n=367
k=4

1 0 0 0 0 1 0 0 0 0
1 0 1 1 1 1 1 1 1

n = 367 1 0 1
& 1
1

cnt => 1 + 1 + 1 + 1 + 1

int cnt = 0
while(n>0) {
if((n & 1) != 0) cnt++
n >>= 1.
}

10110111

n = n & (n-1)

40 -> 1 0 1 0 0 0
39 -> 1 0 0 1 1 1
32 1 0 0 0 0 0
31 0 1 1 1 1 1
0

cnt += 1 + 1 + 1

101
& 001
001

1010
1001

+ * - / %

int & ^ Bitwise operator
& | ^ << >>
left shift, Right shift

& => 12 & 10 => 8

And (&)

1100
& 1010
1000

Or (|)

1100 -> 12
| 0010 -> 2
1110 => 14

12 | 2 => 14

Xor (^)

12 ^ 5 => 9

1100
^ 0101
1001

Same => 0
diff => 1

0 ^ 0 => 0
0 ^ 1 => 1
1 ^ 0 => 1
1 ^ 1 => 0

XOR

I. n^0 = n (1010 ^ 0000 = 1010)

II. n^n = 0 (1010 ^ 1010 = 0000)

III. a^b = b^a

4^2^5^6^2^5^3^4^3
⇓
2^2^4^4^3^3^5^5^6
0 ^ 0 ^ 0 ^ 0 ^ 6
0^6 => 6

n = 0/16 1/8 0/4 1/2 0/1 => 10
n = n<<1 << => ×2
n = 1/16 0/8 1/4 0/2 0/1 => 20
n = n<<2 => ×4

>>

n = n>>1
0/16 1/8 0/4 1/2 –/1 => 5
>> => /2

21/2 => 10 ×2 => 20

10101 => 21
>>1
01010 => 10
<<1
10100 => 20

367 => 1/256 0/128 1/64 1/32 0/16 1/8 1/4 1/2 1/1

366
-256
110
-64
46
-32
14

110|110

110
55
27
13
6
3
1
0

183
367
↓
91
+83
↓
9145
22 # 5 2 + 0

101101111

14 => 1/8 1/4 1/2 0/1

365

0 — 9
0 1

10 365

Dec 3 × 10^2 + 6 × 10^1 + 5 × 10^0 = 365
x -> Dec, Bin, Hex, Octa

101

0-1

1 × 2^2 + 0 × 2^1 + 1 × 2^0 =
4 + 0 + 1 = 5

×1000 ×100 ×10 ×1

1/8 0/4 1/2 0/1 => 8+2 = 10

4 2 6 2 8 3 4 3 2 2
xor of all = 6^8
kth bit is set | kth bit is not set
2 6 2 3 3 | 4 8 4 1 1
6 | 8
110
^1000

⊕ - /

int mask = (1<<k);
return (n & mask) != 0:

n=367
k= 5
& Bitmask

Check kth Bit is set in n

n= 0 0 0 1 0 0 0 0 0
0 0 0 1 0 0 0 0 0

set => 1
unset => 0

mask = 1<<k
if(n & mask != 0) return true
else return false

25 => 1/16 1/8 0/4 0/2 1/1

25 => 1/16 1/8 0/4 0/2 1/1

1100 => 12

$\log_2 n$

4^2^5^6^2^5^3^4^3

n => 100
^ 10
110
101
011
101
110
010
100
101
011
100
011

100
010
101
110
010
101
011
100
011
110 => 6