# Bit - Manipulation

## operators

1> AND

| a | b | a & b |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

NOT = Complement

| | |
|---|---|
| 0 | 1 |
| 1 | 0 |

* when you & 1 with any number digits remain same

Ex

```
  1 1 0 0 1 0 1 0 0
& 1 1 1 1 1 1 1 1 1
  1 1 0 0 1 0 1 0 0
```

$(0.62)_{10} \to (\ )_2$

$0.62 \times 2 = 1.24 - 1$

$0.24 \times 2 = 0.48 - 0$

$0.48 \times 2 = 0.96 - 0$

$0.96 \times 2 = 1.92 - 1$

$0.92 \times 2 = 1.84 - 1$

$(0.62)_{10} = (0.10011\ldots)_2$

2> OR

| a | b | a or b |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

3> XOR (^) (if and only if)
exclusive OR

| a | b | a^b |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

observation

$a \wedge 1 = \bar{a}$

$a \wedge 0 = a$

$a \wedge a = 0$

4 Complement (~) or ($x^c$)

$a = 10110$

$\tilde{a} = 01001$

---

1) Decimal → 0, 1, 2, ... 9

Base = 10

2) Binary → 0 & 1

Base = 2

3) octal → 0, 1, 2, ... 7

Base = 8

4) Hexa Decimal → 0 to 9 & A, B, C, D, E, F

Base = 16 $(10)_{10} = (A)_{16}$

$(12)_{10} = (C)_{16}$

Decimal to base b

Q Convert $(17)_{10}$ to base 2

Base → Remainders

| Base | | Remainder |
|---|---|---|
| 2 | 17 | 1 |
| 2 | 8 | 0 |
| 2 | 4 | 0 |
| 2 | 2 | 0 |
| | 1 | |

$\boxed{10001}$

Any base to decimal

$(10001)_2 \longrightarrow (\ ?\ )_{10}$

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

5/ Left Shift operator (<<)

$(10)_{10} = (1010)_2$

$\boxed{a << 1 = 2a}$

$\boxed{a << b = a \times 2^b}$

$10 << 1$

Step-1 $\quad 1010 << 1 \Rightarrow 10100$

6> Right shift

$11001 >> 1 = 1100$

$a >> b = a \times 2^{-b}$

arr[] = [2, 3, 3, 4, 2, 6, 4]

u = u ^ n

a ^ 1 = a

a ^ 0 = a

a ^ a = 0

u = 0
n = 2

u = 2
n = 3

u = 1
n = 3

u = 2
n = 4

u = 6
n = 2

u = 4
n = 6

u = 2
n = 4

u = 6

101
111
01

010
100
110

0
1
10
11
01

10
10
100

2|4
2|2
1

0
0
1

for
100 - 4
101 - 5
110 →

10 0
1
10
01

u = 0

```
for (int i : arr) {
    u = u ^ i
```

3 return u

Q Finding i^th bit of a number.

1 0 1 1 0 1 1 0
8 7 6 5 4 3 2 1^st

5^th
1 0 1 1 0 1 1 0
& 0 0 0 1 0 0 0 0
0 1 0

{called mask}

making mask ⇒ 1 << 4
at 5^th index

or 1 << (n-1)
n = i

Ans n & (1 << (i-1))

set $i^{th}$ bit to 1

or
$0 \to 1$
$1 \to 1$

```
     1 0 1 0 1 1 0
 or  0 0 0 1 0 0 0
 -----------------
     1 0 1 1 1 1 0
```

reset $i^{th}$ bit (turning $1 \to 0$)

```
           3th
           ↓
     1 0 1 0 1 1 0
 &   1 1 1 1 0 1 1   → Making Mask
 -----------------
     1 0 1 0 0 1 0
```

$(!(1 << 2))$ → compliment

Negative of a number in binary form

1 Byte → 8 bit

10 = | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

→ this tells us if number is +ve or -ve → Reserve bit "0" "1"

→ if no. is odd or even

Compliment of a number

```
       n = 1 0 1
mask = XOR 1 1 1
       ---------
           0 1 0
```

Finding No. of bits in $()_2$ for 6

(-10) = Step(1) take complement
Step (2) add 1 to it

8 bit size

10 = | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 |

```
compliment ⇒   1 1 1 1 0 1 0 1
                             + 1
               ---------------
(-10) ⇒        1 1 1 1 0 1 1 0
```

