***Remove duplicates from a sorted array***

Given a sorted array, the task is to remove the duplicate elements from the array.

**Examples:**

Input : arr[] = {2, 2, 2, 2, 2}

Output : arr[] = {2}

new size = 1

Input : arr[] = {1, 2, 2, 3, 4, 4, 4, 5, 5}

Output : arr[] = {1, 2, 3, 4, 5}

new size = 5

**Method 1:** (Using extra space)

C++Java

import java.util.\*;

import java.io.\*;

import java.lang.\*;

class GFG

{

static int remDups(int arr[], int n)

{

int temp[] = new int[n];

temp[0] = arr[0];

int res = 1;

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

{

if(temp[res - 1] != arr[i])

{

temp[res] = arr[i];

res++;

}

}

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

{

arr[i] = temp[i];

}

return res;

}

public static void main(String args[])

{

int arr[] = {10, 20, 20, 30, 30, 30}, n = 6;

System.out.println("Before Removal");

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

{

System.out.print(arr[i]+" ");

}

System.out.println();

n = remDups(arr, n);

System.out.println("After Removal");

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

{

System.out.print(arr[i]+" ");

}

}

}

**Output**

**Before Removal**

**10 20 20 30 30 30**

**After Removal**

**10 20 30**

**Time Complexity : O(n)**  
**Auxiliary Space : O(n)**

**Method 2:** (Constant extra space)

C++Java

import java.util.\*;

import java.io.\*;

import java.lang.\*;

class GFG

{

static int remDups(int arr[], int n)

{

int res = 1;

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

{

if(arr[res - 1] != arr[i])

{

arr[res] = arr[i];

res++;

}

}

return res;

}

public static void main(String args[])

{

int arr[] = {10, 20, 20, 30, 30, 30}, n = 6;

System.out.println("Before Removal");

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

{

System.out.print(arr[i]+" ");

}

System.out.println();

n = remDups(arr, n);

System.out.println("After Removal");

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

{

System.out.print(arr[i]+" ");

}

}

}

**Output**

**Before Removal**

**10 20 20 30 30 30**

**After Removal**

**10 20 30**

**Time Complexity : O(n)**  
**Auxiliary Space : O(1)**