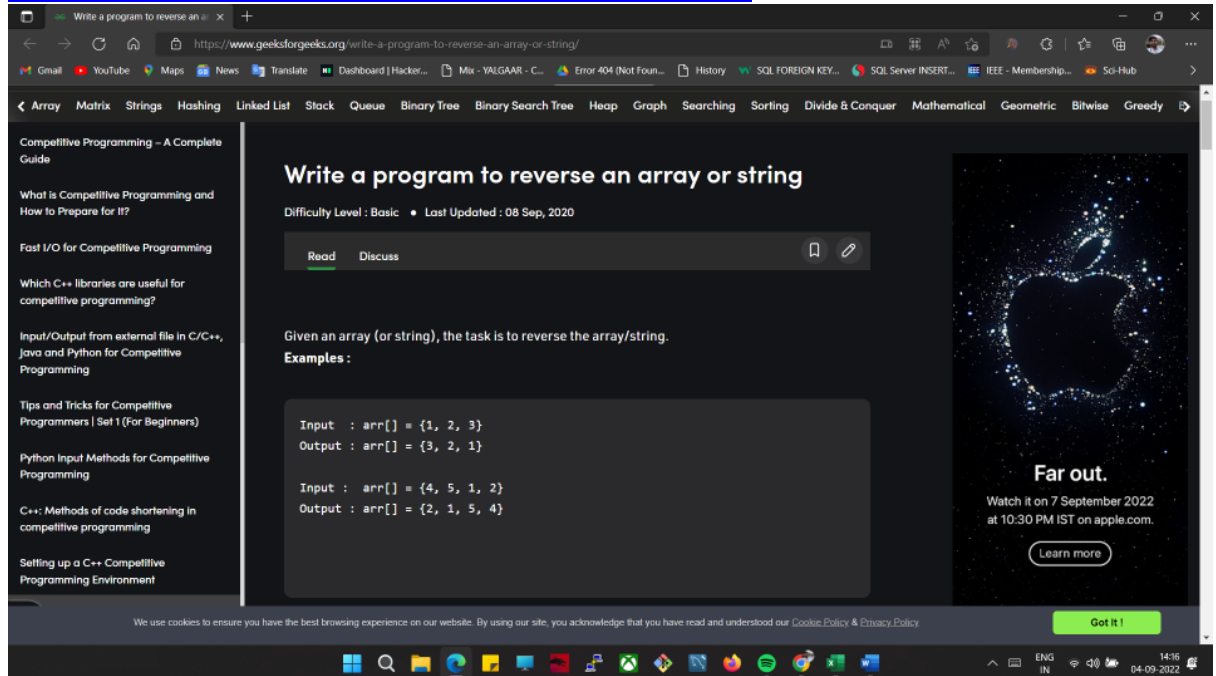


DSA Questions by Love Babbar:-

TOPIC :- ARRAY

1. Reverse the array

[Write a program to reverse an array or string - GeeksforGeeks](https://www.geeksforgeeks.org/write-a-program-to-reverse-an-array-or-string/)



Solution :- `#include<bits/stdc++.h>`

`using namespace std;`

`int main(){`

`int N;`

`cin >> N;`

`vector<long long>v(N);`

`for(int i=0;i<N;i++){`

`cin >> v[i];`

`}`

`for(int i=N-1;i>=0;i--){`

```

    cout << v[i] << " ";
}
}

```

2. Find the maximum and minimum element in an array

The screenshot shows the GeeksforGeeks website with the article "Maximum and minimum of an array using minimum number of comparisons". The article is categorized under "Divide and Conquer" and "Divide and Conquer Algorithm | Introduction". The difficulty level is "Easy" and it was last updated on 02 Sep, 2022. The article describes a task to find the maximum and minimum element of an array of size N using the minimum number of comparisons. It provides two examples:

