Announcement: 23rd June → Bit manipulation, sorting, maths and modular

[0-9] Decimal num system

7 3 4 : Seven hundred thirty four : $700 + 30 + 4$: $7 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$

6 9 7 4 : $6 \times 10^3 + 9 \times 10^2 + 7 \times 10^1 + 4 \times 10^0$

Bits: 0 & 1s

AMA (Just send a whatsapp)

8510955377

Octal numbers [0-7] base is 8

$(331)_8 \longrightarrow$ 3 3 1 $= 3 \times 64 + 3 \times 8 + 1$

8^2 8^1 8^0

$= (217)_{10}$

$(971)_8$ ✗

Halloween = Christmas ?

OCT 31 DEC 25

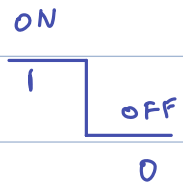
$(31)_8 = (25)_{10}$

Decimal

$(111)_8 \longrightarrow 64 + 8 + 1 = 73$

8^2 8^1 8^0

Binary [0 1] Base 2 [0, 1]

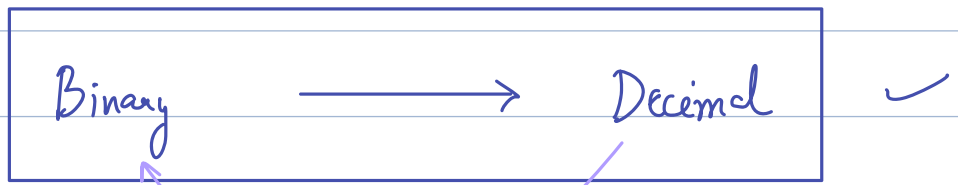


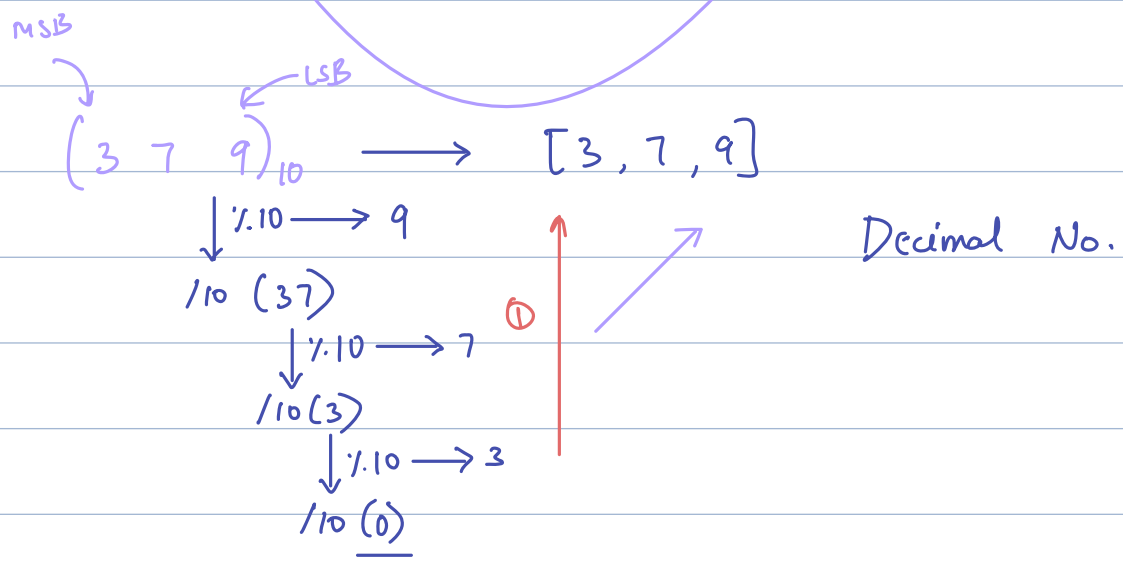
$(1\ 0\ 1\ 1\ 0\ 1\ 0)_2 = 2^6 + 2^4 + 2^3 + 2$

