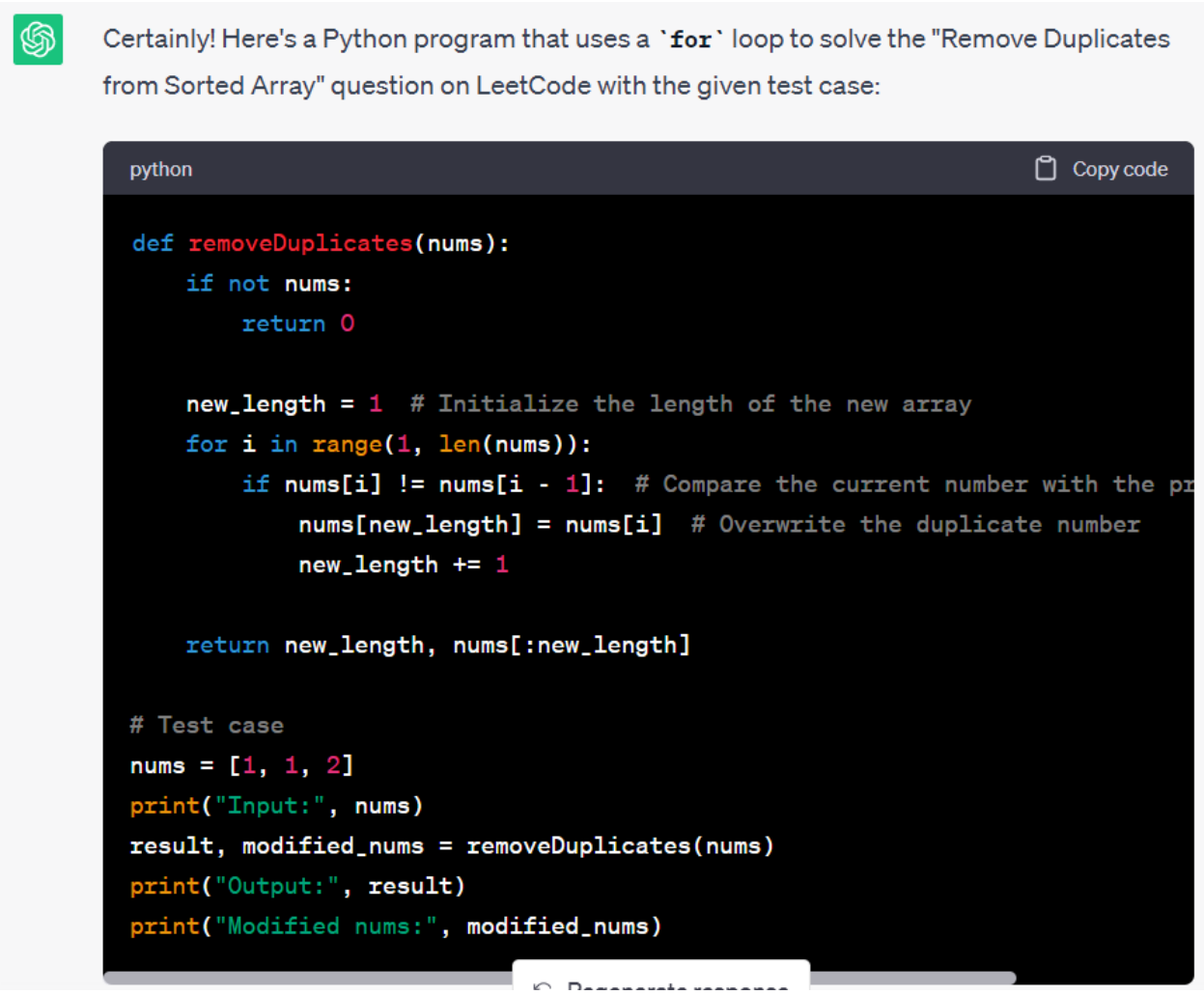
Step 1: Ask ChatGPT



Certainly! Here's a Python program that uses a `for` loop to solve the "Remove Duplicates from Sorted Array" question on LeetCode with the given test case:

python Copy code

```python
def removeDuplicates(nums):
    if not nums:
        return 0

    new_length = 1  # Initialize the length of the new array
    for i in range(1, len(nums)):
        if nums[i] != nums[i - 1]:  # Compare the current number with the pr
            nums[new_length] = nums[i]  # Overwrite the duplicate number
            new_length += 1

    return new_length, nums[:new_length]

# Test case
nums = [1, 1, 2]
print("Input:", nums)
result, modified_nums = removeDuplicates(nums)
print("Output:", result)
print("Modified nums:", modified_nums)
```