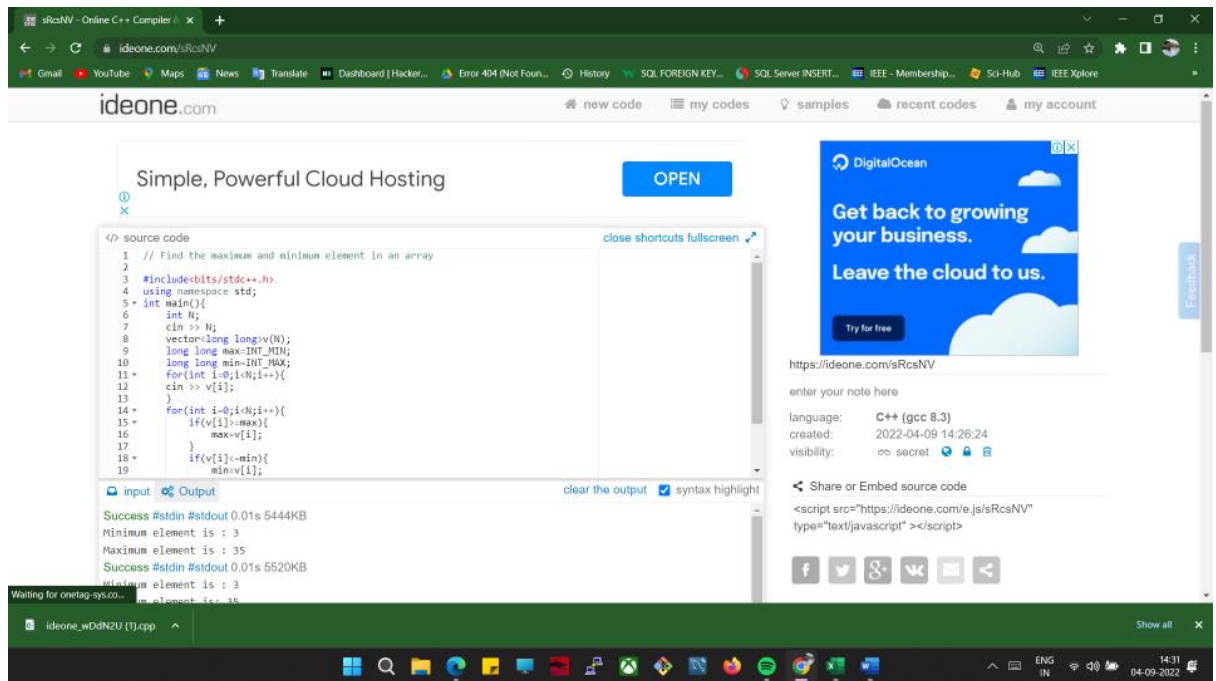
```

Input: arr[] = {3, 5, 4, 1, 9}
Output: Minimum element is: 1
        Maximum element is: 9

Input: arr[] = {22, 14, 8, 17, 35, 3}
Output: Minimum element is: 3
        Maximum element is: 35

```

The website also features a sidebar with navigation links for various topics like Array, Matrix, Strings, Hashing, Linked List, Stack, Queue, Binary Tree, Binary Search Tree, Heap, Graph, Searching, Sorting, Divide & Conquer, Mathematical, Geometric, Bitwise, and Greedy. A cookie notice is visible at the bottom of the page.



Solution :- // Find the maximum and minimum element in an array

```
#include<bits/stdc++.h>
using namespace std;
int main(){
    int N;
    cin >> N;
    vector<long long>v(N);
    long long max=INT_MIN;
    long long min=INT_MAX;
    for(int i=0;i<N;i++){
        cin >> v[i];
    }
    for(int i=0;i<N;i++){
```

```

        if(v[i]>=max){
            max=v[i];
        }
        if(v[i]<=min){
            min=v[i];
        }
    }

    cout << "Minimum element is : " << min
    << "\n";
    cout << "Maximum element is : " << max;
}

```

3. Find the "Kth" max and min element of an array

[Kth smallest element | Practice | GeeksforGeeks](https://practice.geeksforgeeks.org/problems/kth-smallest-element5635/1)

Kth smallest element ★

Medium Accuracy: 46.66% Submissions: 100k+ Points: 4

This problem is part of GFG SDE Sheet. Click here to view more.

