Documentation: Checking for Power of Two

Overview

The isPowerOfTwo function is designed to determine whether a given integer (n) is a power of two. A number is considered a power of two if it can be expressed in the form (2^x), where (x) is a non-negative integer. This function leverages bitwise operations to achieve its goal without the need for loops or recursion, providing an efficient solution to the problem.

Functionality

The function accepts a single integer (n) as input and returns a boolean value. The output will be true if (n) is a power of two and false otherwise. The function works by first checking if (n) is greater than zero. If (n) is less than or equal to zero, it cannot be a power of two, so the function will return false. This check is crucial as it eliminates negative numbers and zero from consideration, as they do not meet the criteria for being a power of two.

Bitwise Operation

To efficiently determine if (n) is a power of two, the function employs a bitwise operation, specifically (n & (n - 1)). This operation clears the lowest set bit of (n). If (n) is a power of two, this operation will yield zero because there is only one bit set to 1 in its binary representation. Consequently, if the result of this operation is zero and (n) is greater than zero, then (n) qualifies as a power of two.

Constraints

The function operates under the constraint that (n) can range from (-2^{31}) to (2^{31} - 1). This range corresponds to the limits of a 32-bit signed integer. The implementation ensures that it efficiently handles edge cases such as the minimum and maximum values of integers within this range. Notably, all negative integers and zero are promptly excluded from being considered powers of two.

Conclusion

The isPowerOfTwo function provides a robust and efficient means to check if a number is a power of two using bitwise logic. By avoiding loops and recursion, it minimizes computational overhead and maintains clarity in its operation. This function serves as a valuable tool in programming scenarios where quick determination of powers of two is necessary, such as in optimization problems, algorithm design, and systems programming.