# 题目

给定一个正整数，输出它的补数。补数是对该数的二进制表示取反。

**示例 1:**

输入: 5

输出: 2

解释: 5 的二进制表示为 101（没有前导零位），其补数为 010。所以你需要输出 2 。

**示例 2:**

输入: 1

输出: 0

解释: 1 的二进制表示为 1（没有前导零位），其补数为 0。所以你需要输出 0 。

**注意:**

给定的整数保证在32位带符号整数的范围内。

你可以假定二进制数不包含前导零位。

本题与1009相同：

https://leetcode-cn.com/problems/complement-of-base-10-integer/

# 分析

## 方法一：位运算

class Solution {

public:

int findComplement(int num) {

int tmp = 1;

while (tmp < num)//当tmp>num时,此时正好右侧多1位

{

tmp <<= 1;

tmp += 1;

}

return (tmp^num);

//此时^操作必然最后多的1位为1,其余为0,即取反

}

};