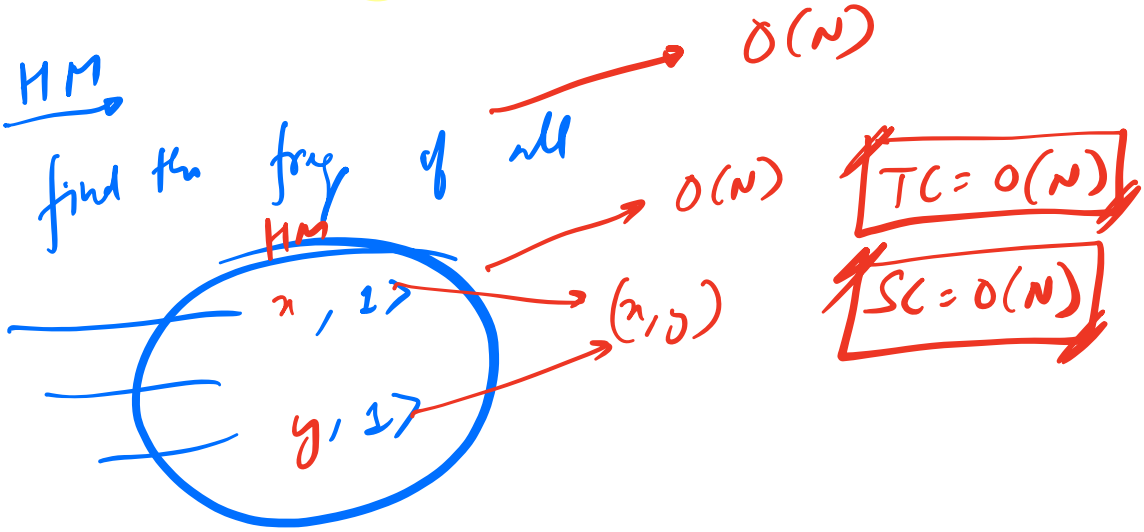
Q Given N elements.
Every element repeats twice except 2 elements, which occur exactly once!
find these 2 elements!

