Problem 2048: Next Greater Numerically Balanced Number

Problem Information

Difficulty: Medium
Acceptance Rate: 62.99%
Paid Only: No
Tags: Hash Table, Math, Backtracking, Counting, Enumeration

Problem Description

An integer `x` is **numerically balanced** if for every digit `d` in the number `x`, there are **exactly** `d` occurrences of that digit in `x`.

Given an integer `n`, return _the**smallest numerically balanced** number **strictly greater** than _`n` _._

Example 1:

Input: n = 1 **Output:** 22 **Explanation:** 22 is numerically balanced since: - The digit 2 occurs 2 times. It is also the smallest numerically balanced number strictly greater than 1.

Example 2:

Input: n = 1000 **Output:** 1333 **Explanation:** 1333 is numerically balanced since: - The digit 1 occurs 1 time. - The digit 3 occurs 3 times. It is also the smallest numerically balanced number strictly greater than 1000. Note that 1022 cannot be the answer because 0 appeared more than 0 times.

Example 3:

Input: n = 3000 **Output:** 3133 **Explanation:** 3133 is numerically balanced since: - The digit 1 occurs 1 time. - The digit 3 occurs 3 times. It is also the smallest numerically balanced number strictly greater than 3000.

Constraints:

* `0 <= n <= 106`

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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