Bit Manipulations

- Part 1

Apr 11, 2022

AGENDA

- Number systems.
  - ↘ (Binary number system)
- Operations on Binary nos.
- Some interesting problems.

Roman.

CDXXVI + CXLIV =

Number systems.

406 + 164 = 570.

Expand 1432 !

* 1000 + 400 + 30 + 2.

$1 \times 10^3 + 4 \times 10^2 + 3 \times 10^1 + 2 \times 10^0$

face value. ← 1243

1000 200 40 3.

1000 → Place value.

1 2 3 4 5 6 7 8 9 0.

1 → face value.

512. Place value = 10.

132 → Place value = 100.

1 2 3 4 5 6 7 8 9 Ø.

↓

543. = 500+40+3.

(10) → Base of the number system.

↓ Decimal

↓ Deci (10)

$(ABC)_{26} \longrightarrow (\quad)_{10}$?

A → .

8 symbols.

0 1 2 3 4 5 6 7

↓

$(324)_8$ =

↓

Expand $(324)_8$

$(328)_{10} = 3\times10^2 + 2\times10^1 + 8\times10^0$

$(324)_8 = 3\times8^2 + 2\times8^1 + 4\times8^0$

= ↓

$(\quad)_{10}$

$(924)_8 = ?$ 9 is not a valid symbol in Base-8.

$(A2)_{10}$ ✗ A is not a valid symbol in decimal.

$(1307)_{25} = 1\times25^3 + 3\times25^2 + 0\times25^1 + 7\times25^0$

13x - -

1307

1307

* Each digit should have its own place value.

Quiz

$(02101)_3 = 0\times3^4 + 2\times3^3 + 1\times3^2 + 1\times3^0$

$= 54 + 9 + 1$

$= 64.$

Computers.

Uses Binary no. system.

V = 5V → 5

V = 4V → 4

V = 3V → 3.

High. ~5V.
or Low. ~ 0.5V.

Computer uses a number system with base 2.

0 and 1
false True
Low High.
Off On

$(10111)_2 = 1\times2^4 + 0\times2^3 + 1\times2^2 + 1\times2^1 + 1\times2^0$

$(43)_{10} = (\ldots gfedcba)_2$

$\Rightarrow 43 = a\times2^0 + b\times2^1 + c\times2^2 + d\times2^3 + \ldots$

a = 1 or 0

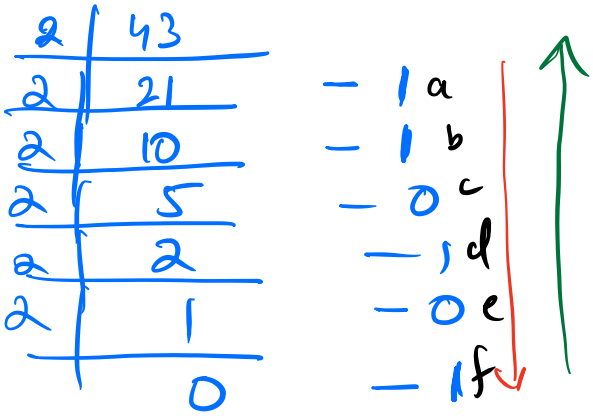
Find the values of a, b, c, d...

$b\times2^1 + c\times2^2 + d\times2^3 + \ldots$: Multiple of 2

$43 \% 2 = a$

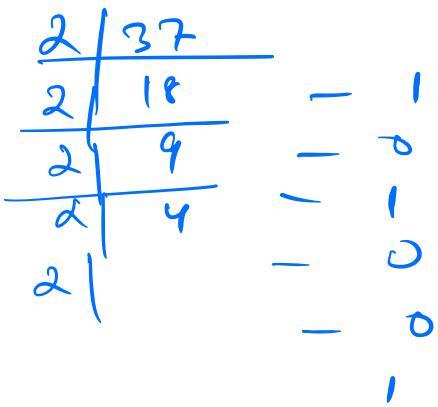
$43//2 = b\times2^0 + c\times2^1 + d\times2^2 + \ldots$