A: [3, 6, 4, 4, 3, 8] → ANS ⟨6, 8⟩

1) HM

find the freq of all

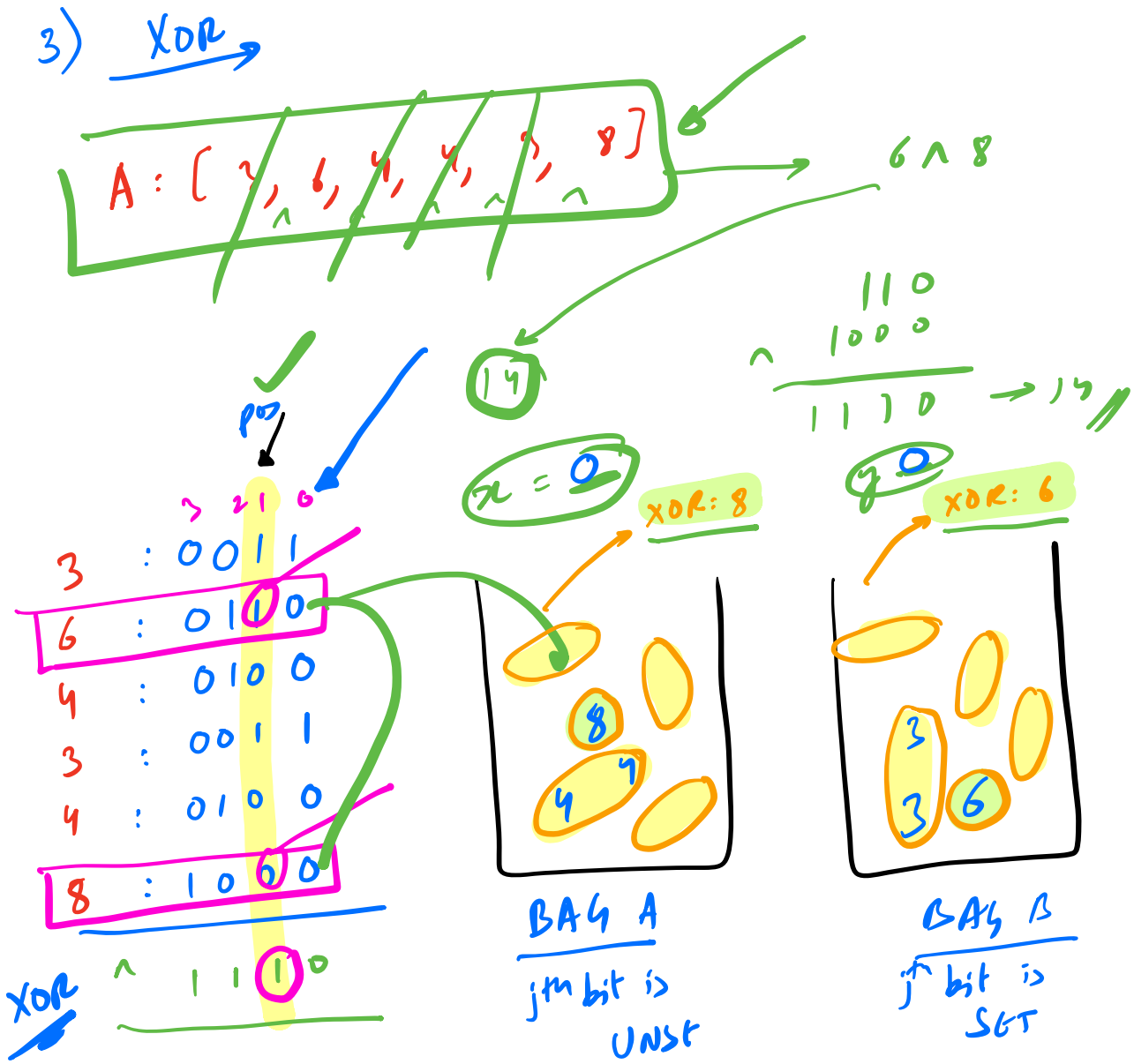


2) Sort

Sorted A: [3, 3, 4, 4, 6, 8]

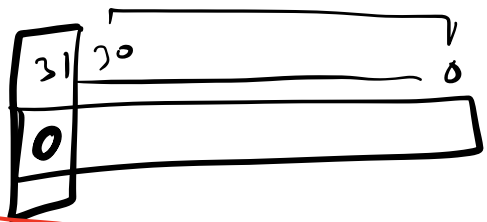
TC: O(N log N)

SC: O(1)


3) XOR
A : [3, 6, 4, 4, 3, 8]
6 ^ 8
1 1 0
^ 1 0 0 0
1 1 1 0 → 14
pos
14
x = 0
q 0
XOR: 8
XOR: 6
3 2 1 0
3 : 0 0 1 1
6 : 0 1 1 0
4 : 0 1 0 0
3 : 0 0 1 1
4 : 0 1 0 0
8 : 1 0 0 0
^ 1 1 1 0
XOR
BAG A
jth bit is UNSET
BAG B
jth bit is SET

```
XOR = 0;
for (i: 0 ⟶ N-1) {
    XOR = XOR ^ A[i];
}
pos = -1;
for (i: 0; i<31; i++) {
    if ((XOR & (1<<i)) > 0) {
        pos = i; break;
    }
}
x = 0, y = 0;
for (i: 0 ⟶ N-1) {
    if ((A[i] & (1<<pos)) > 0) {
        y = y ^ A[i];
    }
    else {
        x = x ^ A[i];
    }
}
print (x, y)
```

O(N)



O(1)

O(N)

TC = O(N)

SC = O(1)

Q Given an Array of size N.
Array contains all elements in [1, N+2]
except 2 elements. Find the 2 missing elements.

N = 4 [1, 4+2] → [1, 6)

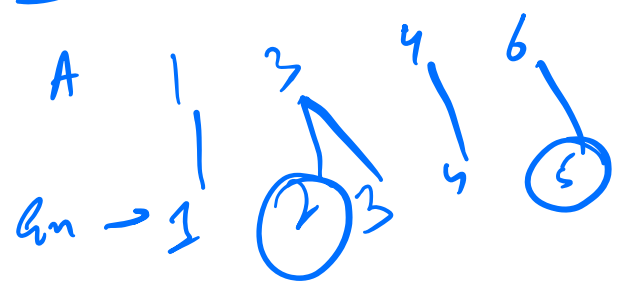
A : [3, 6, 1, 4] → (2,5)

1) HM

TC O(N)
SC O(N)

2) SORT

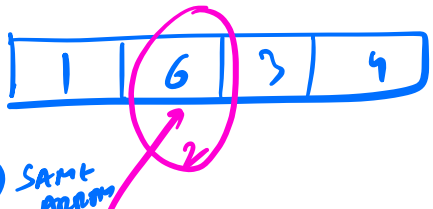
TC : O(NlgN)
SC : O(1)



