

Today's Content :-

→ Number System basics

→ Binary to decimal

→ Add 2 binary no's.

→ Bitwise operators.

Quote :-

It's not the will to win that matters,
everyone has that.

It's the will to prepare to win that
matters.

Number System bases :-

↳ Decimal No. System $\begin{cases} \rightarrow [0-9] \\ \rightarrow [10] \end{cases}$

$$\begin{array}{ccc} 7 & 3 & 4 \\ \downarrow & \downarrow & \downarrow \\ 10^2 & 10^1 & 10^0 \end{array} \quad : \quad 700 + 30 + 4$$

$$\begin{array}{cccc} 6 & 5 & 9 & 4 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ 10^3 & 10^2 & 10^1 & 10^0 \end{array} \quad : \quad 6000 + 500 + 90 + 4$$

$$2 \ 4 \ 5 \quad : \quad 200 + 40 + 5$$

Other number systems :-

Binary, ternary,

hexa

Octal

Number System

$\begin{cases} [0-7] \\ [8] \end{cases}$

Octal

$$\begin{array}{ccc} & (1 \ 2 \ 5)_8 & \rightarrow 1*8^2 + 2*8^1 + 5*8^0 \\ & \swarrow \downarrow \downarrow & \\ 8^2 & 8^1 & 8^0 \end{array} \quad \Rightarrow \quad 85$$

Binary $\begin{cases} \rightarrow \text{every digit } [0-1] \\ \rightarrow \text{base } [2] \end{cases}$

$$\begin{array}{cccccc}
 & 4 & 3 & 2 & 1 & 0 \\
 (& 1 & 0 & 1 & 1 & 0)_2 & = 2^4 + 0 + 2^2 + 2^1 + 0 = \underline{\underline{22}}
 \end{array}$$

\downarrow 2^0
 \downarrow 2^1
 \downarrow 2^2
 \downarrow 2^3
 \downarrow 2^4

e.g.) $\begin{array}{cccccc} & 4 & 3 & 2 & 1 & 0 \\ (& 1 & 0 & 1 & 0 & 0)_2 \end{array} \rightarrow 2^4 + 2^2 = \underline{\underline{20}}$

\downarrow 2^4 \downarrow 2^2

$\begin{array}{ccc} & 2 & 1 & 0 \\ (& 1 & 2 & 0)_2 \end{array}$
→ invalid binary representation

$$\begin{array}{ccccccccc}
 & 7 & 6 & 5 & 4 & 3 & 2 & 1 & 0 \\
 & 1 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\
 \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\
 2^7 & 2^6 & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 &
 \end{array}$$

$\Rightarrow 2^7 + 2^6 + 2^5 + 2^1 + 2^0$
 $\approx 128 + 64 + 32 + 2 + 1$
 $\approx \underline{\underline{227}}$

Decimal to Binary.

$$\begin{array}{r|l} 2 & 37 \\ \hline 2 & 18 \\ \hline 2 & 9 \\ \hline 2 & 4 \\ \hline 2 & 2 \\ \hline 2 & 1 \\ \hline 0 & \end{array}$$

$$\begin{array}{cccccc} & 5 & 4 & 3 & 2 & 1 & 0 \\ \rightarrow & 1 & 0 & 0 & 1 & 0 & 1 \\ & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow & \downarrow \\ & 2^5 & 2^4 & 2^3 & 2^2 & 2^1 & 2^0 \end{array}$$

$$\Rightarrow 2^5 + 2^2 + 2^0 \\ \Rightarrow \underline{37}_2$$

$$\begin{array}{r|l} 2 & 25 \\ \hline 2 & 12 \\ \hline 2 & 6 \\ \hline 2 & 3 \\ \hline 2 & 1 \\ \hline 0 & \end{array}$$

$$(11001)_2$$

$$\begin{array}{r|l} 2 & 19 \\ \hline 2 & 9 \\ \hline 2 & 4 \\ \hline 2 & 2 \\ \hline 2 & 1 \\ \hline 0 & \end{array} \rightarrow (10011)_2$$

$$(0 \overset{4}{2} \overset{3}{1} \overset{2}{0} \overset{1}{1})_3 \rightarrow$$