$= 64 + 16 + 8 + 2$

$= 90$

$(1\ 0\ 1\ 1)_2 = 8 + 2 + 1 = 11$





Decimal to binary

$(20)_{10} = (10100)_2$

$$2\overline{)1}\quad 0$$
$$\underline{0}$$
$$1$$

2	20	→ %2 = 0
2	10	→ %2 = 0
2	5	→ %2 = 1
2	2	→ %2 = 0
2	1	→ %2 = 1
	0	

$(45)_{10}$

2	45	%2 → 1
2	22	%2 → 0
2	11	%2 → 1
2	5	%2 → 1
2	2	%2 → 0
	1	→ %2 → 1

$(101101)_2$

$(25)_{10}$

2	25	%2 → 1
2	12	%2 → 0
2	6	%2 0
2	3	%2 1
	1	→ %2 1

$(11001)_2$

(Decimal to Binary)

+ve

Octal → Binary
↳ Decimal ↗

Add two Decimal numbers

Sum: 17 % 10 = 7

Carray: 17 / 10 = 1

```
  1  1  1
  3  4  9  7
+ 4  8  7  4
-------------
  8  3  7  1
-------------
```

Num / 10

```
Carry  1  1  1  1
       5  6  7  7
+      8  8  2  8
-------------------
Num    1 14 15 10 15
-------------------
Num%10 1  4  5  0  5
```

Num/2

```
Carry :  0  1  1  0
         1  0  1  1  0
         0  0  1  1  1
-----------------------
Num      1  1  3  2  1
-----------------------
Num%2    1  1  1  0  1
```

Num/2

```
Carry :  0  0  1  1
         1  0  0  1  1   → 19
         0  1  0  0  1   → 9
-----------------------
Num      1  1  1  2  2
-----------------------
Num%2    1  1  1  0  0   → 28
```

Addition of binary numbers

0 → False / Unset / off

1 → True / Set / on

Strict parents

Clean your room and complete HW

TOY

Lenient parents : Clean your room OR complete HW

XOR : Same same puppy shem

Truth table operators

Bit Manipulation (& | ^ ~ >> <<)

& → and, | → or, ^ → XOR, ~ → NOT, >> → Rightshift, << → leftshift

a	b	a & b	a | b	a ^ b	~a
0	0	0	0	0	1
0	1	0	1	1	1
1	0	0	1	1	0
1	1	1	1	0	0

XOR: Addition without carry

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

$a \,\&\, b = a \times b$ absolutely not

VI

print (5 & 6) ⟶ 4

5 → 101
6 → 110
100 ⟶ 4

print (20 & 45) ⟶ 4

20 → 010100
45 → 101101
000100 → 4

20 → 010100
xor 45 → 101101
111001 → 57

5 & 1 = 1

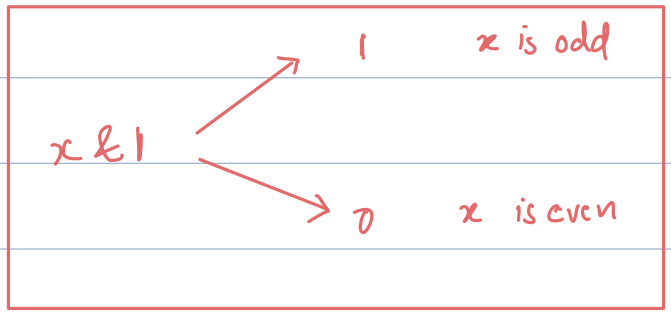
101
& 001
001

6 & 1 = 0

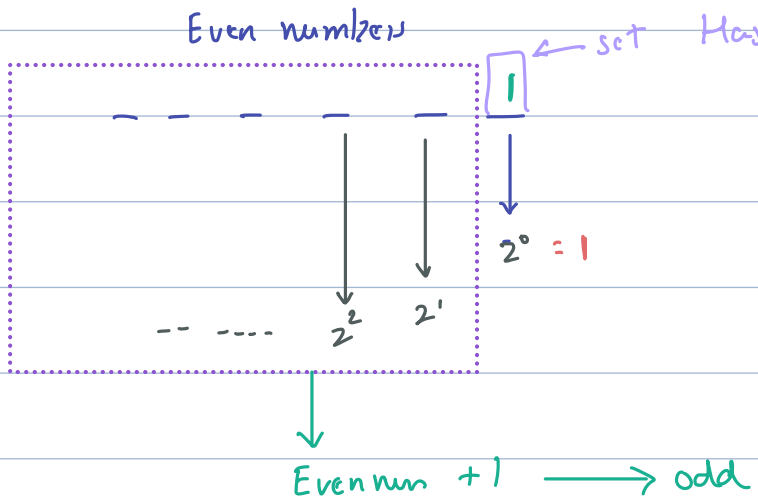
110
001
000

00001

Property 1)



