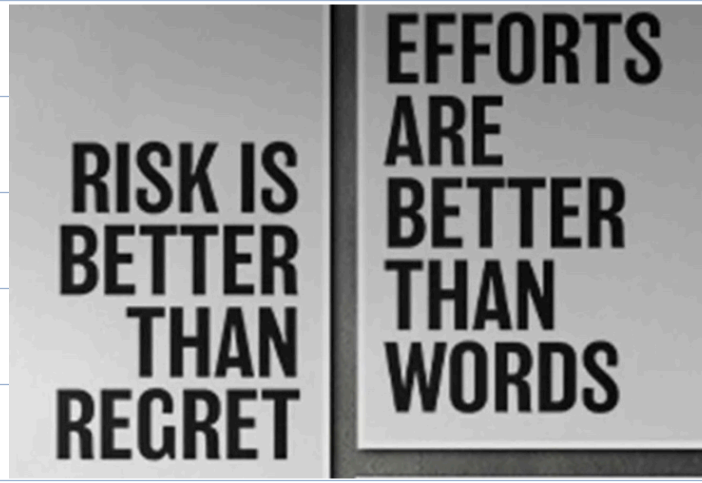
Agenda:

1. Truth Table
2. Basic AND XOR OR Properties
3. Left Shift & Right Shift
4. Power of left shift operator (set, flip, unset, check)
5. Count set bits
6. Single Number Problem



$$\overset{3}{1}\overset{2}{0}\overset{1}{1}\overset{0}{0}$$

$(1010)_2$

$2^3 + 2^1 = 8 + 2 = \underline{10}$

$$\overset{4}{1}\overset{3}{0}\overset{2}{1}\overset{1}{1}\overset{0}{0}$$

$(10110)_2$

$2^4 + 2^2 + 2^1 = 16 + 4 + 2 = \underline{22}$

Truth Table

		AND	OR	XOR
A	B	A & B	A | B	A ^ B
0	0	0	0	0
0	1	0	1	1
1	0	0	1	1
1	1	1	1	0

(Exclusive OR)
↓
same same puppy shame

(Addition without carry)

0 → False / Unset Bit

1 → True / Set Bit

NOT

A	~A
0	1
1	0

Properties of AND, OR & XOR

1) Even / odd

5 → 101　　　　6 → 110

9 → 1001　　　　10 → 1010

11 → 1011　　　　12 → 1100

A = 5 → 101
& 001
001 → 1

A = 6 → 110
& 001
000 → 0

A & 1 → 1 ⇒ A is odd
　　　→ 0 ⇒ A is even

2) $A \,\&\, 0 = \underline{0}$

$A = 5 \rightarrow 101$

$$\begin{array}{r} 101 \\ \&\ \underline{000} \\ \underline{000} \end{array} \rightarrow 0$$

3) $A \,\&\, A = \underline{A}$

$A = 5 \rightarrow 101$

$$\begin{array}{r} 101 \\ \&\ \underline{101} \\ \underline{101} \end{array} \rightarrow 5$$

4) $A \mid 0 = \underline{A}$

$A = 5 \rightarrow 101$

$$\begin{array}{r} 101 \\ \mid\ \underline{000} \\ \underline{101} \end{array} \rightarrow 5$$

5) $A \mid A = \underline{A}$

$A = 5 \rightarrow 101$

$$\begin{array}{r} 101 \\ \mid\ \underline{101} \\ \underline{101} \end{array} \rightarrow 5$$

6) $\boxed{A \wedge 0 = A}$

$A = 5 \rightarrow 101$

$$\begin{array}{r} 101 \\ \wedge\ \underline{000} \\ \underline{101} \end{array} \rightarrow 5$$

7) $\boxed{A \wedge A = 0}$

$A = 5 \rightarrow 101$

$$\begin{array}{r} 101 \\ \wedge\ \underline{101} \\ \underline{000} \end{array} \rightarrow 0$$

