

Solution (Bitwise Approach)

A number is a power of two if it has exactly one bit set in its binary representation. This can be checked using the condition:

$$n > 0 \quad \text{and} \quad (n \& (n - 1)) == 0$$

- Powers of two have only one bit set (e.g., 1 (01), 2 (10), 4 (100), 8 (1000), etc.).
- Subtracting 1 flips all bits after the rightmost set bit.
- Performing $n \& (n - 1)$ results in 0 if n is a power of two.

Code Implementation (Python)

```
def isPowerOfTwo(n): return n > 0 and (n &
(n - 1)) == 0 # Test cases
print(isPowerOfTwo(1)) # True
print(isPowerOfTwo(16)) # True
print(isPowerOfTwo(3)) # False
```