why?



property 2:

A & 0	= 0
A & A	= A

S: 101
S: 101
101

S: 101
0: 000
101

property 3

A | 0	= A
A | A	= A

S: 101
or S: 101
101

property 4

$A \wedge A = 0$

$A \wedge 0 = A$

A: 101010
A: 101010
000000

$1 \wedge 0 = 1$

$0 \wedge 0 = 0$

A: 1001
0: 0000
1001

Optional HOTS:

$A \wedge 1 =$ even A+1, odd A-1

Toggles the 0^{th} bit

$A | 1 \longrightarrow$

5: 101
001
100 = 4

6: 110
1: 001
111 = 7

Break (10:36 - 10:46)

Commutative property :

$A + B = B + A$

$A = 5 \quad 101$

$B = 6 \quad \& \; 110$

100

$A \& B = B \& A$

$A | B = B | A$

$A \wedge B = B \wedge A$

110

$\& \; 101$

100

Associative property

$$(A + B) + C = (A + C) + B = A + (B + C)$$

$(A \& B) \& C = (A \& C) \& B = A \& (B \& C)$

$(A | B) | C = (A | C) | B = A | (B | C)$

$(A \wedge B) \wedge C = (A \wedge C) \wedge B = A \wedge (B \wedge C)$

$1 \& 3 \& 5 = (1 \& 3) \& 5 = (1 \& 5) \& 3$

$= (5 \& 3) \& 1$

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

↓

$a \wedge \cancel{b \wedge b \wedge c \wedge c \wedge d \wedge d} = a$

0 0 0

0

✓

$\cancel{1 \wedge 3 \wedge 3 \wedge 3 \wedge 2 \wedge 1} = 3 \wedge 2 = 1$

11
10
01

Q1) Given an array of size N. All elements are repeating even no. of times. Except ~~one~~ → 2 (Advance) element which is occurring odd no. of times. Find element occurring odd times.

A = { 2, 2, 3, 4, 3, 7, 3, 3, 7, 4, 6 }

output: 6

2 : 2

3 : 4

4 : 2

7 : 2

6 : 1

1) Count occurance of all elements :

$O(N^2)$

$A \wedge A = 0$

$A \wedge A \wedge A \wedge A = 0$

$A \wedge A \wedge A = A$

unique = 2 | 0 | 3 | 3∩4 | 4 | 4∩7 | 4∩7∩3 | 4∩7 | 4 | 0 | 6

{ 2 , 2 , 3 , 4 , 3 , 7 , 3 , 3 , 7 , 4 , 6 }

unique = 0

for (i=0; i<N; i++){

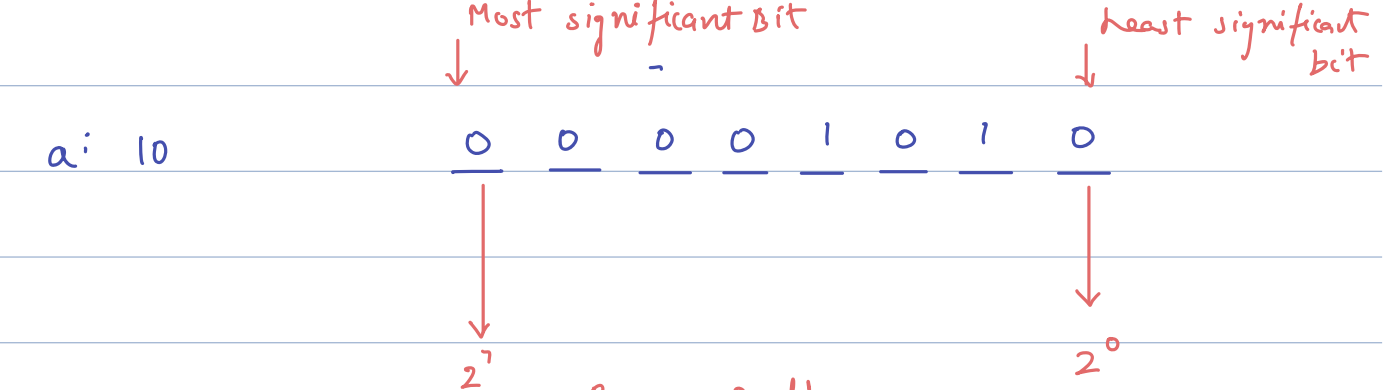
unique = unique ∩ A[i]

