Problem 672: Bulb Switcher II

Problem Information

Difficulty: Medium
Acceptance Rate: 49.90%
Paid Only: No
Tags: Math, Bit Manipulation, Depth-First Search, Breadth-First Search

Problem Description

There is a room with `n` bulbs labeled from `1` to `n` that all are turned on initially, and **four buttons** on the wall. Each of the four buttons has a different functionality where:

* **Button 1:** Flips the status of all the bulbs. * **Button 2:** Flips the status of all the bulbs with even labels (i.e., `2, 4, ...`). * **Button 3:** Flips the status of all the bulbs with odd labels (i.e., `1, 3, ...`). * **Button 4:** Flips the status of all the bulbs with a label `j = 3k + 1` where `k = 0, 1, 2, ...` (i.e., `1, 4, 7, 10, ...`).

You must make **exactly** `presses` button presses in total. For each press, you may pick **any** of the four buttons to press.

Given the two integers `n` and `presses`, return _the number of**different possible statuses** after performing all _`presses` _button presses_.

Example 1:

Input: n = 1, presses = 1 **Output:** 2 **Explanation:** Status can be: - [off] by pressing button 1 - [on] by pressing button 2

Example 2:

Input: n = 2, presses = 1 **Output:** 3 **Explanation:** Status can be: - [off, off] by pressing button 1 - [on, off] by pressing button 2 - [off, on] by pressing button 3

Example 3:

Input: n = 3, presses = 1 **Output:** 4 **Explanation:** Status can be: - [off, off, off] by pressing button 1 - [off, on, off] by pressing button 2 - [on, off, on] by pressing button 3 - [off, on, on] by pressing button 4

Constraints:

* `1 <= n <= 1000` * `0 <= presses <= 1000`

Code Snippets

C++:

```
class Solution {
public:
int flipLights(int n, int presses) {

}
};
```

Java:

```
class Solution {
public int flipLights(int n, int presses) {

}
}
```

Python3:

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