

Problem 326: Power of Three

Problem Information

Difficulty: Easy

Acceptance Rate: 50.31%

Paid Only: No

Tags: Math, Recursion

Problem Description

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

An integer `n` is a power of three, if there exists an integer `x` such that `n == 3x`.

Example 1:

Input: `n = 27` **Output:** `true` **Explanation:** `27 = 33`

Example 2:

Input: `n = 0` **Output:** `false` **Explanation:** There is no `x` where `3x = 0`.

Example 3:

Input: `n = -1` **Output:** `false` **Explanation:** There is no `x` where `3x = (-1)`.

Constraints:

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

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

Code Snippets

C++:

```
class Solution {  
public:  
    bool isPowerOfThree(int n) {  
  
    }  
};
```

Java:

```
class Solution {  
    public boolean isPowerOfThree(int n) {  
  
    }  
}
```

Python3:

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