8) Commutative Property

$$\boxed{\begin{array}{l} A \,\&\, B = B \,\&\, A \\ A \mid B = B \mid A \\ A \wedge B = B \wedge A \end{array}}$$

9) Associative Property

$$\boxed{\begin{array}{lcl} (A \,\&\, B) \,\&\, C & = & A \,\&\, (B \,\&\, C) \\ (A \mid B) \mid C & = & A \mid (B \mid C) \\ (A \wedge B) \wedge C & = & A \wedge (B \wedge C) \end{array}}$$

Q → $a \wedge b \wedge a \wedge d \wedge b$

$= (a \wedge a) \wedge (b \wedge b) \wedge d$

$= 0 \wedge 0 \wedge d = \underline{\underline{d}}$

Q → $1 \wedge 3 \wedge 5 \wedge 3 \wedge 2 \wedge 1 \wedge 5$

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

$= 0 \wedge 0 \wedge 0 \wedge 2 = \underline{\underline{2}}$

v. imp.

Q → Given an integer array where every number occurs twice & only 1 number is unique, find that number.

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

(Ans) → 1

Sol 1 →

```
ans = A[0]
for i → 1 to (N-1) {
|     ans ^= A[i]
}
return ans
```

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

Sol 2 →

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

	2	1	0
2 →	0	1	0
3 →	0	1	1
5 →	1	0	1
6 →	1	1	0
3 →	0	1	1
6 →	1	1	0
2 →	0	1	0
# set bits →	3	6	3
	1	0	1

1 0 1 → 5 (Ans)

set bits → odd ⇒ 1

even ⇒ 0

Left Shift Operator <<

let say, we have 8 bit unsigned number.

discarded

	7	6	5	4	3	2	1	0		
A = 45	0	0	1	0	1	1	0	1	45	$*2$
A << 1	0	1	0	1	1	0	1	0	90	$*2$
A << 2	1	0	1	1	0	1	0	0	180	$*2$
A << 3	0	1	1	0	1	0	0	0	~~360~~ → 104	

data lose

$$\boxed{\begin{array}{l} A << 1 = A * 2 \\ A << n = A * 2^{n} \\ 1 << n = 2^{n} \end{array}}$$

Right Shift Operator $\gg$

	7	6	5	4	3	2	1	0	discarded
A = 45	0	0	1	0	1	1	0	1	45
$A \gg 1$	0	0	0	1	0	1	1	0	22
$A \gg 2$	0	0	0	0	1	0	1	1	11

45 /2 → 22 /2 → 11 → ⋮ → 0

$$\boxed{\begin{aligned} A \gg 1 &= A/2 \\ A \gg n &= A/2^n \end{aligned}}$$

Power of Left Shift

AND

		5	4	3	2	1	0
A = 45	→	1	0	1	1	0	1
1 << 3	→ &	0	0	1	0	0	0
		0	0	1	0	0	0

$001000 \rightarrow 2^3$

		5	4	3	2	1	0
A = 45	→	1	0	1	1	0	1
1 << 4	→ &	0	1	0	0	0	0
		0	0	0	0	0	0

$000000 \rightarrow 0$

A & (1 << i) → (1 << i) ⇒ i^{th} bit is set

→ 0 ⇒ i^{th} bit is unset

Check i^{th} bit

OR

$$\begin{array}{lcl} & & 5\ 4\ 3\ 2\ 1\ 0 \\ A = 45 & \rightarrow & 1\ 0\ 1\ 1\ 0\ 1 \\ 1 << 3 & \rightarrow & \mid\ 0\ 0\ 1\ 0\ 0\ 0 \\ \hline & & 1\ 0\ 1\ 1\ 0\ 1 \rightarrow \underline{45} \end{array}$$

