Lec-2

12/06/25 10:16 AM

Bitmasking + math

Bit-wise → operation

0 → Bit wise → and
1 → Bit wise → or

^ → XOR
~ → not a
<< → Left shift
>> → Right shift

A	B	A&B	A|B	A^B
1	1	1	1	0
1	0	0	1	1
0	1	0	1	1
0	0	0	0	0

a^b^c = (a^b)^c
= a^(b^c)
= (a^c)^b

a^a = 0
a^0 = a

~(1) = 0

n%2 == 0

n(n+1)/2 - sum(arr)

N = 7 (1+7)

2	1	4	7	3	6
0	1	2	3	4	5

(1^n) (a^n) → 1st

2 | 7 | 7 (1)

(n & 1)

byte b = 5

a → 00000101 → 10

00001010 → 10

1 << 10 → 00001010

00001010

byte b = -11

00001011
11110100
1
11110101

a<<1 = a*2 = 10

a<<n = a*2^n

(1<<n) = 1*2^n

O(1)

The problem is:

You are given a *lucky number* n. *Lucky numbers* are the positive integers whose decimal representations contain only the lucky digits 4 and 7. For example, numbers **47**, **744**, **4** are lucky and **5**, **17**, **467** are not.

If we sort all lucky numbers in increasing order, what's the 1-based index of n?

Tavas is not as smart as SaDDas, so he asked you to do him a favor and solve this problem so he can have his headphones back.

Input

The first and only line of input contains a lucky number n ($1 \le n \le 10^9$).

4 7 44 47 74 77 444 447

1 2 3 4 5

2^n → sum = n

10^9 → 1000000000

pure

number

2 → 1
4 → 2
8 → 3
16 → 4
32 → 5

StrCount

digit digit digit digit digit

0 digit (mod)

47474 → length = n

1<<n = 1*2^n

n digit

(1<<n) - 2

sum { 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 ... 2^(n-1) }

a(r^n - 1) / (r - 1) = 2(2^(n-1) - 1) / (2 - 1) = (2*2^(n-1) - 2) / 1 = 2^n - 2

47 ---

47474 → 2+8+16 → 26

47474

47474

47474

47474

47474

47474

101
011
110 → 8

1,2,1,2,2,5

a & mable

c ^ a ^ 5

100 → max

111
010
1100
1011

C = {7, 12, 11, 5, 2, 5, 24, 11}

7	2
7	11
12	2
5	24
5	11

16

12

24

0 11 00
1 100 0
1 011 00

12^118

0 1 0 0 0

1

0101 0111
1111
0101 1000

1010 1000

1000

n*(n-1)

n^2

n&(-n) → 2^n

n

0101 0100
0101 0011
0101 0000
0101 0000

abc

→ 3 → 2^3 = 8

abc

000 → -
001 → a
010 → b
011 → ab
100 → c
101 → ac
110 → bc
111 → abc

i = 5

101 → 2

i>>1

1 0 >> 1

1>>2

ac

i = 5

abc

101 → 2

while (i>>c)
if (c&1) ans

i = i>>1