Bit Manipulation - 2

Q Given an integer array where every no. occurs twice except one number. find that unique no.

Soln → ans = ∀i, ^a[i]

```
ans = a[0]
for (i=1 to n-1) {
    ans ^= a[i]
}
return ans
```

TC : O(N)

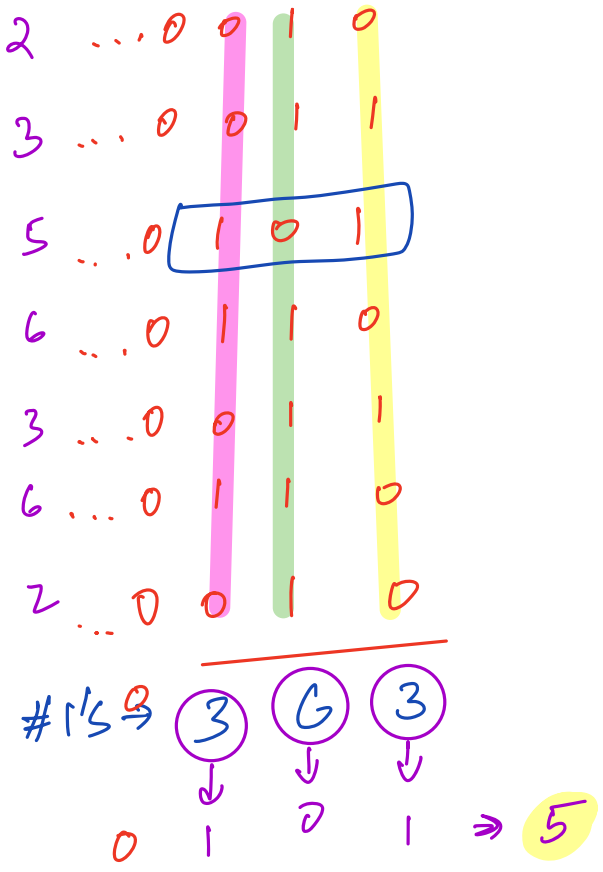
SC : O(1)

Quiz : $120 \wedge 5 \wedge 6 \wedge 6 \wedge 120 \wedge 5$

$(120 \wedge 120) \wedge (5 \wedge 5) \wedge (6 \wedge 6)$

$0 \wedge 0 \wedge 0 \Rightarrow 0$

A = [2 3 5 6 3 6 2]
0 1 2 3 4 5 6



1's is odd → 1
1's is even → 0

```
ans = 0 // 000 ..... 0
for(i=0 to 31){ // every bit
    cnt=0
    for(j=0 to n-1){
        if((A[j] & (1<<i)) > 0){
            cnt++
        }
    }
    if(cnt%2 == 1){
        // set ith bit in ans
        ans |= (1<<i)
    }
}
return ans.
```

TC = O(N)
SC = O(1)

Question 2 Given an integer array where every element occurs thrice except 1 element, find that unique element.

A = [4 5 5 4 1 6 6 4 5 6] (0 1 2 3 4 5 6 7 8 9) ans = 1

Brute force → for all i iterate & count frequency

TC = O(N²)

SC = O(1)

↓

use hashmap

TC = O(N)

SC = O(N)

A = [5 3 5 2 2 5 2]

5 ^ 3 ^ 5 ^ 2 ^ 2 ^ 5 ^ 2

= (5 ^ 5) ^ 5 ^ (2 ^ 2) ^ 2 ^ 3

= 0 ^ 5 ^ 2 ^ 3 = 5 ^ 2 ^ 3 = 4

5	1	0	1
3	0	1	1
5	1	0	1
2	0	1	0
2	0	1	0
5	1	0	1
2	0	1	0

cnt	
#1's is a multiple of 3 → 0	3K
else → 1	3K+1

#1's → (3) (4) (4)

↓ ↓ ↓

0 1 1

```
ans = 0 // 000 .... 0
for(i=0 to 31){ // every bit
    cnt=0
    for(j=0 to n-1){
        if((A[j]&(1<<i)) > 0){
            cnt++
        }
    }
    if(cnt%3==1){
        // set ith bit in ans
        ans |= (1<<i)
    }
}
return ans.
```

Q → Every element occurs K times except 1. find that unique element.