When you run this program, it will output:

yaml Copy code

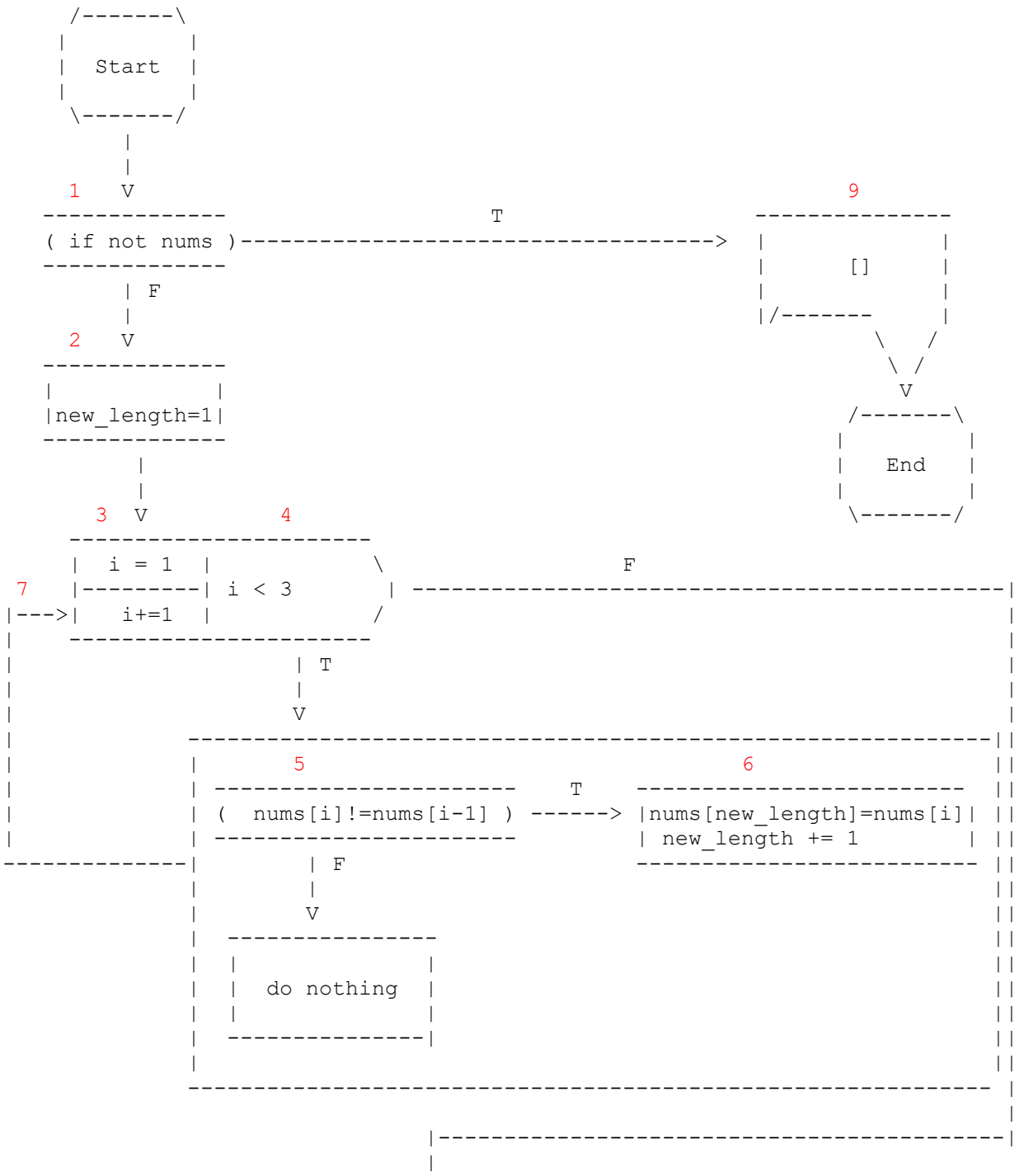
```yaml
Input: [1, 1, 2]
Output: 2
Modified nums: [1, 2]
```