$$\begin{array}{lcl} & & 5\ 4\ 3\ 2\ 1\ 0 \\ A = 45 & \rightarrow & 1\ 0\ 1\ 1\ 0\ 1 \\ 1 << 4 & \rightarrow & \mid\ 0\ 1\ 0\ 0\ 0\ 0 \\ \hline & & 1\ 1\ 1\ 1\ 0\ 1 \rightarrow \underline{61} \end{array}$$

$$\boxed{A \mid (1 << i) \rightarrow \text{set } i^{th} \text{ bit in } A}$$

XOR

$$\begin{array}{lcl} & & 5\ 4\ 3\ 2\ 1\ 0 \\ A = 45 & \rightarrow & 1\ 0\ 1\ 1\ 0\ 1 \\ 1 << 3 & \rightarrow & \wedge\ 0\ 0\ 1\ 0\ 0\ 0 \\ \hline & & 1\ 0\ 0\ 1\ 0\ 1 \rightarrow \underline{37} \end{array}$$

$$\begin{array}{lcl} & & 5\ 4\ 3\ 2\ 1\ 0 \\ A = 45 & \rightarrow & 1\ 0\ 1\ 1\ 0\ 1 \\ 1 << 4 & \rightarrow & \wedge\ 0\ 1\ 0\ 0\ 0\ 0 \\ \hline & & 1\ 1\ 1\ 1\ 0\ 1 \rightarrow \underline{61} \end{array}$$

$$\boxed{A \wedge (1 << i) \rightarrow \text{toggle / flip } i^{th} \text{ bit}}$$

Q → Unset i^{th} bit in given integer A.

$A = 45$ $\quad i = 3$ $\quad$ Ans $= \underline{37}$

$\quad i = 4$ $\quad$ Ans $= \underline{45}$

Sol → check + flip

```
if (A & (1<<i) > 0)
    A = A ^ (1<<i)
return A
```

$TC = \underline{O(1)}$ $SC = \underline{O(1)}$

Q → count the no. of set bits in given integer N.

$N = 45$ Ans $= \underline{4}$

$N = 5$ Ans $= \underline{2}$

int → 32 bits

```
cnt = 0
for i → 0 to 31 {
|    if ((N & (1 << i)) > 0) cnt++
}
return cnt
```

$TC = \underline{O(31)} \rightarrow \underline{O(1)}$ $SC = \underline{O(1)}$

Q → Given an integer array where every number occur twice except for exactly 2 numbers. Find those 2 numbers.

$$\begin{array}{ccccccccc} & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ A = [& 4 & 5 & 4 & \underline{1} & 6 & 6 & 5 & \underline{2} \ ] \end{array}$$

Ans = $\underline{1, 2}$

→ XOR of all elements → 1^2 $\quad$ x^y (unique no.)

→ $x \wedge x = 0$

⇒ $\boxed{x \wedge y > 0 \quad \because x \neq y}$

→ check which bit is set → let say K^{th} bit

⇒ K^{th} bit is set in either x or y but not both.

→ Divide the entire array into 2 sets → one where K^{th} bit is 1 & second where K^{th} bit is 0.

$$\begin{array}{ccccccccc} & 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ A = [& 4 & 5 & 4 & 1 & 6 & 6 & 5 & 2 \ ] \end{array}$$

$1 \wedge 2 = 3 \rightarrow 11$ (K = bit 0)

K^{th} bit 0 → { 4 4 6 6 2 } → xor of all elements

1 → { 5 1 5 }

```
all = A[0]
for i → 1 to (N-1) {
    all ^= A[i]
}

K = 0
for i → 0 to 31 {
    if ((all & (1<<i)) > 0) {
        K = i
        break
    }
}

x = 0    y = 0
for i → 0 to (N-1) {
    if ((A[i] & (1<<K)) > 0)    x ^= A[i]
    else                        y ^= A[i]
}

return x, y
```

$TC = O(N + 31 + N) = \underline{O(N)}$

$SC = \underline{O(1)}$
