Problem 2549: Count Distinct Numbers on Board

Problem Information

Difficulty: Easy
Acceptance Rate: 61.54%
Paid Only: No
Tags: Array, Hash Table, Math, Simulation

Problem Description

You are given a positive integer `n`, that is initially placed on a board. Every day, for `109` days, you perform the following procedure:

* For each number `x` present on the board, find all numbers `1 <= i <= n` such that `x % i == 1`. * Then, place those numbers on the board.

Return _the number of**distinct** integers present on the board after_ `109` _days have elapsed_.

Note:

* Once a number is placed on the board, it will remain on it until the end. * `%` stands for the modulo operation. For example, `14 % 3` is `2`.

Example 1:

Input: n = 5 **Output:** 4 **Explanation:** Initially, 5 is present on the board. The next day, 2 and 4 will be added since 5 % 2 == 1 and 5 % 4 == 1. After that day, 3 will be added to the board because 4 % 3 == 1. At the end of a billion days, the distinct numbers on the board will be 2, 3, 4, and 5.

Example 2:

Input: n = 3 **Output:** 2 **Explanation:** Since 3 % 2 == 1, 2 will be added to the board. After a billion days, the only two distinct numbers on the board are 2 and 3.

Constraints:

* `1 <= n <= 100`

Code Snippets

C++:

```cpp
class Solution {
public:
int distinctIntegers(int n) {

}
};
```

Java:

```java
class Solution {
public int distinctIntegers(int n) {

}
}
```

Python3:

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