Google
Amazon
ms (GS)
Directi
flipkart

Q, Given an array of size N. Return if there exists a

majority element.

↳ freq $> \frac{N}{2}$

Ex   A: 1,6,1,1,2,1

$N = 6$

freq $> \frac{6}{2}$

min freq = 4

A: 3,4,3,6,1,3, 2, 5, 3,3

$N = 10$

freq $> \frac{10}{2} > 5$

min freq = 6.

return False

```
bool    majority (int A[], int N)

    for(i=0; i<N; i++)
        // check freq of A[i]
        freq = 0
        for( j=0; i<N; j++)
        {
            if( arr[i] == Arr[i])     freq++
        }

        If ( freq > N/2 )
        {
            return True
        }

    return False
```

$TC: O(N^2)$

$SC: O(1)$

Q.   At max. How many M E can there be in an array?

$\xleftarrow{\hspace{1cm}} \frac{N}{2} \xrightarrow{\hspace{1cm}} \xleftarrow{} \frac{N}{2} \xrightarrow{\hspace{1cm}}$
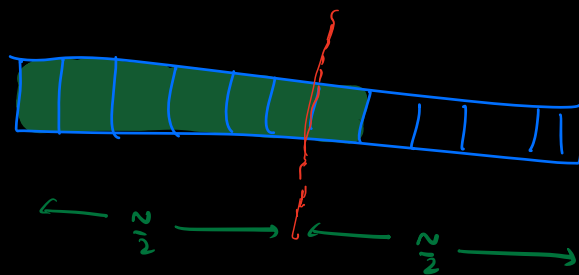
Only one M E can be there.

$$mE > \frac{11}{2}$$

$$mE > 5$$

min freq = 6

Bhavana 𝘹𝘹𝘹𝘹𝘹𝘹

Sai 𝘹

Shankar 𝘹𝘹𝘹

Nishchith 𝘹

Bhavana

mE
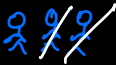
N	min Freq
11	6
9	5
7	4
5	3
3	2

Majority Freq.

6
↓
5

$\frac{N}{2}$

5
↓
4

N = 10

Obs: If 1 majority & 1 non-majority are deleted.
Then mE won't change.

obs: If 2 non-majority are deleted.
Then mE won't change.

A: 3 4 3 6 1 3 2 5 3 3 3

mE = 3̶ 4̶ 2̶ 3

Freq = 4̶ 0̶ 4̶ 0̶ 4̶ 0̶ 4̶ 0̶ 4̶ 2̶ 3

A: 4 3 4 3 4 2 4 2 4 4 3

mf   4̷ 4̷ 4̷ 4̷ 4
fre 1̷ 0̷ 1̷ 0̷ 1̷ 0̷ 1̷ 0̷ 1̷ 1̷ 1

A.   1 2 3

mf   1̷ 3
fre 1̷ 0̷ 1

d 2 3 4 5 6 7

mf :   1̷ 3̷ 5̷ 7
Frea : 1̷ 0̷ 1̷ 0̷ 1̷ 0̷ 1

fre(7) = 1̷ > 7/2

return False.

1 1 1 1 2 3 4 5 6 7 8 9

mf   1̷ 6̷ 8
fre 1̷ 2̷ 3̷ 4̷ 3̷ 2̷ 1̷ 2̷ 1̷ 2̷ 1̷ 0

Moore's voting algorithm.

TC: O(N)
SC: O(1)

```
int arr[]

int  mE = arr[0]
    freq = 1

for( J=1 ; i< N; j++)
{
        if ( arr[j] == mE]      freq++

        else
        {
                if( freq == 0 )
                {
                    mE = arr[i]
                    freq = 1
                }
                else
                {
                    freq--
                }
        }
}

count = 0

for( i=0 ; i < N; i++)
{
        if ( arr[i] == mE)      count++
}

if ( count > N/2 )      return True

else    return False.
```

TC: O(N)

SC: O(1)

Q.   Given an array. Q queries.

S, e, O => sum of all odd indexed elements
from s to e

S, e, e => sum of all even indexed elements
from s to e

$A = [\ 2,\ 3,\ 1,\ -1,\ 0,\ 8,\ 5,\ 4\ ]$

indices: 0 1 2 3 4 5 6 7

Q : 2

S	e	o/e	
3	6	O	$-1 + 8 = 7$
1	5	E	$1 + 0 = 1$

	0	1	2	3	4	5
A =	2	3	1	6	4	5
PS_E	2	2	3	3	7	7
PS_O	0	3	3	9	9	14

S, e, O

$PS_O[e] - PS_O[s-1]$
if $(s == 0)$
x

S, e, e

$PS_e[e] - PS_e[s-1]$
if $(s == 0)$
x

TC: $O(N + N + Q)$

↓ (↓

Create   Create $O(1)$ for
DFS   Pre   each query.

SC: $O(N)$

Doubts

3 3) 4 3 1 3 2 3 1 3

$mt = 8\ 3$

free = 1 2 1 0 1 0 1 0 1 0 1

$6 > \frac{11}{2}$

$6 > 5$

ans = 3.

$[m\ 6]_{free} > \frac{N}{2}$ $N = 10$

$(m\ 6)_{free} > \frac{10}{2}$

$(m\ t)_{free} > 5$

$(m\ t)_{free} \geq 6$

$\min\ Freq\ _{m\ 6} \geq 6$

$N \leq 10^5$

$N^2\ 10^{10}$ X

100 seconds

$N \leq 10^3$
            2
      N^2 10^6 ✓
$N \leq 10^4$

N^2 10^8 ✓

1 3 1 4 2 5 7 7

mt = ↓ 2 7
hu = ↓ ∅ ↓ ∅ ↓ ∅ ↓ 2