↓

$$0 \times 3^4 + 2 \times 3^3 + 1 \times 3^2 + 0 \times 3^1 + 1 \times 3^0$$

$$\approx 0 + 54 + 9 + 0 + 1 \Rightarrow \underline{64}$$

$$(1 \overset{2}{3} \overset{1}{1})_8$$

$$\rightarrow 1 \times 8^2 + 3 \times 8^1 + 1 \times 8^0$$

$$\Rightarrow 64 + 24 + 1$$

$$\underline{89}$$

$$(9 \overset{1}{A})_{16}$$

$$\begin{matrix} \downarrow & \downarrow \\ 16^1 & 16^0 \end{matrix}$$

→

$$9 \times 16^1 + \overset{10}{A} \times 16^0$$

→

$$16 \times 9 + 10 \times 1$$

$$\Rightarrow \underline{154}$$

Add 2 Decimal numbers :-

$$\begin{array}{r} \overset{1}{7} \overset{1}{8} 9 \\ 0 \ 1 \ 4 \ 2 \\ \hline 0 \ 9 \ 3 \ 1 \end{array}$$

$$\begin{array}{r} \overset{12/10}{7} \overset{8/10}{8} \overset{12/10}{3} \ 9 \\ \overset{11/10}{3} \ 9 \ 4 \ 8 \\ \hline \overset{11/10}{1} \ \overset{12/10}{1} \ 7 \ 8 \ 7 \end{array}$$

$$d = 8/10,$$

$$c = 5/10$$

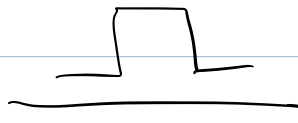
Add 2 Binary no., $d = 8/2$

$c = 8/2$

$$\begin{array}{r} \begin{array}{cccc} 1/2 & 3/2 & 2/2 & 1/2 \\ 0 & 1 & 1 & 0 \end{array} \\ \begin{array}{cccccc} 1 & 0 & 1 & 1 & 0 & \end{array} \\ \hline \begin{array}{cccccc} 1/2 & 0 & 0 & 1 & 1 & 1 \end{array} \\ \hline \begin{array}{cccccc} 1/2 & 1/2 & 3/2 & 2/2 & 1/2 & \end{array} \\ 0 \quad \begin{array}{cccccc} 1 & 1 & 1 & 0 & 1 & \end{array} \\ \hline \end{array}$$

$$\begin{array}{r} \begin{array}{cccc} 2/2 & 1/2 & 2/2 & 1/2 \\ 1 & 0 & 1 & 0 \end{array} \\ \begin{array}{cccccc} 3/2 & 1 & 1 & 0 & 1 & 1 \end{array} \\ \hline \begin{array}{cccccc} 1 & 1 & 0 & 1 & 0 & 1 \end{array} \\ \hline \end{array}$$

\therefore why binary :-

\rightarrow $> \sim : 1$ 
 $< \sim : 0$

int x = 25

\hookrightarrow machine stores it in binary.

$$x = 11\underline{001}$$

Print(x)

(Bitwise operators)

(& , | , ^ , ~ , << , >>)
 And or xor inverse left shift right shift

Truth table :-

a	b	$a \& b$	$a b$	$a \wedge b$	$\sim a$	$\sim b$
0	0	0	0	0	1	1
0	1	0	1	1	1	0
1	0	0	1	1	0	1
1	1	1	1	0	0	0

xor :-

Addition without carry

$$\begin{array}{r} 0 \\ 0 \\ \hline 0 \end{array}$$

$$\begin{array}{r} 0 \\ 1 \\ \hline 1 \end{array}$$

$$\begin{array}{r} 1 \\ 0 \\ \hline 1 \end{array}$$

$$\begin{array}{r} 1 \\ 1 \\ \hline 2/2 \\ 2/2 \\ \hline 0 \end{array}$$

don't consider carry

Some basic problems on Bitwise

$$a = 29, \quad b = 19$$

a : 1 1 1 0 1

b : 1 0 0 1 1

1 0 0 0 1

→ 17

print(a & b)

print(a | b)

1 1 1 1 1

→ 31

print(a ^ b)