3) SEND HOME TECHNIQUE

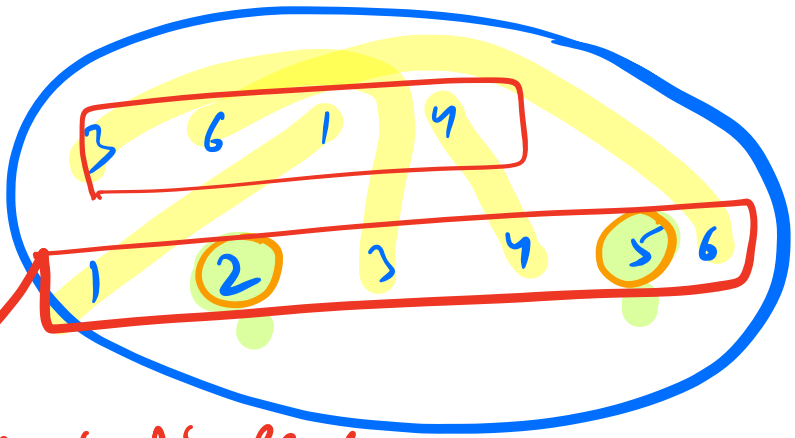
CAST WORK

TC : O(N)
SC : O(N) → O(1) SAME ARRAY



4) N = 4

A :



[1, N+2]

SAME AS PREV !!!

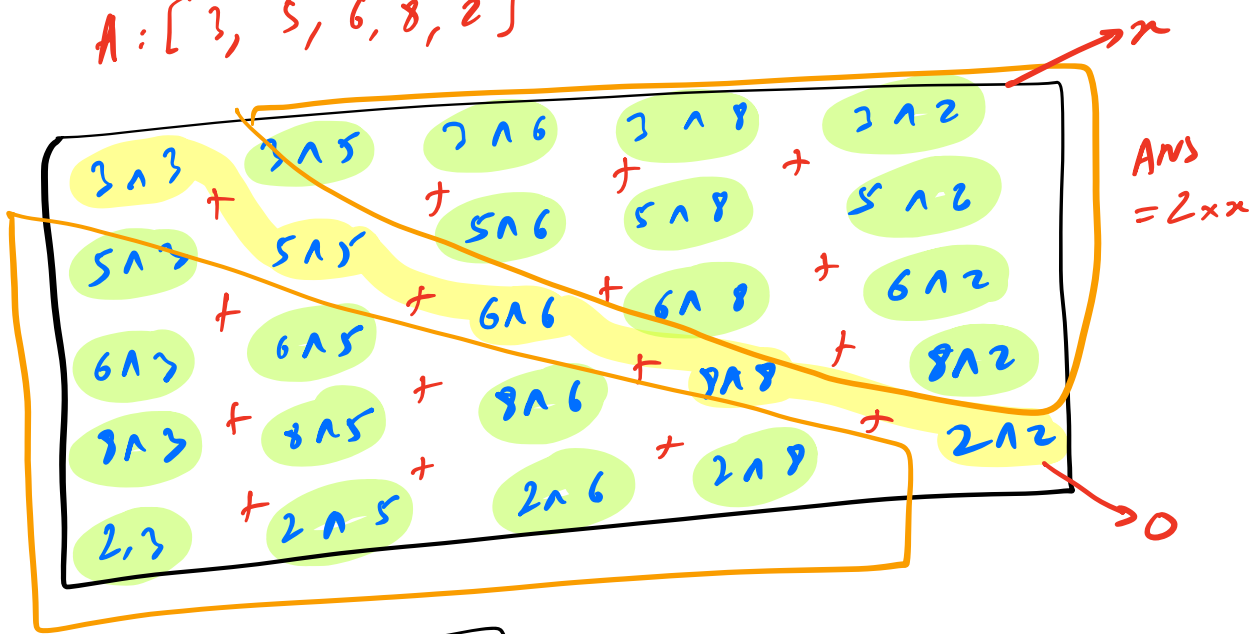
TC : O(N)

SC = O(1)

```
xor = 0
for (i: 0 ⟶ N-1) {
    xor = xor ^ A[i]
}
for (i: 1 ⟶ N+2) {
    xor = xor ^ i;
}
.
.
.
SAME ..
```

Q Given N elements.
Calculate the XOR sum of all pairs.