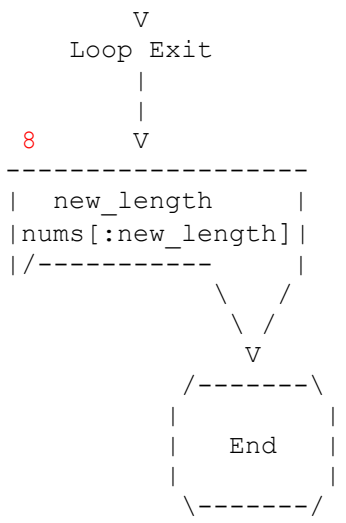
Step 2: Study ChatGPT's Python code and explanation.

The program will loop over the entire nums list and compare the current number with the previous number. When the nums list is empty, return []; otherwise compare the current number with the previous number one by one. Here the new_length is working as a pointer to indicate the position of the number that will be replaced with the current number. Set up the initial value of new_length to be 1.

When the current number is the same as the previous number, move to the next number, and when the current number is not the same as the previous number, replace the number at the new_length position with the current number and add new_length by 1. In the end, the program will return the value of new_length and the first new_length – 1 number from modified nums list.

Step 3: Convert ChatGPT's Python code into a flowchart.

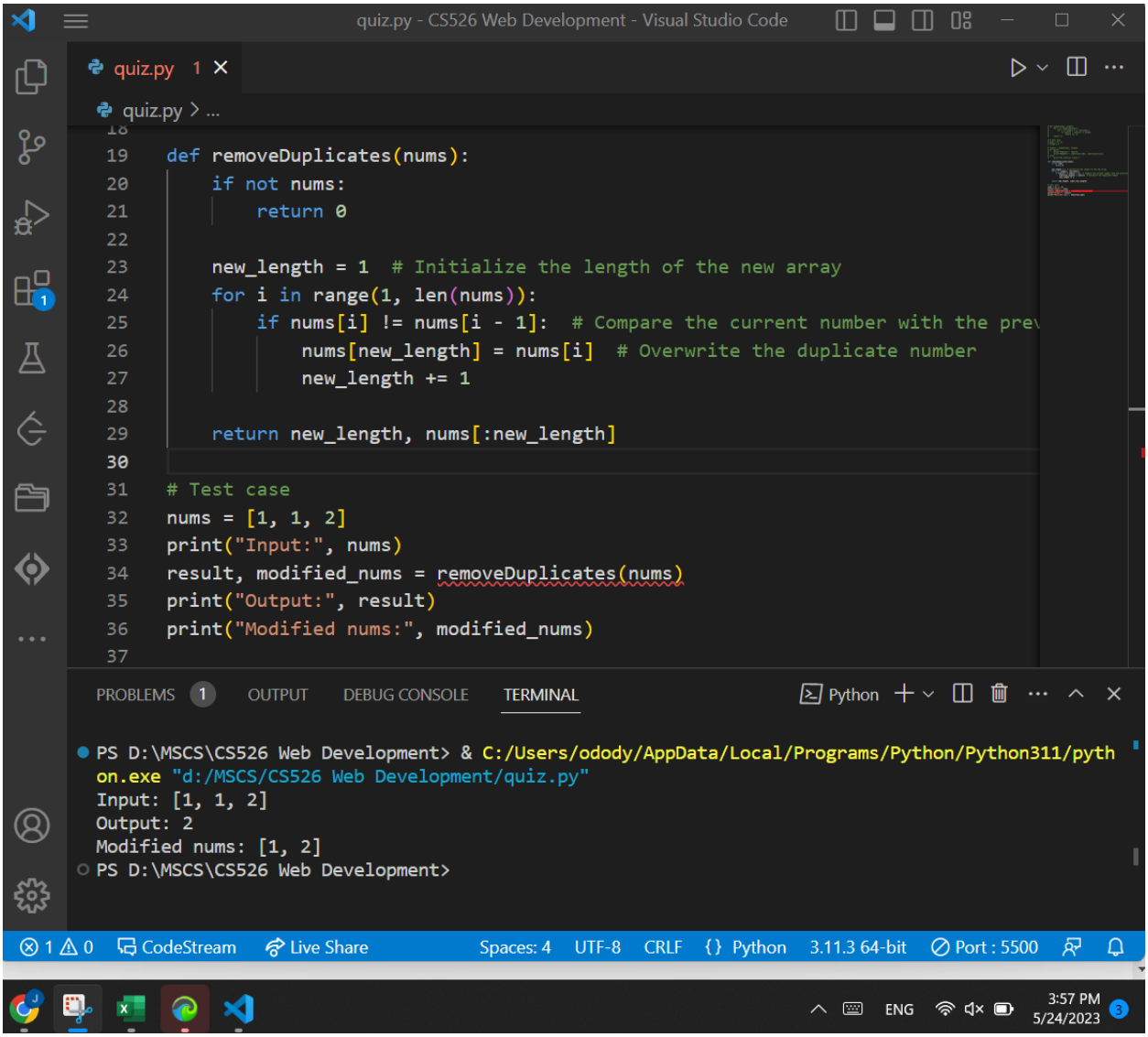


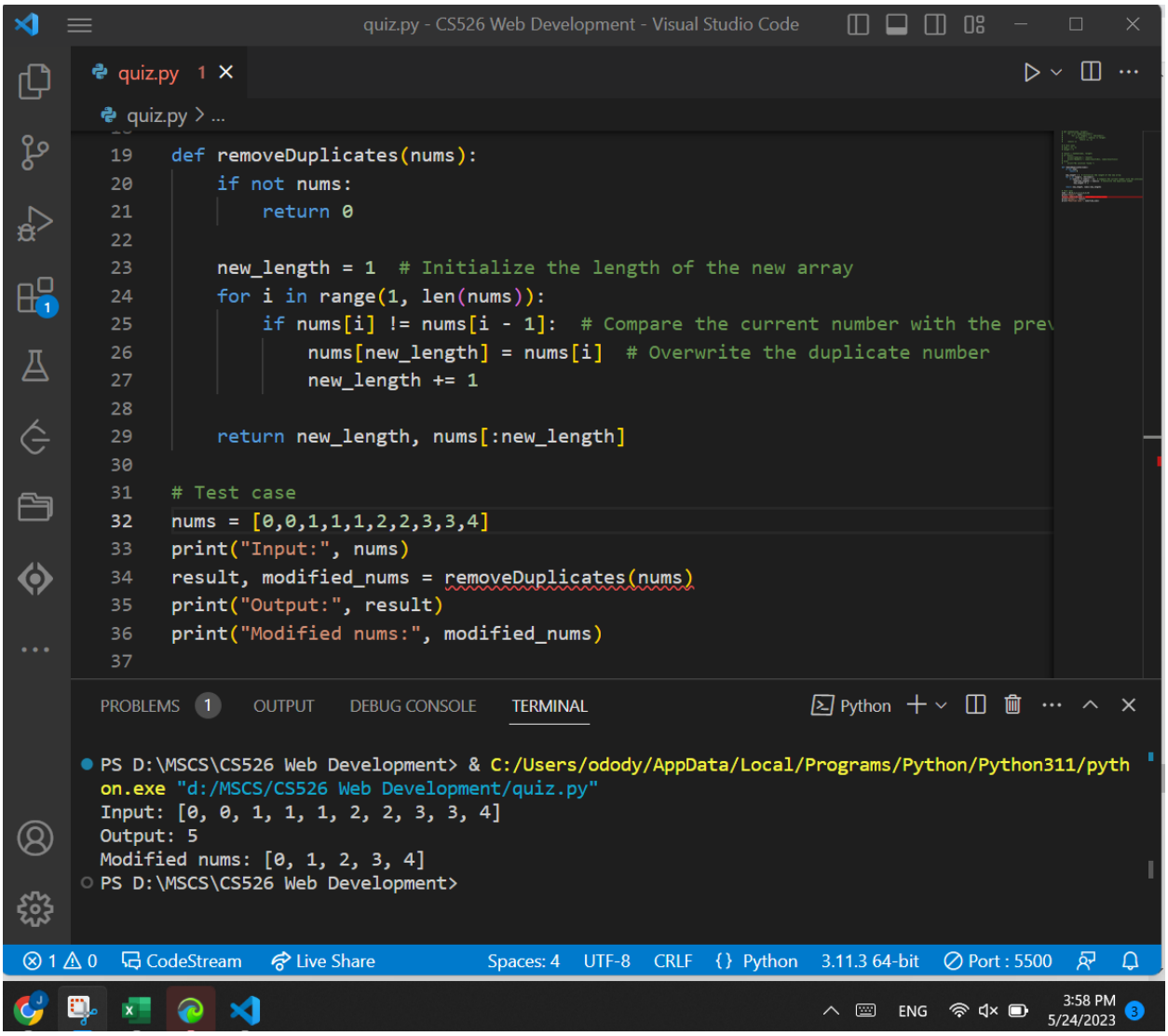


Step 4: Verify your flowchart with a trace table.

	RAM						screen
Step	i	if not nums	new_length	i < 3	nums[i] != nums[i-1]	nums	
