Problem 964: Least Operators to Express Number

Problem Information

Difficulty: Hard
Acceptance Rate: 48.65%
Paid Only: No
Tags: Math, Dynamic Programming, Memoization

Problem Description

Given a single positive integer `x`, we will write an expression of the form `x (op1) x (op2) x (op3) x ...` where each operator `op1`, `op2`, etc. is either addition, subtraction, multiplication, or division (`+`, `-`, `*`, or `/)`. For example, with `x = 3`, we might write `3 * 3 / 3 + 3 - 3` which is a value of 3.

When writing such an expression, we adhere to the following conventions:

* The division operator (`/`) returns rational numbers. * There are no parentheses placed anywhere. * We use the usual order of operations: multiplication and division happen before addition and subtraction. * It is not allowed to use the unary negation operator (`-`). For example, "`x - x`" is a valid expression as it only uses subtraction, but "`-x + x`" is not because it uses negation.

We would like to write an expression with the least number of operators such that the expression equals the given `target`. Return the least number of operators used.

Example 1:

Input: x = 3, target = 19 **Output:** 5 **Explanation:** 3 * 3 + 3 * 3 + 3 / 3. The expression contains 5 operations.

Example 2:

Input: x = 5, target = 501 **Output:** 8 **Explanation:** 5 * 5 * 5 * 5 - 5 * 5 * 5 + 5 / 5. The expression contains 8 operations.

Example 3:

Input: x = 100, target = 100000000 **Output:** 3 **Explanation:** 100 * 100 * 100 * 100. The expression contains 3 operations.

Constraints:

* `2 <= x <= 100` * `1 <= target <= 2 * 108`

Code Snippets

C++:

```
class Solution {
public:
int leastOpsExpressTarget(int x, int target) {

}
};
```

Java:

```
class Solution {
public int leastOpsExpressTarget(int x, int target) {

}
}
```

Python3:

```
class Solution:
def leastOpsExpressTarget(self, x: int, target: int) -> int:
```