A : [3, 5, 6, 8, 2]



I) BF TC $O(N^2)$
SC $O(1)$

II)

3 ∧ 5 : 6 : 0 1 1 0

3 ∧ 6 : 5 : 0 1 0 1

3 ∧ 8 : 11 : 1 0 1 1

3 ∧ 2 : 1 : 0 0 0 1

5 ∧ 6 : 3 : 0 0 1 1

5 ∧ 8 : 13 : 1 1 0 1

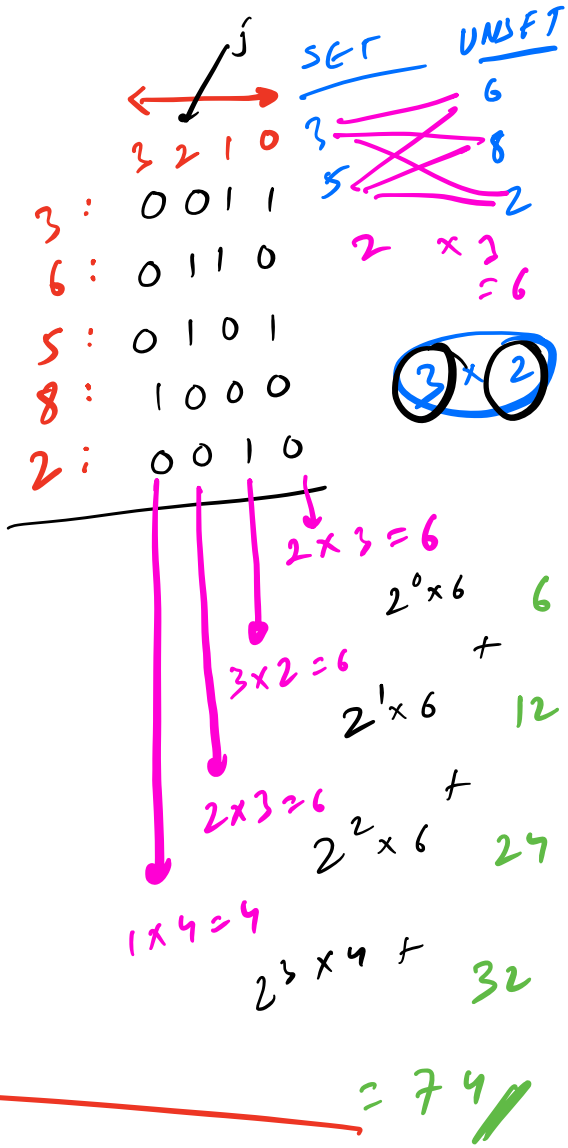
5 ∧ 2 : 7 : 0 1 1 1

6 ∧ 8 : 14 : 1 1 1 0

6 ∧ 2 : 4 : 0 1 0 0

8 ∧ 2 : 10 : 1 0 1 0

74



ANS : 148

x 2

= 74

m_bits → $\sim \log_2(MAX(A))$ → N : 3

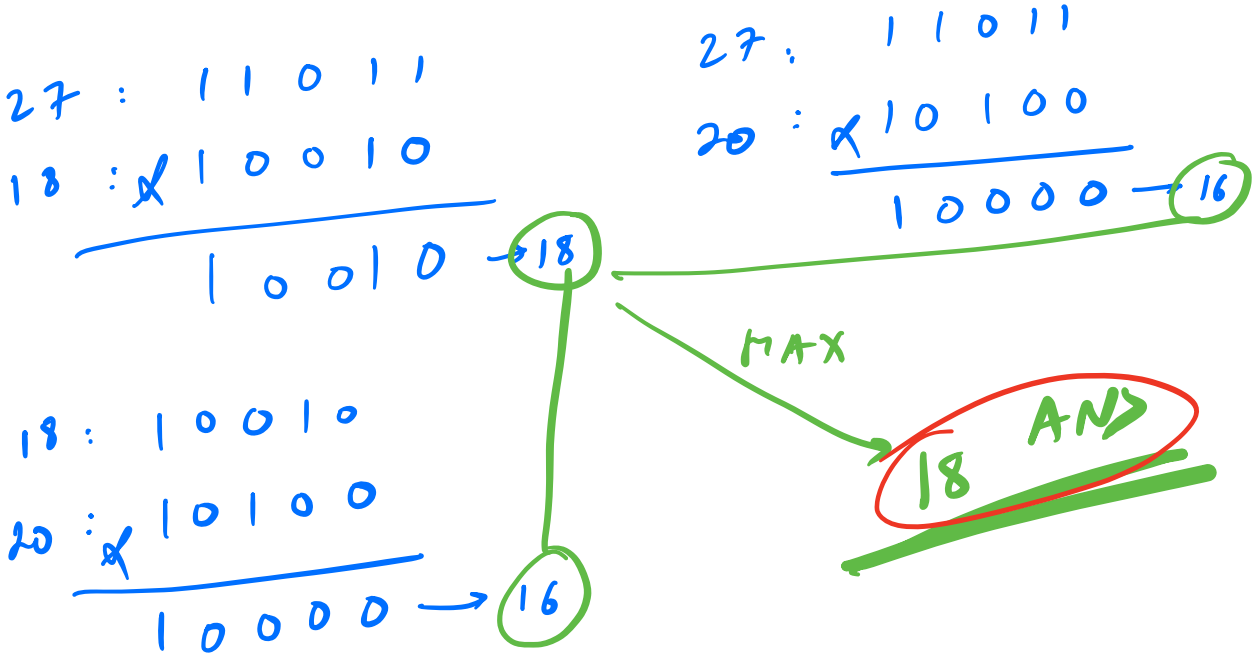
```
ANS = 0
for (j : 0 ⟶ m_bits) {
    cZero = 0, cOne = 0
    for (i = 0 ⟶ N-1) {
        if ((A[i] & (1<<j)) > 0) cOne++;
        else cZero++;
    }
    ANS += (1<<j) * cZero * cOne;
}
ret (ANS << 1);
```