Range of number in 1 byte

1> 1 byte [ | | | | | | | | ]

↓ for sign of Number

for saving No we have 7-bits

$\Rightarrow \underset{\text{+ve nos}}{2^7} + \underset{\text{-ve nos}}{2^7 - 1}$

-127 to 128

---

Q arr = [ 2, 2, 3, 2, 7, 7, 8, 7, 8, 8 ]

How you find number apparing only one tyme.

| | | | |
|---|---|---|---|
| | | 1 | 0 |
| | | 1 | 0 |
| | | 1 | 1 |
| | | 1 | 0 |
| | 1 | 1 | 1 |
| | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| | 0 | 0 | 0 |
| 1 | | | |
| 3 | 3 | 7 | 4 |
| % 3 | % 3 | % 3 | % 3 |
| 0 | 0 | 1 | 1 |

% 3

→ 3

Q) Find the magic number.

| | | 3 | 2 | 1 | . . . . | |
|---|---|---|---|---|---|---|
| | | 5 | 5 | 5 | | |
| 1 | $\rightarrow$ | 0 | 0 | 1 | $\rightarrow$ | 5 |
| 2 | $\rightarrow$ | 0 | 1 | 0 | $\rightarrow$ | 25 |
| 3 | $\rightarrow$ | 0 | 1 | 1 | $\rightarrow$ | $30 \; (5^2 + 5^1)$ |
| 4 | $\rightarrow$ | 1 | 0 | 0 | $\rightarrow$ | $5^3 = 125$ |
| 5 | $\rightarrow$ | 1 | 0 | 1 | $\rightarrow$ | $(5^3 + 5^1) = 130$ |

$n = 6 = 110$

$n \,\&\, 1 \Rightarrow$ give me last digit

$n >> 1$

$(0 * 5^1) + (1 * 5^2) + (1 \times 5^3) + 0 \cdots$

---

Q) find no. of digits in base b.

$(6)_{10} = 1$

$(6)_2 \Rightarrow 3$

$\log_b a = x$

$a = b^x$

$\log_2 6 = x$

$6 = 2^x$

$\log_2 10 = 2^{3.32}$

$(\text{int})(3.32) + (1)$

~~for~~ formula = $(\text{int})\left(\log_b n\right) + 1$

# Pascal's Triangle

row1 → 1
row 2 → 1 1
row 3 → 1 2 1
1 3 3 1
1 4 6 4 1
1 5 10 10 5 1
1 6 15 20 15 6 1
1 7 21 35 35 21 7 1

1000 → $2^2$ ($1^0$)

Q find the sum of $n^{th}$ row

Sum of each row →

$$^nC_0 + {}^nC_1 + {}^nC_2 + \dots + {}^nC_n = 2^n$$

for $n^{th}$ row; sum $= 2^{n-1}$

---

Q if the n is in power of two or not

100000 = 11111+1

10000
& 01111
0

Sol^n ⇒ $(n \,\&\, (n-1)) = 0$

↳ is in power of two

else = 0

---

Q calculate = $a^b$

$3^6 = 3 \times 3 \times 3 \times 3 \times 3 \times 3$

$3^6 = 3^{110}$

ans = 1 | n = 110
base = 3 | n & 1 = 0 → ignor
base = 9 | n = 11 >> 1 ⇒ 1 & 1 ⇒ 1 → ans = ans × base

O($\log(b)$)

```
base = 3;
power = 6;
ans = 1;
while (power > 0) {
  if (p & 1) == 1 {
     ans = ans * base
  }
  b = b * b
  p = p >> 1; }
```

Q find no. of set bits

$n = 9$

$n = 1001$

$n \& (-n) = 0001$

$n - [n \& (-n)] = 1000$

$$\begin{array}{lr} n = & 1001 \\ \& & 1000 \\ \hline & 1000 \end{array}$$

$$8 \& 7 \Rightarrow \begin{array}{r} 1000 \\ \& \; 111 \\ \hline 0 \end{array}$$

no. of set bit = No. of iteration.

```
while (n > 0){
    count ++;
    n = n & (n-1);
}
return count;
```

$\boxed{111}$

0
1
10
11

10110
10110
$\overline{\quad 110}$

Q finding position of right most set bit.

