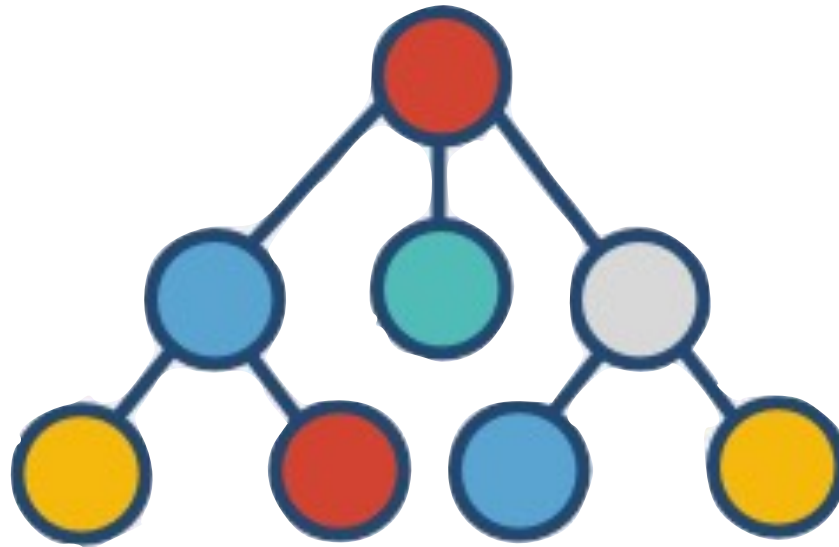


DATA STRUCTURE & ALGORITHMS



(By Prince Agarwal)
(“HELLO WORLD”)

INTERVIEW PREPARATION



Majority Element

[2, 3, 1, 3, 3]

N = 5

$N/2 = 5/2 = 2$

Ans = 3

[2, 3, 1, 3, 3, 4, 3, 1]

N = 8

$N/2 = 8/2 = 4$

Ans = -1

We want a Number which is greater than $N/2$

Hello world

INTERVIEW PREPARATION



Majority Element

[2, 3, 1, 3, 3, 4, 3, 1]

N = 8

Ans = -1

$N/2 = 8/2 = 4$

Naive Solution

[2, 3, 1, 3, 3, 4, 3, 1]



[2, 3, 1, 3, 3, 4, 3, 1]



$O(N^2)$

[2, 3, 1, 3, 3, 4, 3, 1]



[2, 3, 1, 3, 3, 4, 3, 1]



Hello world

▶

My Mac

Build cipher: Succeeded | 08/10/20 at 4:33 PM

+

↔

📄

📄

📄

<

>

📄 cipher

📁 cipher

📄 main.cpp

No Selection

☰

📄

```
8
9  #include <stdio.h>
10 #include <stdlib.h>
11 #include <string.h>
12
13 int majorityElement(int arr[], int n)
14 {
15     for(int i=0; i<n; i++){
16
17         // Take one Element from Array (say, element)
18         int ele = arr[i];
19         int count =0;
20
21         for(int j=0; j<n; j++){
22             if(arr[j] == ele)
23                 count++;
24         }
25
26         if(count<=(n/2))
27             return -1;
28         else
29             return arr[i];
30     }
31
32 }|
33
```

INTERVIEW PREPARATION

Majority Element by **Moore's Voting** Algorithm

[2, 3, 1, 3, 3, 4, 3, 1]

N = 8

Time Complexity: $O(N)$

Ans = -1

$N/2 = 8/2 = 4$

1st Step: We find the Candidate which is Majority In Array

2nd Step: We check Wether the Candidate is in Majority or not ?