$TC = O(\log(Max(A)) \times N)$

$SC = O(1)$

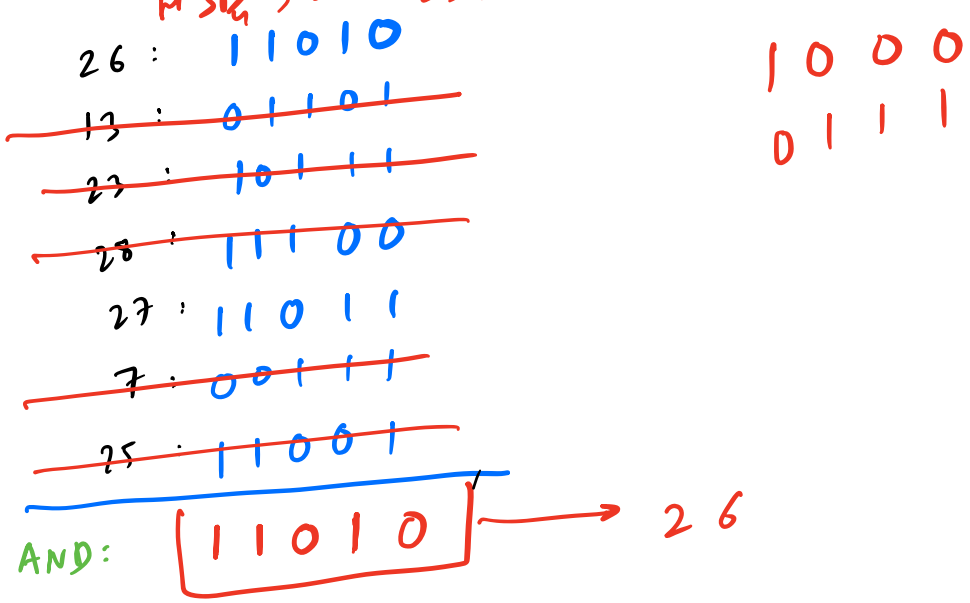
Q Given an array.
find the MAX value of (A[i] & A[j]) $\forall (i,j)$ $i \neq j$

A: [27, 18, 20]



A: [26, 13, 23, 28, 27, 7, 25]

MSB → LSB



1000
0111

1. find MAX(A)
2. max_bits = ~ log(max(A)) : 4

```
ANS = 0;
for(j = max_bits; j >= 0; j--) {
    cnt = 0;
    for(i: 0 ⟶ N-1) {
        if((A[i] & (1<<j)) > 0) {
            cnt++;
        }
    }
    if(cnt >= 2) {
        ANS = (ANS | (1<<j));
        for(i: 0 ⟶ N-1) {
            if((A[i] & (1<<j)) == 0) {
                A[i] = 0;
            }
        }
    }
}
out ANS;
```

TC = O(N · lg(MAX(A)))

SC = O(1)