Problem 1088: Confusing Number II

Problem Information

Difficulty: Hard
Acceptance Rate: 47.08%
Paid Only: Yes
Tags: Math, Backtracking

Problem Description

A **confusing number** is a number that when rotated `180` degrees becomes a different number with **each digit valid**.

We can rotate digits of a number by `180` degrees to form new digits.

* When `0`, `1`, `6`, `8`, and `9` are rotated `180` degrees, they become `0`, `1`, `9`, `8`, and `6` respectively. * When `2`, `3`, `4`, `5`, and `7` are rotated `180` degrees, they become **invalid**.

Note that after rotating a number, we can ignore leading zeros.

* For example, after rotating `8000`, we have `0008` which is considered as just `8`.

Given an integer `n`, return _the number of**confusing numbers** in the inclusive range _`[1, n]`.

Example 1:

Input: n = 20 **Output:** 6 **Explanation:** The confusing numbers are [6,9,10,16,18,19]. 6 converts to 9. 9 converts to 6. 10 converts to 01 which is just 1. 16 converts to 91. 18 converts to 81. 19 converts to 61.

Example 2:

Input: n = 100 **Output:** 19 **Explanation:** The confusing numbers are [6,9,10,16,18,19,60,61,66,68,80,81,86,89,90,91,98,99,100].

Constraints:

* `1 <= n <= 109`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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