$\underbrace{1011 0}_{a}\;1\;\underbrace{00}_{b}$

$N = a1b$

$-N = \bar{a}.1b = ?$

Ans $= (N \& (-N))$

Ex $1100 = 12$

$12 \& (-12) = 4\ (100)$

$12 \rightarrow$ Negitive $(0-12) = -12$

$1\,[0 \ldots 0\,0\,0\,0\,0]$

$-[0 \ldots 0\,0\,1\,1\,0\,0]$

$\hookrightarrow$ 1 byte

$1\,[0\,0\,0 \ldots 0] = 11111111+1$

$11 \ldots 8$ types $+ 1 - 0 \ldots 1100$

$\hookrightarrow$ equal to complement of No.

Q Find XOR of nums from 0 to a.

| a | XOR from 0 to a |
| --- | --- |
| 0 | 0 |
| 1 | 0 ^ 1 = 1 |
| 2 | 0 ^ 1 ^ 2 = 3 |
| 3 | 3 ^ 3 = 0 |
| 4 | 0 ^ 4 = 4 |
| 5 | 1 |
| 6 | 7 |
| 7 | 0 |
| 8 | 8 |
| 9 | 1 |

0 → 0 1

1 → 0 0

Sol^n → looks like

| | |
| --- | --- |
| a % 4 = 0 | ans = a |
| a % 4 = 1 | " = 1 |
| " " = ~~3~~2 | " = a+1 |
| " " = 3 | " = 0 |

---

XOR b/w a & b

XOR (0 to b) ^ (0 to (a-1))

Prime No → Divisible by one and with it self.

Divisors of 36

1 x 36
2 x 18
3 x 12
4 x 9
6 x 6
9 x 4
12 x 3
18 x 2
36 x 1

$C \le \sqrt{n}$

2 x 2
3 x 3
4 x 4

6

$C * C < \sqrt{n}$

2 x 2
3 x 3

```
for i=2 ; i<√n ; i++ {
    if(n%i==0){

    }
}
```

Q find prime no.s b/w 40

Now cross All the multipul of 2 → 3, 5, 7

2 3 ~~4~~ 5 ~~6~~ 7 ~~8~~ ~~9~~ ~~10~~
11 ~~12~~ 13 ~~14~~ ~~15~~ ~~16~~ 17 ~~18~~ 19 ~~20~~
~~21~~ ~~22~~ 23 ~~24~~ ~~25~~ ~~26~~ ~~27~~ ~~28~~ 29 ~~30~~
31 ~~32~~ ~~33~~ ~~34~~ ~~35~~ ~~36~~ 37 ~~38~~ ~~39~~ ~~40~~

Time complxity
no. of multuples of 2 till 40 = $\frac{40}{2}$
" " 3 " 40 = $\frac{40}{3}$

$\left(\frac{n}{2} + \frac{n}{3} + \frac{n}{5} + \cdots\right)$

$n\left(\frac{1}{2} + \frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \cdots\right)$

using taylor → $\log(\log N)$

Time complyxity → $O(N * \log(\log N))$

way to find squre root of a number.

36

0 18 36

$\hookrightarrow 18 \times 18 = 36$ ✗

$\downarrow$

$9 \times 9 = 36$ ✗

$\downarrow$

$4 \times 4 = 36$ ✗

$\downarrow$

$6 \times 6 = 36$ ✓

if (m × m > n)

(e = m − 1)

else s = m+1

$\sqrt{40} \Rightarrow ?$ (6.32)

this we can get from above

$6.1 \times 6.1 < 40$

$\downarrow$ T

$6.2 \times 6.2 < 40$

$\downarrow$ T

$6.3 \times 6.3 < 40$

$\downarrow$ ✓

$6.4 \times 6.4 < 40$

$\downarrow$ ✗

$6.31 \times 6.31 < 40$

$\downarrow$ T

$6.32 \times 6.32 < 40$ → Ans

$\downarrow$ ✗

```
double incr = 0.1
for (i = 0; i < p; i++)
while ( _ <= target ) {
_ = _ + incr }
}
_ -= icr
incr /= 10;
```

another way to find sqrt

Newton Raphson method

$$\text{root} = \frac{\left(x + \frac{N}{x}\right)}{2}$$

x = any sqrt you assumed/gess

lets say you gess ✓ Ans ] $x = \sqrt{N}$

$$\frac{\left(\sqrt{N} + \frac{N}{\sqrt{N}}\right)}{2} = \sqrt{N} + \frac{\sqrt{N}\sqrt{N}}{\sqrt{N}} = \sqrt{N}$$

x → you assumed

$\sqrt{N}$ → is actual Ans

error ⇒ ? = $|\sqrt{N} - x|$

So, we are keep changing the value of x till the ~~er~~ error becoms minimal.