$(43//2) \% 2 = b$



$(43)_{10}$

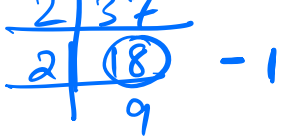
$= (101011)_2$



100101

Q. Given a decimal no., return the binary representation.

```
def binarise(n):
    arr = []
    while(n > 0):
        r = n % 2
        arr.append(r)
        n = n // 2
    arr = reversed(arr)
    return arr
```



Popular Number systems:

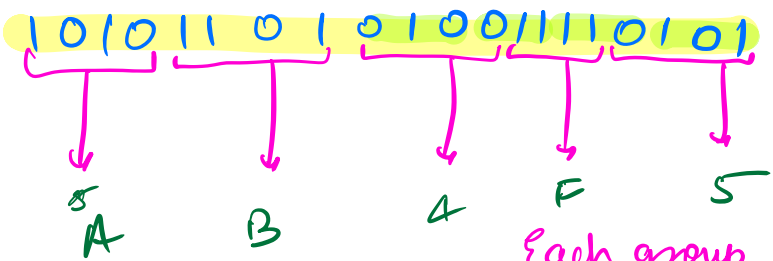
1. Decimal → Base 10. Humans!
2. Binary → Base 2 Computers!
3. Octal. → Base 8 (0, 1, 2, 3 ... 7)
4. Hexadecimal → Base 16

A B C D E F

0-9, 10, 11, 12, 13, 14, 15.

131 → (13)1
131 ✗

$(DAD)_{16} \rightarrow (\quad)_{10}$



Each group can be converted to Hexadecimal digit.

0-15

1111 → 15
0000 → 0

+AB4F5

Hexadecimal.

Colors

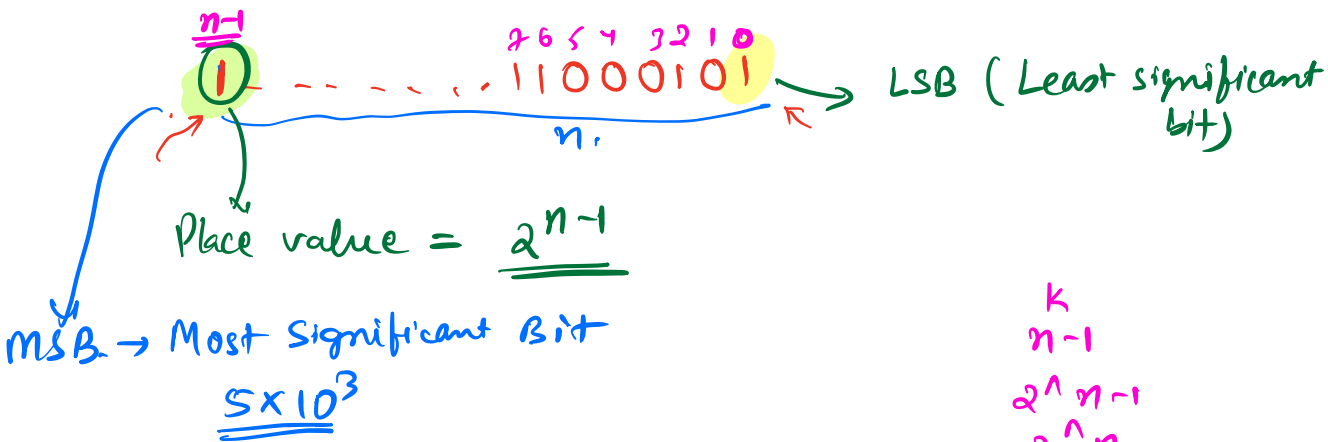
IP addresses.