Sol 1 → above sol^n

```
if(cnt % K == 1)
    ans |= (1<<i)
}
```

Sol 2 → if (K is even) ⇒ **ans = fi ^ a[i]**

A = [4 4 4 4 2]

$(4 \wedge 4) \wedge (4 \wedge 4) \wedge 2$

$0 \wedge 0 \wedge 2 = 2$

Question 3

Given an integer array where every no. occurs twice except 2 numbers. find the 2 unique no.

A = [4 5 5 4 1 6 6 2]   ans = 1, 2

$4 \wedge 5 \wedge 5 \wedge 4 \wedge 1 \wedge 6 \wedge 6 \wedge 2 = 1 \wedge 2 = 3$

3 → 0 1 1 (bits 2 1 0)

10
01

  111110
^ 111101
  000011

> 0

$A \wedge A = 0$

if $(A != B) \rightarrow A \wedge B > 0$

↓

atleast 1 set bit

↓

it is set bit in exactly 1 of the 2 numbers.

Code

```
xor=0
for ( i=0 to n-1){
    xor ^= A[i]
}                          // xor = 3
b=-1 // optional
for ( i=0 to 31 ){
    if ( xor & (1<<i) >0 ){
        b=i
        break
    }
}                          // b=0
ans1=0, ans2=0
for(i=0 to n-1){
    if( (A[i] & (1<<b)) >0 )
        ans1 ^= A[i]
    else
        ans2 ^= A[i]
return{ans1, ans2}
```

A = [4 5 5 4 1 6 6 2]

0th bit = 1 → 5 5 1 → xor → 1

0th bit = 0 → 4 4 6 6 2 → xor → 2

A = [2 6 5 5]

$2^n b$ = 0 1 0 / 1 1 0 / 1 0 0 (2 1 0)

b = 2

2 6 5 5

2nd bit = 0 → 2

2nd bit = 1 → 6 5 5

2

6

2

TC = O(N) SC = O(1)

Question 4

Given an integer array, find max (A[i] & A[j])
s.t. i != j.

A = [5 4 6 8 5]
    0 1 2 3 4

ans = 5&5 = 5

5 0 1 0 1
4 0 1 0 0
6 0 1 1 0
8 1 0 0 0
5 0 1 0 1

5&6 = 4

A = [21 18 24 17 16]

21 → 1 0 1 0 1
18 → 1 0 0 1 0
24 → 1 1 0 0 0
17 → 1 0 0 0 1
16 → 1 0 0 0 0

ans = 21&17 = 17

24&16 = 1 0 0 0 0 = 16

A = [5 4 3 2 1]

5 → 1 0 1 } 5&4 = 4

4 → 1 0 0

3 → 0 1 1

2 → 0 1 0

1 → 0 0 1

Bruteforce → ∀i,j find a(i) & a(j) and take max value.