0 1 1 1 0

→ 14

a : 1 1 0 1

b : 1 0 1 0

a & b

1 0 0 0

a | b

1 1 1 1

a ^ b

0 1 1 1

Properties :-

$$\begin{array}{r} a = 10 : \quad 1 \ 0 \ 1 \ 0 \\ \& \ 1 : \quad 0 \ 0 \ 0 \ 1 \\ \hline \text{print}(a \& 1) \quad 0 \ 0 \ 0 \ 0 \\ \hline \downarrow \\ 0 \end{array}$$

$$\begin{array}{r} a = 11 : \quad 1 \ 0 \ 1 \ 1 \\ \quad 0 \ 0 \ 0 \ 1 \\ \hline \text{print}(a \& 1) \quad 0 \ 0 \ 0 \ 1 \\ \hline \quad 1 \end{array}$$

$$\begin{array}{r} a = 14 : \quad 1 \ 1 \ 1 \ 0 \\ \quad 0 \ 0 \ 0 \ 1 \\ \hline \text{print}(a \& 1) \quad 0 \ 0 \ 0 \ 0 \\ \hline \quad 0 \end{array}$$

$$\begin{array}{r} a = 13 : \quad 1 \ 1 \ 0 \ 1 \\ \quad 0 \ 0 \ 0 \ 1 \\ \hline \text{print}(a \& 1); \quad 0 \ 0 \ 0 \ 1 \\ \hline \quad 1 \end{array}$$

// Observation

$\text{print}(10 \& 1) = 0$

$\text{print}(14 \& 1) = 0$

$\text{print}(11 \& 1) = 1$

$\text{print}(13 \& 1) = 1$

$\text{if}(a \& 1 == 0) \{$

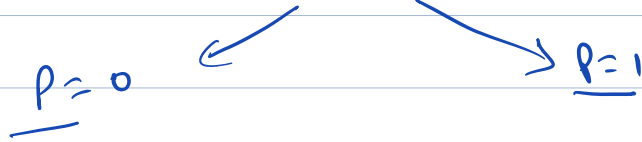
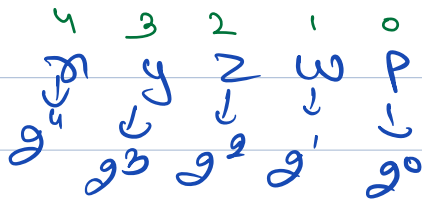
 a is even

$\}$

$\text{if}(a \& 1 == 1) \{$

 a is odd

$\}$



$$(x \cdot 2^4 + y \cdot 2^3 + z \cdot 2^2 + w \cdot 2^1 + 0)$$

$$x \cdot 2^4 + y \cdot 2^3 + z \cdot 2^2 + w \cdot 2^1 + 1$$

Basic properties :-

$$a \& 1 \begin{cases} \text{even} : 0 \\ \text{odd} : 1 \end{cases}$$

$$a \& 0 = a$$

$$a \& 0 = 0$$

$$a \& a = a$$

$$\begin{array}{cccc} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 1 & 0 \end{array}$$

$$a \mid 1 = \text{True}$$

$$a \mid a = 0$$

$$a \mid 0 = a$$

$$a \mid 1 \begin{cases} \text{even} : a+1 \\ \text{odd} : a-1 \end{cases}$$

$$a \mid a = a$$

$$a = 10100 \rightarrow 20$$

$$\wedge 1 = 00001$$

$$a = 11001 \rightarrow 25$$

$$1 = 00001$$

$$\begin{array}{ccccc} 1 & 1 & 0 & 0 & 1 \\ \hline 1 & 1 & 0 & 0 & 0 \end{array} \rightarrow 4$$

$$\begin{array}{ccccc} 1 & 0 & 1 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 & 1 \end{array} \rightarrow 21$$

10:36 → 10:46 pm :-

$$\begin{array}{cccccc} \begin{array}{c} 1/2 \\ 0 \end{array} & \begin{array}{c} 0/2 \\ 0 \end{array} & & \begin{array}{c} 1/2 \\ 0 \end{array} & & \\ 1 & 1 & 0 & 0 & 0 & \rightarrow 24 \\ 0 & 0 & 0 & 0 & 1 & \rightarrow 1 \\ \hline 1/2 & 1/2 & 0/2 & 0/2 & 1/2 & \\ 1 & 1 & 0 & 0 & 1 & \end{array}$$

