Problem 231: Power of Two

Problem Information

Difficulty: Easy
Acceptance Rate: 49.58%
Paid Only: No
Tags: Math, Bit Manipulation, Recursion

Problem Description

Given an integer `n`, return _`true` if it is a power of two. Otherwise, return `false`_.

An integer `n` is a power of two, if there exists an integer `x` such that `n == 2x`.

Example 1:

Input: n = 1 **Output:** true **Explanation:** 20 = 1

Example 2:

Input: n = 16 **Output:** true **Explanation:** 24 = 16

Example 3:

Input: n = 3 **Output:** false

Constraints:

* `-231 <= n <= 231 - 1`

Follow up: Could you solve it without loops/recursion?

Code Snippets

C++:

```
class Solution {
public:
bool isPowerOfTwo(int n) {

}
};
```

Java:

```
class Solution {
public boolean isPowerOfTwo(int n) {

}
}
```

Python3:

```
class Solution:
def isPowerOfTwo(self, n: int) -> bool:
```