Hello world

```
int majorityElement(int a[], int n)
{
    // 1st Step
    int count=1; int res=0;
    for(int i=1;i<n;i++){
        if(a[res] == a[i])
            count++;
        else
            count--;

        if(count==0){
            count=1;
            res=i;
        }
    }

    // 2nd Step
    count=0;
    for(int i=0;i<n;i++){
        if(a[res]==a[i])
            count++;
    }

    if(count<=(n/2))
        return -1;
    else
        return a[res];
}
```

INTERVIEW PREPARATION

Majority Element by Moore's Voting Algorithm

[2, 3, 1, 3, 3, 4, 3, 1]

N = 8

Time Complexity: O (N)

Ans = -1

$N/2 = 8/2 = 4$

1st Step: We find the Candidate which is Majority In Array

2nd Step: We check Whether the Candidate is in Majority or not ?

```
// 2nd Step
count=0;
for(int i=0;i<n;i++){
    if(a[res]==a[i])
        count++;
}

if(count<=(n/2))
    return -1;
else
    return a[res];
```

```
}
```

INTERVIEW PREPARATION

Majority Element by **Moore's Voting** Algorithm

[2, 3, 1, 3, 3, 4, 3, 1]

N = 8

Time Complexity: O (N)

Ans = -1

$N/2 = 8/2 = 4$

1st Step: We find the Candidate which is Majority In Array

```
int majorityElement(int a[], int n)
{
    // 1st Step
    int count=1; int res=0;
    for(int i=1;i<n;i++){
        if(a[res] == a[i])
            count++;
        else
            count--;

        if(count==0){
            count=1;
            res=i;
        }
    }
}
```


INTERVIEW PREPARATION

Majority Element by Moore's Voting Algorithm

[8, 8, 6, 6, 6, 4, 6]

N = 7

$N/2 = 7/2 = 3$

```
int majorityElement(int a[], int n)
{
    // 1st Step
    int count=1; int res=0;
    for(int i=1;i<n;i++){
        if(a[res] == a[i])
            count++;
        else
            count--;

        if(count==0){
            count=1;
            res=i;
        }
    }
}
```

Hello world

Majority Element by Moore's Voting Algorithm

INDEX [0, 1, 2, 3, 4, 5, 6]

[8, 8, 6, 6, 6, 4, 6]

N = 7

$N/2 = 7/2 = 3$

Initialise	Count = 1	Res = 0
i = 1	Count = 2	
i = 2	Count = 1	
i = 3	Count = 0	
	Count = 1	Res = 3
i = 4	Count = 2	
i = 5	Count = 1	
i = 6	Count = 2	

```
int majorityElement(int a[], int n)
{
    // 1st Step
    int count=1; int res=0;
    for(int i=1;i<n;i++){
        if(a[res] == a[i])
            count++;
        else
            count--;

        if(count==0){
            count=1;
            res=i;
        }
    }
}
```

Hello world

■ Majority Element by **Moore's Voting** Algorithm

Example 1: [8, 8, 6, 6, 6, 4, 6]



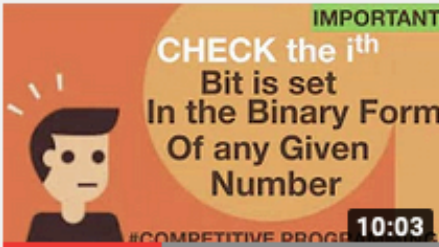




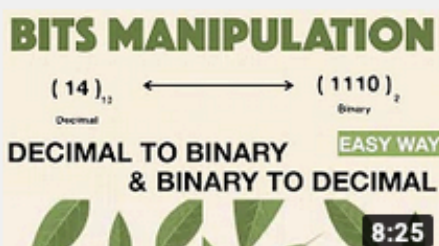
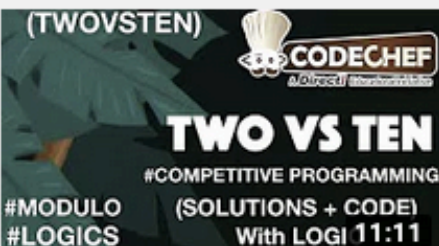








■ Majority Element by **Moore's Voting** Algorithm

Example 1: [8, 8, 6, 6, 6, 4, 6]

✓ ✓ ✗ ✗ ✓ ✗ ✓

Example 1: [6, 8, 7, 6, 6]