1		F					
2			1				
3	1						
4				T			
5					F		
7	2						
4				T			
5					T		
6			2			[1, 2, 2]	
7	3						
4				F			
8							2, [1, 2]

Step 5: Using any Python IDE to test the Python code with the following test cases


quiz.py - CS526 Web Development - Visual Studio Code
quiz.py 1
quiz.py > ...
def removeDuplicates(nums):
if not nums:
return 0
new_length = 1 # Initialize the length of the new array
for i in range(1, len(nums)):
if nums[i] != nums[i - 1]: # Compare the current number with the prev
nums[new_length] = nums[i] # Overwrite the duplicate number
new_length += 1
return new_length, nums[:new_length]
# Test case
nums = [1, 1, 2]
print("Input:", nums)
result, modified_nums = removeDuplicates(nums)
print("Output:", result)
print("Modified nums:", modified_nums)
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL
Python
PS D:\MSCS\CS526 Web Development> & C:/Users/odody/AppData/Local/Programs/Python/Python311/python.exe "d:/MSCS/CS526 Web Development/quiz.py"
Input: [1, 1, 2]
Output: 2
Modified nums: [1, 2]
PS D:\MSCS\CS526 Web Development>
1 0
CodeStream
Live Share
Spaces: 4
UTF-8
CRLF
Python
3.11.3 64-bit
Port : 5500
ENG
3:57 PM
5/24/2023


quiz.py - CS526 Web Development - Visual Studio Code
quiz.py 1
quiz.py > ...
19 def removeDuplicates(nums):
20 if not nums:
21 return 0
22
23 new_length = 1 # Initialize the length of the new array
24 for i in range(1, len(nums)):
25 if nums[i] != nums[i - 1]: # Compare the current number with the prev
26 nums[new_length] = nums[i] # Overwrite the duplicate number
27 new_length += 1
28
29 return new_length, nums[:new_length]
30
31 # Test case
32 nums = [0,0,1,1,1,2,2,3,3,4]
33 print("Input:", nums)
34 result, modified_nums = removeDuplicates(nums)
35 print("Output:", result)
36 print("Modified nums:", modified_nums)
37
PROBLEMS 1
OUTPUT
DEBUG CONSOLE
TERMINAL
Python
PS D:\MSCS\CS526 Web Development> & C:/Users/odody/AppData/Local/Programs/Python/Python311/python.exe "d:/MSCS/CS526 Web Development/quiz.py"
Input: [0, 0, 1, 1, 1, 2, 2, 3, 3, 4]
Output: 5
Modified nums: [0, 1, 2, 3, 4]
PS D:\MSCS\CS526 Web Development>
1 0
CodeStream
Live Share
Spaces: 4
UTF-8
CRLF
Python
3.11.3 64-bit
Port : 5500
ENG
3:58 PM
5/24/2023