Problem 1553: Minimum Number of Days to Eat N Oranges

Problem Information

Difficulty: Hard
Acceptance Rate: 35.94%
Paid Only: No
Tags: Dynamic Programming, Memoization

Problem Description

There are `n` oranges in the kitchen and you decided to eat some of these oranges every day as follows:

* Eat one orange. * If the number of remaining oranges `n` is divisible by `2` then you can eat `n / 2` oranges. * If the number of remaining oranges `n` is divisible by `3` then you can eat `2 * (n / 3)` oranges.

You can only choose one of the actions per day.

Given the integer `n`, return _the minimum number of days to eat_ `n` _oranges_.

Example 1:

Input: n = 10 **Output:** 4 **Explanation:** You have 10 oranges. Day 1: Eat 1 orange, 10 - 1 = 9. Day 2: Eat 6 oranges, 9 - 2*(9/3) = 9 - 6 = 3. (Since 9 is divisible by 3) Day 3: Eat 2 oranges, 3 - 2*(3/3) = 3 - 2 = 1. Day 4: Eat the last orange 1 - 1 = 0. You need at least 4 days to eat the 10 oranges.

Example 2:

Input: n = 6 **Output:** 3 **Explanation:** You have 6 oranges. Day 1: Eat 3 oranges, 6 - 6/2 = 6 - 3 = 3. (Since 6 is divisible by 2). Day 2: Eat 2 oranges, 3 - 2*(3/3) = 3 - 2 = 1. (Since 3 is divisible by 3) Day 3: Eat the last orange 1 - 1 = 0. You need at least 3 days to eat the 6 oranges.

Constraints:

* `1 <= n <= 2 * 109`

Code Snippets

C++:

```
class Solution {
public:
int minDays(int n) {

}
};
```

Java:

```
class Solution {
public int minDays(int n) {

}
}
```

Python3:

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