Given an array `arr[]` and an integer `K` where `K` is smaller than size of array, the task is to find the **Kth smallest** element in the given array. It is given that all array elements are distinct.

Example 1:

Output Window

Compilation Results Custom Input

Problem Solved Successfully ✓

Test Cases Passed: 156 / 156	Total Points Scored: 4 / 4
Your Total Score:	Total Time Taken:

```

1 // Driver Code Ends
2 // User function template for C++
3
4 class Solution{
5 public:
6     // arr : given array
7     // l : starting index of the array i.e 0
8     // r : ending index of the array i.e size-1
9     // k : find kth smallest element and return using this function
10    int kthSmallest(int arr[], int l, int r, int k) {
11
12        sort(arr, arr+r+1);
13
14        return arr[k-1];
15    }
16 };
17
18
19
20
21
22
23
24
25

```

Compile & Run Submit

Solution :- class Solution{
public:

```
// arr : given array
// l : starting index of the array i.e 0
// r : ending index of the array i.e size-1
// k : find kth smallest element and return using
this function
```

```
int kthSmallest(int arr[], int l, int r, int k) {

    sort(arr, arr+r+1);

    return arr[k-1];

}
```

```
};
```

4. Given an array which consists of only 0, 1 and 2.
Sort the array without using any sorting algo

[Sort an array of 0s, 1s and 2s | Practice | GeeksforGeeks](https://practice.geeksforgeeks.org/problems/sort-an-array-of-0s-1s-and-2s/1/)

The screenshot shows the GeeksforGeeks practice interface for the problem 'Sort an array of 0s, 1s and 2s'. The problem is marked as 'Easy' with an accuracy of 51.36% and over 100k submissions. It is part of the GFG SDE Sheet. The problem description states: 'Given an array of size N containing only 0s, 1s, and 2s; sort the array in ascending order.' An example is provided: Input: N = 5, arr[] = {0 2 1 2 0}, Output: 0 0 1 2 2. The explanation says: '0s 1s and 2s are segregated into ascending order.' The code editor on the right shows a C++ solution template with a class 'Solution' and a method 'sort012' that needs to be implemented. The bottom of the screen shows a Windows taskbar with various application icons and the system clock at 15:00 on 04-09-2022.

Sort an array of 0s, 1s and 2s

Easy Accuracy: 51.36% Submissions: 100k+ Points: 2

This problem is part of GFG SDE Sheet. Click here to view more.

Given an array of size N containing only 0s, 1s, and 2s; sort the array in ascending order.

Example 1:

Output Window

Compilation Results Custom Input

Problem Solved Successfully

Test Cases Passed: 35 / 35	Total Points Scored: 2 / 2
-------------------------------	-------------------------------

Your Total Score: Total Time Taken:

```
1 // Driver Code Ends
2 class Solution
3 {
4     public:
5     void sort012(int a[], int n)
6     {
7         int c0=0,c1=0,c2=0;
8         for(int i=0;i<n;i++){
9             if(a[i]==0)c0++;
10            else if(a[i]==1)c1++;
11            else c2++;
12        }
13        int k=0;
14        for(int i=0;i<c0;i++)a[k++]=0;
15        for(int i=0;i<c1;i++)a[k++]=1;
16        for(int i=0;i<c2;i++)a[k++]=2;
17    }
18 };
19 // Driver Code Ends
```

