1. 131. Write a program that takes an input list of n numbers and creates a new list containing only the unique elements from the original list. What is the space complexity of the algorithm?
   Test Cases
   Some Duplicate Elements
      - Input: [3, 7, 3, 5, 2, 5, 9, 2]
      - Expected Output: [3, 7, 5, 2, 9] (Order may vary based on the algorithm used)

   Negative and Positive Numbers
      - Input: [-1, 2, -1, 3, 2, -2]
      - Expected Output: [-1, 2, 3, -2] (Order may vary)

   List with Large Numbers
   - Input: [1000000, 999999, 1000000]
   - Expected Output: [1000000, 999999]

PROGRAM:-

```
def unique_elements(nums):
    unique_set = set()
    unique_list = []

    for num in nums:
        if num not in unique_set:
            unique_set.add(num)
            unique_list.append(num)

    return unique_list

# Test Case 1: Some Duplicate Elements
input1 = [3, 7, 3, 5, 2, 5, 9, 2]
output1 = unique_elements(input1)
print(f"Input: {input1}\nExpected Output: [3, 7, 5, 2, 9] (Order may vary)\nOutput: {output1}\n")

# Test Case 2: Negative and Positive Numbers
input2 = [-1, 2, -1, 3, 2, -2]
```

output2 = unique_elements(input2)

print(f"Input: {input2}\nExpected Output: [-1, 2, 3, -2] (Order may vary)\nOutput: {output2}\n")

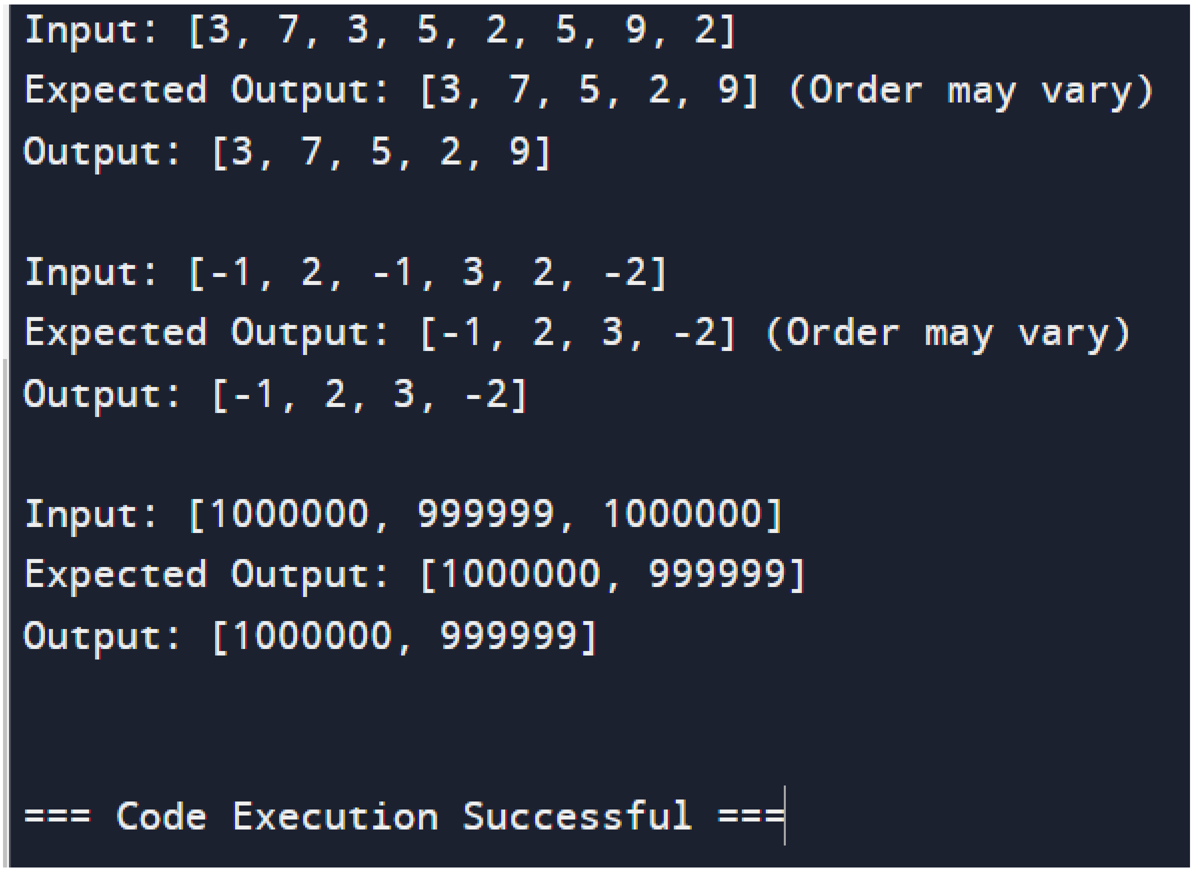
# Test Case 3: List with Large Numbers

input3 = [1000000, 999999, 1000000]

output3 = unique_elements(input3)

print(f"Input: {input3}\nExpected Output: [1000000, 999999]\nOutput: {output3}\n")

OUTPUT:-



```
Input: [3, 7, 3, 5, 2, 5, 9, 2]
Expected Output: [3, 7, 5, 2, 9] (Order may vary)
Output: [3, 7, 5, 2, 9]

Input: [-1, 2, -1, 3, 2, -2]
Expected Output: [-1, 2, 3, -2] (Order may vary)
Output: [-1, 2, 3, -2]

Input: [1000000, 999999, 1000000]
Expected Output: [1000000, 999999]
Output: [1000000, 999999]


=== Code Execution Successful ===
```

TIME COMPLEXITY:- O(n)