✓ ✗ ✓ ✗ ✓

 <p>CHECK the i^{th} Bit is set In the Binary Form Of any Given Number</p> <p>IMPORTANT</p> <p>#COMPETITIVE PROGRAMMING 10:03</p>	 <p>COUNT THE NUMBER OF ONE'S PRESENT IN BINARY NUMBER</p> <p>VERY EASY</p> <p>#COMPETITIVE PROGRAMMING 13:44</p>	 <p>CHECK GIVEN NUMBER IS POWER OF 2 ?</p> <p>(FULL EXPLANATION WITH CODE)</p> <p>#BITWISE #BINARY</p> <p>15:28</p>	 <p>LEFT SHIFT RIGHT SHIFT BITWISE OPERATOR</p> <p>EASY WAY</p> <p>(PART - 02)</p> <p>15:24</p>	 <p>AND NOT XOR OR BITWISE OPERATOR</p> <p>EASY WAY</p> <p>(PART - 01)</p> <p>13:06</p>
<p>Check the i^{th} bit is set, in the binary form of given numbe...</p> <p>1.1K views • 1 year ago</p>	<p>Count the number of one's in binary representation of...</p> <p>1.6K views • 1 year ago</p>	<p>Check a given number is power of 2 Bitwise operato...</p> <p>3.2K views • 1 year ago</p>	<p>Left shift and right shift bitwise operator ...</p> <p>1.4K views • 1 year ago</p>	<p>Bitwise Operators AND NOT OR XOR Competitiv...</p> <p>1.8K views • 1 year ago</p>
 <p>BITS MANIPULATION</p> <p>DECIMAL TO BINARY & BINARY TO DECIMAL</p> <p>EASY WAY</p> <p>8:25</p>	 <p>TWO VS TEN</p> <p>#COMPETITIVE PROGRAMMING</p> <p>#MODULO #LOGICS</p> <p>(SOLUTIONS + CODE) With LOGI 11:11</p>	 <p>CHEF AND HIS DAILY ROUTINE</p> <p>(SOLUTIONS + CODE) With LOGI 12:56</p>	 <p>EUCLIDEAN ALGORITHM</p> <p>FINDING GCD OF TWO NUMBERS</p> <p>#COMPETITIVE PROGRAMMING</p> <p>12:31</p>	 <p>SEIVE OF ERATOSTHENES</p> <p>PART - 02 (CODE)</p> <p>#COMPETITIVE PROGRAMMING 12:01</p>
<p>Bits Manipulation Decimal to Binary Binary to Decimal...</p> <p>1.5K views • 1 year ago</p>	<p>Program of Two vs Ten Codechef - TWOVSTEN ...</p> <p>1.3K views • 1 year ago</p>	<p>Program of chef and his daily routine - CHEFROUT ...</p> <p>1.7K views • 1 year ago</p>	<p>Euclidean algorithm for finding GCD of 2 numbers ...</p> <p>2K views • 1 year ago</p>	<p>Sieve of Eratosthenes -part 2 Competitive programming...</p> <p>2.2K views • 1 year ago</p>
 <p>SEIVE OF ERATOSTHENES</p> <p>PART - 01 (LOGIC)</p> <p>#COMPETITIVE PROGRAMMING 8:38</p>	 <p>CONCEPT OF PRIME NUMBERS</p> <p>#COMPETITIVE PROGRAMMING 13:38</p>	 <p>USE OF MEMSET()</p> <p>#COMPETITIVE PROGRAMMING 12:00</p>	 <p>FANCY QUOTES</p> <p>#COMPETITIVE PROGRAMMING (SOLUTIONS + CODE) With LOGI 15:46</p>	 <p>STUDYING ALPHABET</p> <p>#COMPETITIVE PROGRAMMING (SOLUTIONS + CODE) With LOGIC 24:28</p>
<p>Sieve of Eratosthenes -part 1 Competitive programming...</p> <p>3.4K views • 1 year ago</p>	<p>Program and concept of prime numbers. ...</p> <p>2.1K views • 1 year ago</p>	<p>memset() function in C/C++ and its syntax. Competitiv...</p> <p>4.3K views • 1 year ago</p>	<p>problem of Fancy Quotes getline() in strings --FANCY...</p> <p>2.1K views • 1 year ago</p>	<p>Concept of Handling the String related problems -...</p> <p>3.4K views • 1 year ago</p>

If You loved it : **SHARE IT**
Subscribe and Comment Below

<https://www.facebook.com/helloworldofficials>