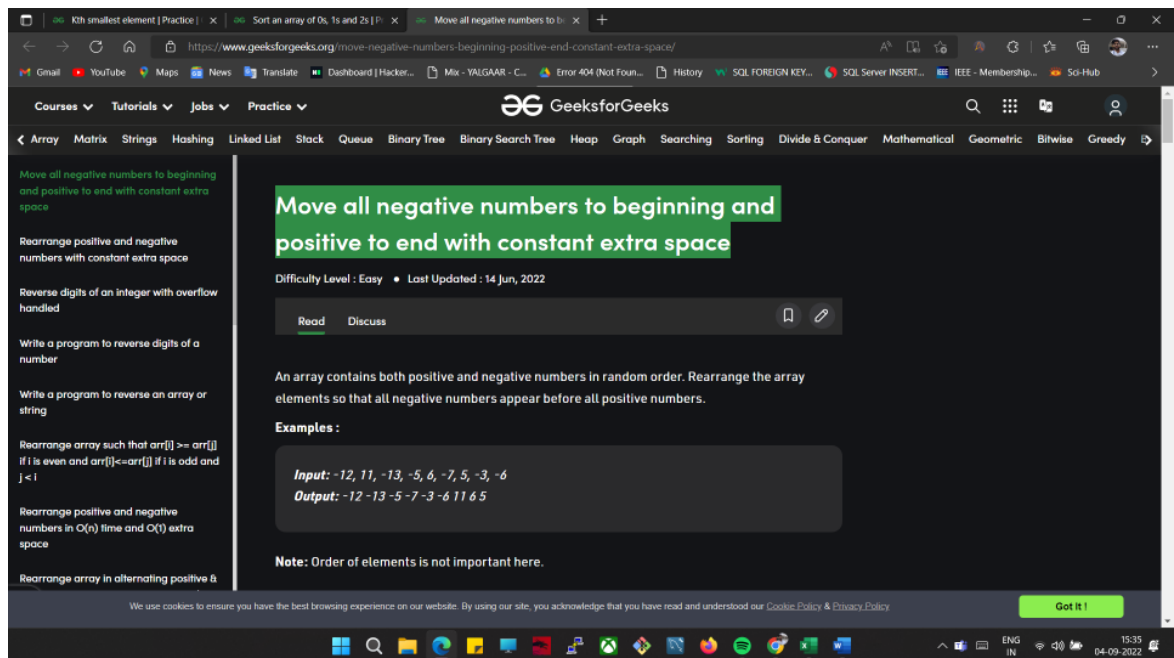
Compile & Run Submit

15:31 04-09-2022

```
Solution :- {
    public:
    void sort012(int a[], int n)
    {
        int c0=0,c1=0,c2=0;
        for(int i=0;i<n;i++){
            if(a[i]==0)c0++;
            else if(a[i]==1)c1++;
            else c2++;
        }
        int k=0;
        for(int i=0;i<c0;i++)a[k++]=0;
        for(int i=0;i<c1;i++)a[k++]=1;
        for(int i=0;i<c2;i++)a[k++]=2;
    }
}
```

};

6. Move all negative numbers to beginning and positive to end with constant extra space



Solution :- //Move all negative numbers to beginning and positive to end with constant extra space

```
#include<bits/stdc++.h>
```

```
using namespace std;
```

```
int main(){
```

```
    int N;
```

```
    cin >> N;
```

```
    vector<int>v(N);
```

```

    for(int i=0;i<N;i++){
        cin >> v[i];
    }
    vector<int>pos;
    vector<int>neg;
    for(int i=0;i<N;i++){
        if(v[i]<0){
            neg.push_back(v[i]);
        }
        else{
            pos.push_back(v[i]);
        }
    }
    int p=pos.size();
    int n=neg.size();
    vector<int>v3(p+n);
    merge(neg.begin(),neg.end(),pos.begin(),pos.e
nd(),v3.begin());
    for(int i=0;i<p+n;i++){
        cout << v3[i] <<" ";
    }

}

```