#000000
#43fa2c

R G B

(0-255)

Properties of Binary numbers.



k
n-1
2^n-1
2^n
N
2^{k-1}

Q. A N-digit Binary no whose MSB is 1. Rest all are 0.

Decimal representation = ? 2^{N-1}

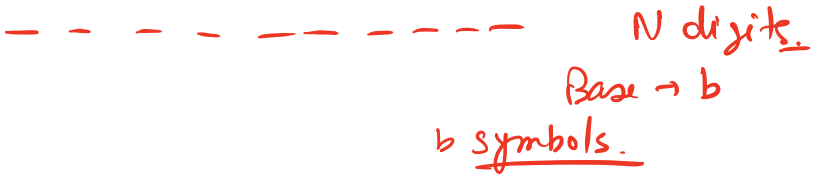
n-1
1 0 0 0 0 0 0

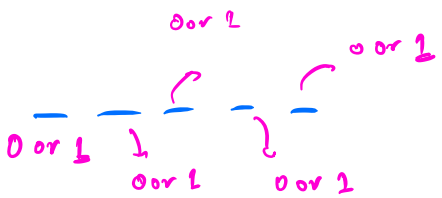
$1 \times 2^{n-1}$

Q. N digit no. base 'b'.

↓

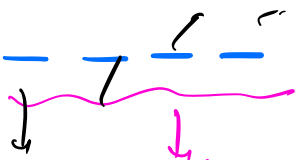
How many possible values can this no. represent?





$2 \times 2 \times 2 \times 2 \times 2$

$= 2^5$

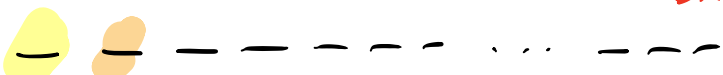


0000
0001
0010
0011
.
.
.
1111

16 nos.

$2^4 = 16$

A N digit no. with base b.



$b \times b \times b \times b \ldots$ N times

$= b^N$

Q. char. → 1 byte → 8 bits.

$2^8 = 256$.

int → 32 bits.

2^{32}

$0 - 2^{32} - 1$

Addition of Binary nos.

$$\begin{array}{r} {\scriptstyle 1} \\ 137 \\ (+)\ 256 \\ \hline 393 \end{array}$$

$$\begin{array}{r} {\scriptstyle 1\,1\,1\,1\,1} \\ 10110101 \\ +\ 10101101 \\ \hline 101100010 \end{array}$$

1+1+1 = ?

1+1 = 10

10+1 = 11

0+0 = 0

1+0 = 1

0+1 = 1

1+1 = 10

2%2=0

2/2=1

7+3 = 0

10%10

10/10

7+8 = 15

15%10 = 5

15/10 = 1 carry forward

$$\begin{array}{r} {\scriptstyle 1} \\ 7 \\ 8 \\ \hline 5 \end{array}$$

Q.

$(abcde)_k = x$

$(abcde0)_k =$

$343 \xrightarrow{\times 10} 3430$

$(3\times10^2 + 4\times10^1 + 3\times10^0)\times10$

$3\times10^3 + 4\times10^2 + 3\times10^1 + 0\times10^0$

$(10110)_2 =$

$(101100)_2 =$

Multiplied by 2.

$(101)_2 = 5$

$(1010)_2 = 10$

$1\times2^2 + 0\times2^1 + 1\times2^0$

$1\times2^3 + 0\times2^2 + 1\times2^1$

Observation / Property -

$(abcdef)_k = x$

$(abcdef0)_k = kx$

1000111
<< 1

$(34)_{11} = (37)_{10}$

$(340)_{11} = 37 \times 11$

$= 407$

Break till. 10:33.

Operators

Decimal nos. → + - * / %

Act on the whole no.

934 / 32 =

Bitwise operators.

Act on the numbers digit by digit.

(1) NOT

↳ flips the bit.

~
!

~0 = 1
~1 = 0

!(101100) = 010011

a+b
a-b
a*b
a/b

*a
/a
%a
} ✗

!a
} ✓

} → Binary operator

← Unary operator

a!b ✗

② OR ← Binary operator

x | y (OR)

Truth table.

x	y	OR
0	0	0
1	0	1
0	1	1
1	1	1

```
  10110 1
(OR) 1000 11
-------------
  10111 1
```

③ AND.

x & y

Truth table.

x	y	AND.
0	0	0
1	0	0
0	1	0
1	1	1

Q

a = 13
b = 10

a | b = ? (15) ✓

1101
(OR) 1010
1111
15

2 | 13
2 | 6 - 1
2 | 3 - 0
1 - 1

a & b = (8) ✓

1101
1010
1000
8

(4) XOR (Exclusive OR)

x ^ y

Truth table

x	y	XOR	OR
0	0	0	0
✓ 1	0	1	1
✓ 0	1	1	1
x 1	1	0	1

13 ^ 10 = ? 7 ✓

1101
1010
0111
7

NAND, NOR. ✓

Properties of these operators.