Writing program →

1) Initially Assign x it N / You find your Ans when error is < 1

2)

```
Public Static double ____ (double n){

double x = n;
double root;
while (true){
    root = (x + n/2) * 0.5
    if ( Math.abs (root - x) < 0.5) {
    break; } x = root :
        } return root;
```

↳ smaller the number higher the prisigeon.

---

FACTORS OF A NUMBER →

20 ÷ 1
20 ÷ 2
20 ÷ 3 - x
20 ÷ 4
20 ÷ 5
20 ÷ 6
20 ÷ 7
20 ÷ 8
20 ÷ 9
20 ÷ 10
20 ÷ 11
20 ÷ 12

```
for (i=1 ; i<= Math. Sqrt (n) ; i++ ){
if (n % i = = 0) {
    if (n/i = = i ) {
        sout (i+ " "); }
    else { sout (i+ " "+ n/i+ " ");
    }
}
```

# Propertyes of Modulo ( % )

★ $(a \pm b) \% m = ((a \% m) \pm (b \% m)) \% m$
$\quad\;\; *\qquad\qquad\qquad\qquad\quad *$

$\left(\frac{a}{b}\right) \% m = \left((a \% m) * (b^{-1} \% m)\right) \% m$

$b^{-1} \% m$ £ Multiplicative modulo inverse (mmi)

Ex 1 $(6 * y) \% 7 = 1$

$y =$ mmi for 6 & $y = 6$
$(6*6) \% 7 = 36 \% 7 = 1$

$b^{-1} \% m$ means that $b$ & $m$ are co-prims.

★ $(a \% m) \% m = a \% m$

★ $m^x \% m = 0 \quad \forall\, x \in$ +ve intgers

★ If $p$ is prime no. which is not a divisor of $b$, then ~~ab~~ $(ab^{p-1} \% p = a \% p)$ due to Fermat's Little theorm.

---

Die hard Example

4-galons of water


$1^{st} \rightarrow \underset{A\;\;B}{(0,0)} \rightarrow (3,0) \rightarrow (0,3)$

$2^{nd} \rightarrow (0,3) \rightarrow (3,3) \rightarrow (1,5) \rightarrow (1,0) \rightarrow (0,1)$

$3^{rd} \rightarrow (0-1) \rightarrow (3,1) \rightarrow (0,4)$

HCF of a & b = Min +ve value of eqn $(ax + yb)$

where x & y are int.

HCF ( 4, 18 )

1, 2, 4     1, 2, 3, 6, 9, 18 .

↳ HCF

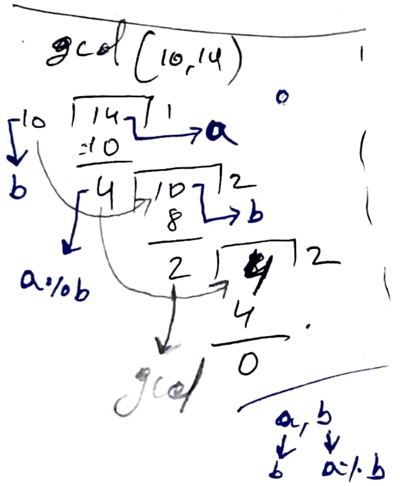
6     ~~18~~ 12

1, 2, 3, 6     1, 2, 3, 4, 6, ~~12~~

~~Eud~~ Euclid's Algo.

gcd (a, b) = gcd ( rem (b, a), a )

or HCF




```
HCF ( a, b ) {
  if ( a == 0 ) {
    return b; }
  return HCF ( b % a , a );
```

LCM

(2, 4) = 4

(3, 7) = 21

$$LCM = \frac{a \times b}{HCF(a, b)}$$