

## 201. 数字范围按位与

难度 中等

👁 236

☆ 收藏

🔗 分享

🌐 切换为英文

🔔 接收动态

🗉 反馈

给定范围  $[m, n]$ ，其中  $0 \leq m \leq n \leq 2147483647$ ，返回此范围内所有数字的按位与（包含  $m, n$  两端点）。

示例 1:

输入:  $[5, 7]$

输出: 4

示例 2:

输入:  $[0, 1]$

输出: 0

通过次数 35,740

提交次数 69,791

## 【解答思路】

### 1. 暴力

逐位与，只需要判断  $i = 0$  或  $i == 2147483647$  的话，就跳出 for 循环即可。

右边界  $n$  是  $2147483647$ ，也就是 `Integer` 中最大的正数，二进制形式是  $01111\dots1$ ，其中有  $31$  个  $1$ 。在代码中当  $i$  等于  $n$  的时候依旧会进入循环。出循环执行  $i++$ ，我们期望它变成  $2147483647 + 1 = 2147483648$ ，然后跳出 for 循环。事实上，对  $2147483647$  加  $1$ ，也就是  $01111\dots1$  加  $1$ ，变成了  $1000\dots000$ ，其中有  $31$  个  $1$ 。而这个二进制在补码中表示的是  $-2147483648$ 。因此我们依旧会进入 for 循环，以此往复，直到结果是  $0$  才出了 for 循环。。

时间复杂度： $O(N)$  空间复杂度： $O(1)$

```
1 public int rangeBitwiseAnd(int m, int n) {
2     //m 要赋值给 i, 所以提前判断一下
3     if(m == Integer.MAX_VALUE){
4         return m;
5     }
6     int res = m;
7     for (int i = m + 1; i <= n; i++) {
8         res &= i;
9         if(res == 0 || i == Integer.MAX_VALUE){
10             break;
11         }
12     }
13     return res;
14 }
```