);
9         for(int i=0;i<arr.length;i++){
10             for(int j=i+1;j<arr.length;j++){
11                 if(arr[i]+arr[j]==s){
12                     int[] a={arr[i],arr[j]};
13                     result.add(a);
14                 }
15             }
16         }
17         return result;
18     }
19 }
20
```

< Prev Next > View hints Last saved on 10:49:42 PM Run Submit

## Java Code

```
import java.io.*;
import java.util.* ;

public class Solution{
    public static List<int[]> pairSum(int[] arr, int s) {
        // Write your code here.
        List<int[]> result= new ArrayList<>();
        Arrays.sort(arr);
        for(int i=0;i<arr.length;i++){
            for(int j=i+1;j<arr.length;j++){
                if(arr[i]+arr[j]==s){
                    int[] a={arr[i],arr[j]};
                    result.add(a);
                }
            }
        }
        return result;
    }
}
```

# Homework

## Unique Number of Occurrences [LeetCode]

Question Link:-

<https://leetcode.com/problems/unique-number-of-occurrences/description/>

The screenshot shows the LeetCode interface for the problem '1207. Unique Number of Occurrences'. The problem is marked as 'Solved'. The description states: 'Given an array of integers arr, return true if the number of occurrences of each value in the array is unique or false otherwise.' An example is provided: 'Input: arr = [1,2,2,1,1,3]' and 'Output: true'. The solution is written in Java, using a HashMap to count occurrences and a HashSet to check for uniqueness. The code is as follows:

```
1 class Solution {
2     public boolean uniqueOccurrences(int[] arr) {
3         // Create a HashMap to store the count of occurrences for each value
4         Map<Integer, Integer> countMap = new HashMap<>();
5
6         // Count the occurrences of each value in the array
7         for (int num : arr) {
8             countMap.put(num, countMap.getOrDefault(num, 0) + 1);
9         }
10
11        // Create a HashSet to store the unique counts
12        Set<Integer> uniqueCounts = new HashSet<>();
13
14        // Check if the count of occurrences is unique for each value
15        for (int count : countMap.values()) {
16            if (uniqueCounts.contains(count)) {
17                return false;
18            }
19            uniqueCounts.add(count);
20        }
21
22        return true;
23    }
24 }
```

### Java Code

```
class Solution {
    public boolean uniqueOccurrences(int[] arr) {
        // Create a HashMap to store the count of occurrences for each value
        Map<Integer, Integer> countMap = new HashMap<>();

        // Count the occurrences of each value in the array
        for (int num : arr) {
            countMap.put(num, countMap.getOrDefault(num, 0) + 1);
        }

        // Create a HashSet to store the unique counts
        Set<Integer> uniqueCounts = new HashSet<>();

        // Check if the count of occurrences is unique for each value
        for (int count : countMap.values()) {
```

```

        if (uniqueCounts.contains(count)) {
            return false;
        }
        uniqueCounts.add(count);
    }

    return true;
}
}

```

## Find all duplicates in an Array [LeetCode]

Question Link:-

<https://leetcode.com/problems/find-all-duplicates-in-an-array/description/>

**442. Find All Duplicates in an Array** Solved ✓

Medium Topics Companies

Given an integer array `nums` of length `n` where all the integers of `nums` are in the range `[1, n]` and each integer appears **once** or **twice**, return an array of all the integers that appears **twice**.

You must write an algorithm that runs in  $O(n)$  time and uses only constant extra space.

**Example 1:**

9.2K 326 5 0

Testcase Result

Accepted Editorial + Solution

Runtime	Details	Memory	Details
3 ms		52.94 MB	
Beats 99.94% of users with Java		Beats 87.80% of users with Java	

```

1 class Solution {
2     public List<Integer> findDuplicates(int[] nums) {
3         List<Integer> result=new ArrayList<Integer>();
4         int[] temp=new int[nums.length+1];
5         for(int i=0;i<nums.length;i++){
6             temp[nums[i]]++;
7         }
8
9         for(int i=0;i<temp.length;i++){
10             if(temp[i]==2){
11                 result.add(i);
12             }
13         }
14         return result;
15     }
16 }

```

Saved to local Ln 9, Col 17

## Java Code

```

class Solution {
    public List<Integer> findDuplicates(int[] nums) {
        List<Integer> result=new ArrayList<Integer>();
        int[] temp=new int[nums.length+1];
        for(int i=0;i<nums.length;i++){
            temp[nums[i]]++;
        }

        for(int i=0;i<temp.length;i++){
            if(temp[i]==2){
                result.add(i);
            }
        }
    }
}

```



```

    }
    return result;
}
}

```

## Sort 0 1 2 [Coding Ninjas]

Question Link:-

[https://www.codingninjas.com/studio/problems/sort-0-1-2\\_631055?source=youtube&campaign=LoveBabbar\\_Codestudiovideo1&utm\\_source=youtube&utm\\_medium=affiliate&utm\\_campaign=LoveBabbar\\_Codestudiovideo1&leftPanelTab=0](https://www.codingninjas.com/studio/problems/sort-0-1-2_631055?source=youtube&campaign=LoveBabbar_Codestudiovideo1&utm_source=youtube&utm_medium=affiliate&utm_campaign=LoveBabbar_Codestudiovideo1&leftPanelTab=0)

The screenshot shows the Coding Ninjas Studio interface. On the left, the 'Current Submission' panel displays a 'Correct Answer' status, test cases (8/8), and a runtime of 4803 ms. Below this is a 'Runtime graph' showing performance over time. The main editor area shows the Java code for the 'sort012' method. The code is as follows:

```

1 import java.util.* ;
2 import java.io.*;
3 public class Solution
4 {
5     //Write your code here
6     int i=0;
7     int left=0;
8     int right=arr.length-1;
9     int temp;
10    while(i<=right){
11        if(arr[i]==0){
12            temp=arr[i];
13            arr[i]=arr[left];
14            arr[left]=temp;
15            i++;
16            left++;
17        }
18        else if(arr[i]==2){
19            temp=arr[i];
20            arr[i]=arr[right];
21            arr[right]=temp;
22            right--;
23        }
24    }
25 }

```

## Java Code

```

import java.util.* ;
import java.io.*;
public class Solution
{
    public static void sort012(int[] arr)
    {
        //Write your code here
        int i=0;
        int left=0;
        int right=arr.length-1;
        int temp;
        while(i<=right){
            if(arr[i]==0){

```

```
        temp=arr[i];
        arr[i]=arr[left];
        arr[left]=temp;
        i++;
        left++;
    }
    else if(arr[i]==2){
        temp=arr[i];
        arr[i]=arr[right];
        arr[right]=temp;
        right--;
    }
    else{
        i++;
    }
}
}
```