Number Complement

Solution

Given a **positive** integer `num`, output its complement number. The complement strategy is to flip the bits of its binary representation.

Example 1:

```
Input: num = 5
Output: 2
Explanation: The binary representation of 5 is 101 (no leading zero bits), and its complement is 010. So you need
```

Example 2:

```
Input: num = 1
Output: 0
Explanation: The binary representation of 1 is 1 (no leading zero bits), and its complement is 0. So you need to
```

Constraints:

- The given integer `num` is guaranteed to fit within the range of a 32-bit signed integer.
- `num >= 1`
- You could assume no leading zero bit in the integer's binary representation.
- This question is the same as 1009: https://leetcode.com/problems/complement-of-base-10-integer/

Java

```java
class Solution {
    public int findComplement(int num) {
        int result=0;
        int power=1;
        while(num!=0){
            result+=(num%2^1)*power;
            num>>=1;
            power<<=1;
        }
        return result;
    }
}
```

Custom Testcase (Contribute)

Run Code Submit