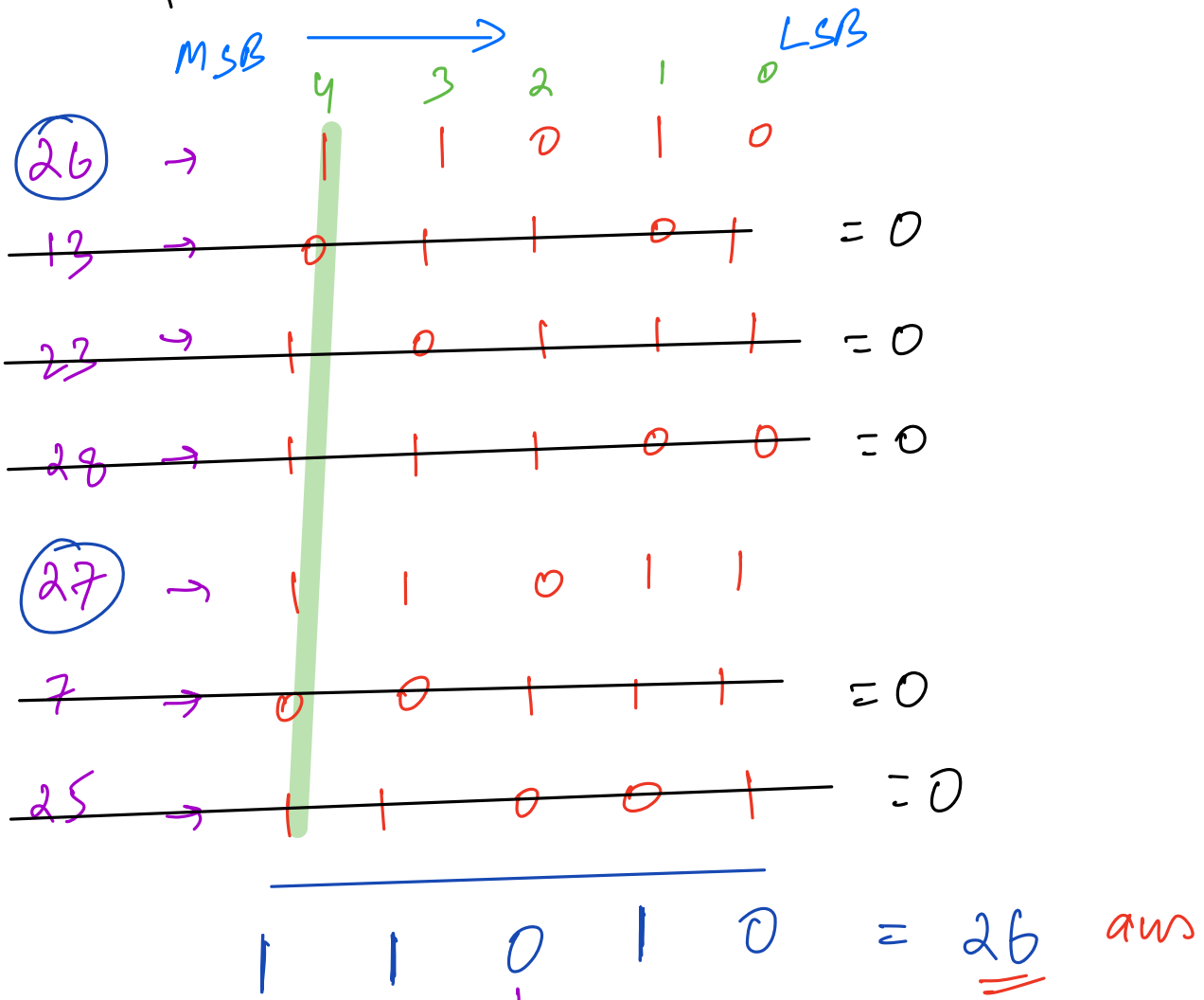
TC = O(N²) SC = O(1)

1 0 0 0 0 > 0 1 1 1 1

16 > 15

⇒ Goal is to make MSB = 1 (from left to right)

A: [26 13 23 28 27 7 25]
0 1 2 3 4 5 6



Code

```
ans=0
for( i=31 to 0 ){
    cnt=0
    for(j=0 to n-1){
        if( (A[j] & (1<<i)) > 0)
            cnt++
    }
```

```
if (cnt >=2) {
    ans |= (1<<i)
    for(j=0 to n-1) {
        if((A[j] & (1<<i)) == 0)
            A[j]=0
    }
}
}
return ans
```

TC = O(N)

SC = O(1)

to find pair (a[i], a[j]) → iterate & take non-zero pair.

A = [9 10 12]

9 → 1001

10 → 1010

12 → 1100

1000 = 8

9 & 10 = 8

9 & 12 = 8

10 & 12 = 8

} ans = 3 pairs

count pairs with max AND ?

ways to select 2 out of n elements = ${}^{n}C_2$

$= \frac{n(n-1)}{2}$

n : remaining non-zero elements