XOR

$5 \wedge 5 = 0$

$5 \wedge 0 = 5$

```
 1101
 1101
------
 0000
```

identity.

$+ \rightarrow 0$

$* \rightarrow 1$

```
 1101
 0000
------
 1101
```

$a + ? = a$

$a * ? = a$

* Identity of XOR = ? 0 ✓

* XOR cancels itself out.

$x \wedge x = 0$

* $0 \wedge x = x$

* $\ldots 00001 \wedge x = \begin{cases} x+1 & \text{if even,} \\ x-1 & \text{if odd.} \end{cases}$

```
x  010010 ←        010011 ←
   000001 ←      ^ 000001
------------     ------------
   010011           010010
```

$x \wedge (..111...11) \wedge x \quad = \quad !x$

$0 \wedge 1 = 1$
$1 \wedge 1 = 0$

$$\begin{array}{r} 010010 \\ 111111 \\ \hline 101101 \end{array}$$

OR.

1. Identity = ?

$$x \mid 0 = x$$

$$\begin{array}{r} 1010110 \\ OR\ 0000000 \\ \hline 1010110 \end{array}$$

2. $000001 \mid x =$ x if odd, $x+1$ if even

$$\begin{array}{r} 10010110 \\ 000000\,1 \\ \hline 10010111 \end{array}$$

$1 \mid 0 = 1$
$0 \mid 0 = 0$

$$\begin{array}{r} 1 \\ 1 \\ \hline 1 \end{array} \qquad \begin{array}{l} 1 \mid 1 = 1 \\ 0 \mid 1 = 1 \end{array}$$

3. $11111..1 \mid x = 111111...11$

4. $x \mid x = x$

AND.

1. Identity ?

$$x \,\&\, 1111\ldots11 = x$$

```
  11010
(AND) 11111
-------
  11010
```

2. $x \,\&\, 0 = 0$

3. $x \,\&\, 000\ldots1 =$ 1 if odd, 0 if even.

4. $x \,\&\, x = x$

```
if (n%2==0)
{   //
}
```

0 or 1

```
if (n & 1)
{   // odd.
}
```

faster

Q. Given an array of size N, with all numbers appearing twice, and 1 number appearing once. Find that no.

Ex: arr = [5, 2, 3, 4, 5, 4, 3, 1, 2, 7, 1]

5 ^ 2 ^ 3 ^ 4 ^ 5 ^ 4 ^ 3 ^ 1 ^ 2 ^ 7 ^ 1
= 7

Brute force -
TC: $O(n^2)$
SC: $O(1)$

Hash (Dictionary):
TC: $O(n)$
SC: $O(n)$

→ Solution with TC: $O(n)$, SC: $O(1)$!

idea: XOR

```
xor = 0
for ele in arr:
    xor = xor ^ ele
return xor
```

identity

```
sum = 0
for ele in arr:
    sum = sum + ele
return sum
```

ele ^ ele.

$5 \wedge 2 = 2 \wedge 5$?

Associative / Commutative operators.

$a | b = b | a$
$a \& b = b \& a$
$a \wedge b = b \wedge a$.

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

arr = {2, 1, 3, 2, 1}

XOR = (0 ^ 2 ^ 1) ^ 3) ^ 2) ^ 1

XOR = 3

2^2

(2^3) ^ 2 = 3

XOR = 00
10
10
^ 01 - 1
11
^ 11 - 3
00
^ 10 - 2
10
^ 01 - 1
11 ~~01~~

2 3 2 2 1 1

↓ 2^3 = 1

2 times.

10
11
01

1 0 1
0 1 1
1 0 1
0 1 0
0 1 0
(XOR) 0 1 1

0.1

1