Bitwise AND of Numbers Range

We need the AND of all numbers from left to right.

Key Insight:

The AND of range [left, right] essentially keeps only the common prefix bits of left and right.

- Any bit that changes within the range will eventually become 0.
- So we find the highest common prefix of left and right.

Approach 1: Shift until equal:

1. While left < right right-shift both until they are equal.
2. Keep track of how many shifts we did.
3. Shift back left afterwards.

Approach 2: Remove lowest set bit of right:

Keep clearing the least significant bit of right until right <= left.
This effectively removes the varying bits.

Both methods run in O(1) to O(log n) time, which is efficient for 32-bit integers.