}

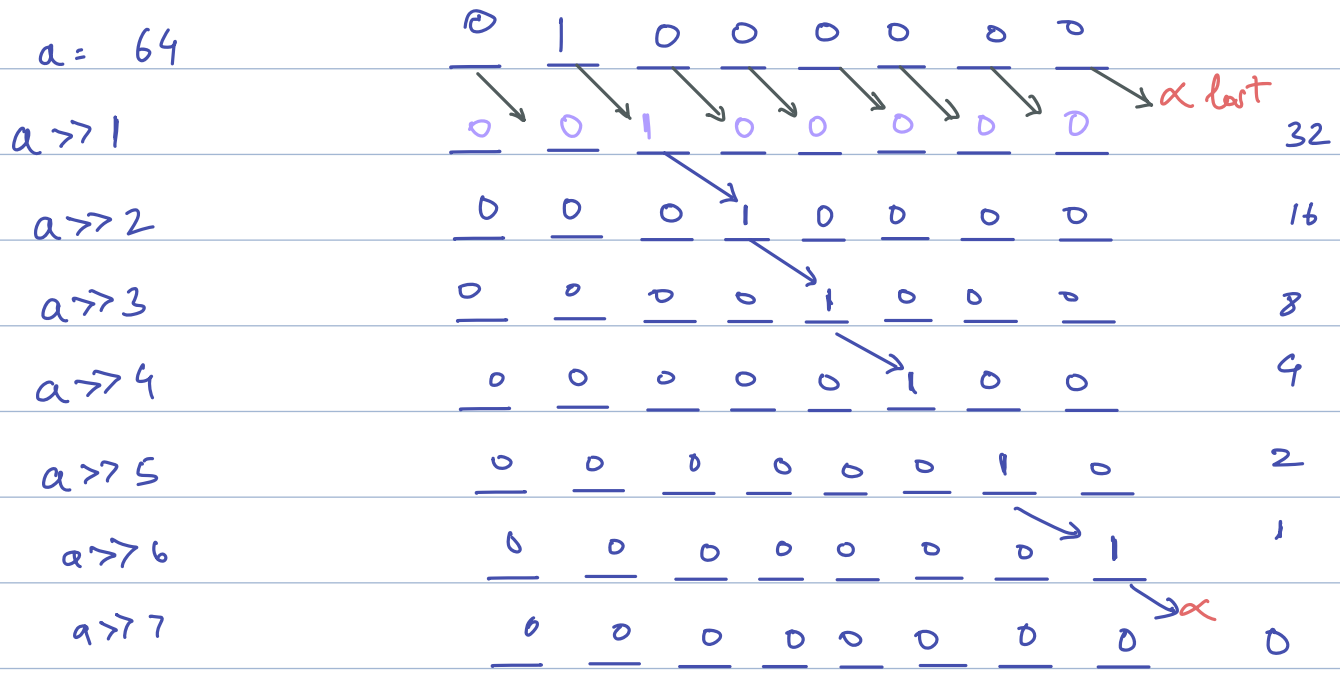
return unique

TC: O(N)

SC: O(1)



Right Shift



$a >> 1 = a/2$

$a >> 2 = a/4$

$a >> k \Rightarrow a/2^k$

** Left shift