Problem 3345: Smallest Divisible Digit Product I

Problem Information

Difficulty: Easy
Acceptance Rate: 64.23%
Paid Only: No
Tags: Math, Enumeration

Problem Description

You are given two integers `n` and `t`. Return the **smallest** number greater than or equal to `n` such that the **product of its digits** is divisible by `t`.

Example 1:

Input: n = 10, t = 2

Output: 10

Explanation:

The digit product of 10 is 0, which is divisible by 2, making it the smallest number greater than or equal to 10 that satisfies the condition.

Example 2:

Input: n = 15, t = 3

Output: 16

Explanation:

The digit product of 16 is 6, which is divisible by 3, making it the smallest number greater than or equal to 15 that satisfies the condition.

Constraints:

* `1 <= n <= 100` * `1 <= t <= 10`

Code Snippets

C++:

```
class Solution {
public:
int smallestNumber(int n, int t) {

}
};
```

Java:

```
class Solution {
public int smallestNumber(int n, int t) {

}
}
```

Python3:

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