**Topic 6.5: Target Sum Expression Problem**

**Question**  
You are given an integer array nums and an integer target. You want to build an expression out of nums by adding one of the symbols '+' and '-' before each integer in nums and then concatenate all the integers.

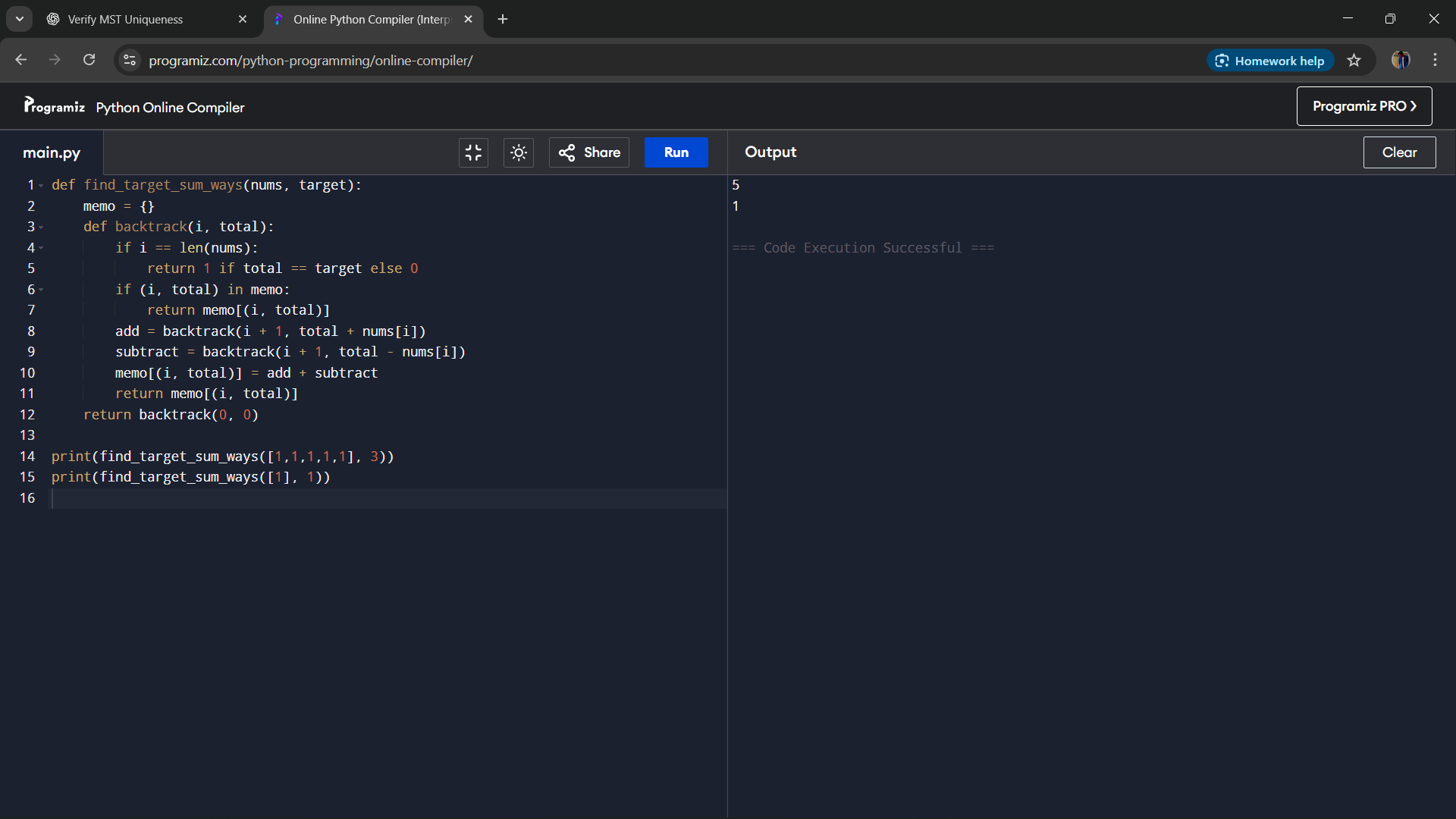
For example, if nums = [2, 1], you can add a '+' before 2 and a '-' before 1 and concatenate them to build the expression "+2-1".

Return the number of different expressions that you can build, which evaluates to target.

**Aim**  
To count the number of different expressions formed using '+' and '-' operators in front of each integer of the array such that the final expression evaluates to the target value.

**Algorithm**

1. Define a recursive or dynamic programming approach to explore all possible symbol assignments.
2. At each step, choose either '+' or '-' for the current element.
3. Recursively compute the result for the remaining elements.
4. If the sum equals the target when reaching the end, count that as a valid expression.
5. Use memoization to optimize repeated subproblems.
6. Return the total number of valid expressions.

**Output**

**Result**  
The program successfully calculates the number of expressions that can be formed with '+' and '-' operators so that the result equals the target.

**Performance Analysis**

* Time Complexity: O(n × sum(nums)) using dynamic programming with memoization.
* Space Complexity: O(n × sum(nums)) for the memoization table.