3094. Guess the Number Using Bitwise Questions II Premium

Medium | Topics | Hint

There is a number `n` between `0` and $2^{30} - 1$ (both inclusive) that you have to find.

There is a pre-defined API `int commonBits(int num)` that helps you with your mission. But here is the challenge, every time you call this function, `n` changes in some way. But keep in mind, that you have to find the **initial value of** `n`.

`commonBits(int num)` acts as follows:

- Calculate `count` which is the number of bits where both `n` and `num` have the same value in that position of their binary representation.
- `n = n XOR num`
- Return `count`.

Return *the number* `n`.

Note: In this world, all numbers are between `0` and $2^{30} - 1$ (both inclusive), thus for counting common bits, we see only the first 30 bits of those numbers.

Constraints:

- $0 <= n <= 2^{30} - 1$
- $0 <= num <= 2^{30} - 1$
- If you ask for some `num` out of the given range, the output wouldn't be reliable.

Seen this question in a real interview before? 1/5

Yes No

Accepted 615 | Submissions 752 | Acceptance Rate 81.8%

Topics

Bit Manipulation Interactive

Hint 1

Ask the number 0 and save the result in `base`.

Hint 2

Ask 2^i for $0 <= i < 30$.

Hint 3

If the result is greater than `base` for some `i`, then this bit is a set bit in `n`.

Hint 4

What can be done to revert the effect of the `XOR`.

Hint 5

Doing `XOR` again with the same number reverts the effect.

Discussion (1)