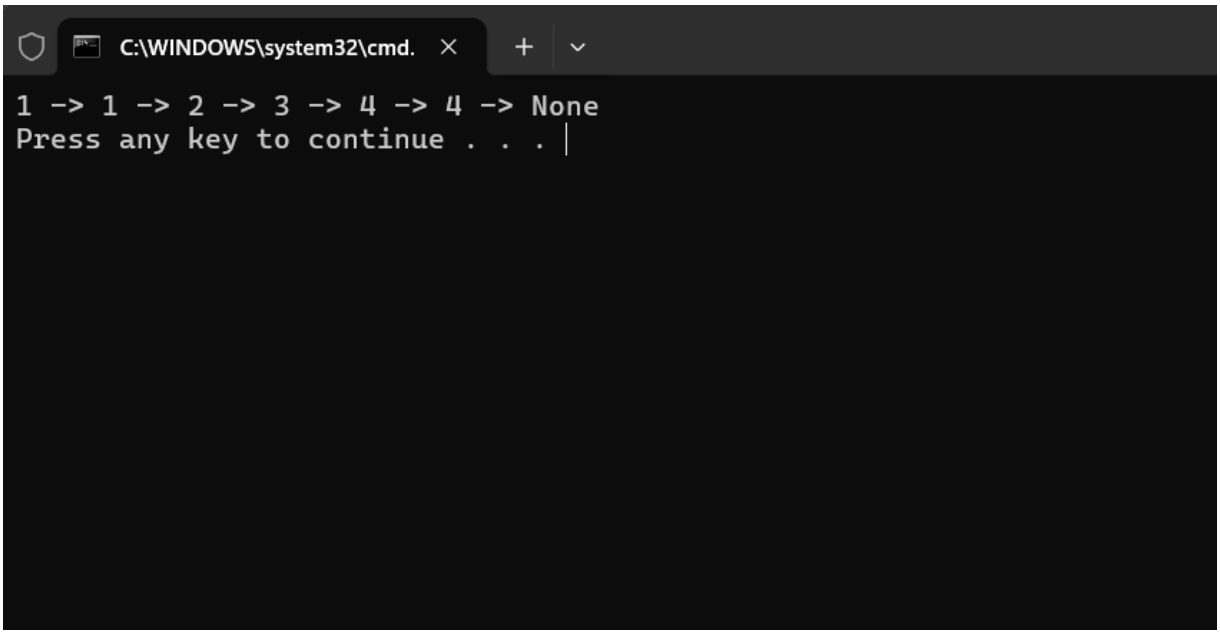
41) Merge Two Sorted Lists You are given the heads of two sorted linked lists list1 and list2. Merge the two lists in a one sorted list. The list should be made by splicing together the nodes of the first two lists. Return the head of the merged linked list.

CODE:

```
class ListNode:
    def __init__(self, val=0, next=None):
        self.val = val
        self.next = next

def mergeTwoLists(list1: ListNode, list2: ListNode) -> ListNode:
    # Create a dummy node to simplify edge cases
    dummy = ListNode()
    current = dummy

    # Traverse both lists
    while list1 and list2:
        if list1.val < list2.val:
            current.next = list1
            list1 = list1.next
        else:
            current.next = list2
            list2 = list2.next
        current = current.next

    # Attach the remaining part of list1 or list2
    if list1:
        current.next = list1
    else:
        current.next = list2

    # Return the merged list, which starts after the dummy node
    return dummy.next

# Helper function to create a linked list from a list
def create_linked_list(lst):
    dummy = ListNode()
    current = dummy
    for val in lst:
        current.next = ListNode(val)
        current = current.next
    return dummy.next

# Helper function to print a linked list
def print_linked_list(node):
    while node:
        print(node.val, end=" -> ")
        node = node.next
    print("None")

# Example usage:
list1 = create_linked_list([1, 2, 4])
list2 = create_linked_list([1, 3, 4])

merged_list = mergeTwoLists(list1, list2)
print_linked_list(merged_list)
```

OUTPUT:



TIME COMPLEXITY : O(n+m)