Problem 2457: Minimum Addition to Make Integer Beautiful

Problem Information

Difficulty: Medium
Acceptance Rate: 38.34%
Paid Only: No
Tags: Math, Greedy

Problem Description

You are given two positive integers `n` and `target`.

An integer is considered **beautiful** if the sum of its digits is less than or equal to `target`.

Return the _minimum**non-negative** integer _`x` _such that_`n + x` _is beautiful_. The input will be generated such that it is always possible to make `n` beautiful.

Example 1:

Input: n = 16, target = 6 **Output:** 4 **Explanation:** Initially n is 16 and its digit sum is 1 + 6 = 7. After adding 4, n becomes 20 and digit sum becomes 2 + 0 = 2. It can be shown that we can not make n beautiful with adding non-negative integer less than 4.

Example 2:

Input: n = 467, target = 6 **Output:** 33 **Explanation:** Initially n is 467 and its digit sum is 4 + 6 + 7 = 17. After adding 33, n becomes 500 and digit sum becomes 5 + 0 + 0 = 5. It can be shown that we can not make n beautiful with adding non-negative integer less than 33.

Example 3:

Input: n = 1, target = 1 **Output:** 0 **Explanation:** Initially n is 1 and its digit sum is 1, which is already smaller than or equal to target.

Constraints:

* `1 <= n <= 1012` * `1 <= target <= 150` * The input will be generated such that it is always possible to make `n` beautiful.

Code Snippets

C++:

```
class Solution {
public:
long long makeIntegerBeautiful(long long n, int target) {

}
};
```

Java:

```
class Solution {
public long makeIntegerBeautiful(long n, int target) {

}
}
```

Python3:

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