After learning Bit Basics,

Students and me:-



Bitwise properties :-

$$\begin{array}{r} 1101 \\ \& \underline{1001} \\ \hline \end{array}$$

$$a \& b = b \& a$$

$$a | b = b | a$$

$$a \wedge b = b \wedge a$$

$$(a \& b \& c) = (c \& b \& a)$$

$$(a \& (b \& c))$$

$$(b \& c) \& a$$

Commutative

Associative

$$(a \& b \& c) = (c \& b \& a) = (b \& a \& c)$$

$$(a | b | c) = (c | b | a) = (b | a | c)$$

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

Ques 5 numbers :-

$$a \wedge b \wedge a \wedge d \wedge b =$$

$$a \wedge a \wedge b \wedge b \wedge d \rightarrow \underline{d}$$

Ques) 7 numbers :-

$$e \wedge f \wedge a \wedge f \wedge e \wedge g \wedge a = g$$

Ques Given N array elements,
every element repeats twice
except 1, find unique element.

arr[5] = { 6, 9, 6, 10, 9 } \rightarrow 10

arr[7] = { 12, 9, 12, 8, 7, 9, 8 } \rightarrow 7

idea 1 :-

1) For every element, iterate on array
& get its frequency.

```
for (i=0; i<n; i++) {  
    for (j=-; j< ; j++) {
```

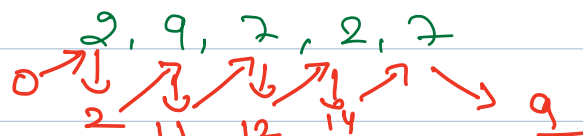
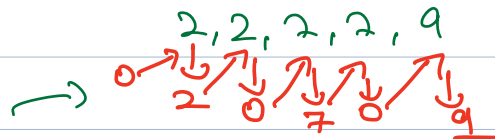
~~~~~

}

}

T.C  $\rightarrow O(N^2)$

S.C  $\rightarrow O(1)$



idea 2 :-

ans = 0

i = 0, i < N, i++ }

ans = ans ^ arr[i]

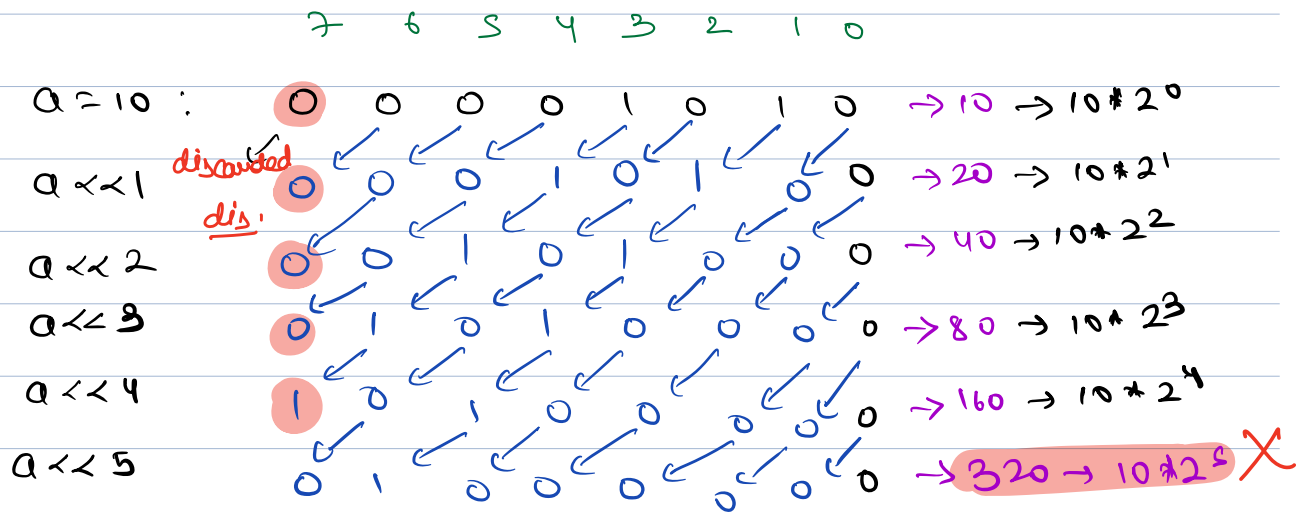
}

return ans :-

T.C  $\rightarrow O(N)$ , S.C  $\rightarrow O(1)$

// left shift :-

8 bit number



$$a \ll 1 \Rightarrow a * 2^1$$

$$a \ll 2 \Rightarrow a * 2^2$$

$$a \ll 3 \Rightarrow a * 2^3$$

$$a \ll n \Rightarrow a * 2^n$$

we are  
losing data

every time