

Add 1 to a number represented as linked list

Number is represented in a linked list such that each digit corresponds to a node in linked list. Add 1 to it. For eg: 1999 is represented as $(1 \rightarrow 9 \rightarrow 9 \rightarrow 9)$ and adding 1 to it should change it to $(2 \rightarrow 0 \rightarrow 0 \rightarrow 0)$.

Approach:

Define the Node class to represent a node in the linked list. Each node has a data attribute to store the value and a next attribute to point to the next node.

Define the add_one function that takes the head of the linked list as input.

Create a dummy node with a value of 0 and set its next attribute to the head of the original linked list. This dummy node helps handle the case where the original number is all 9s.

Initialize a current variable to point to the dummy node.

Traverse the linked list to find the rightmost non-9 digit. This is done by iterating through the linked list while updating the current variable whenever a non-9 digit is encountered.

Add 1 to the value of the rightmost non-9 digit by incrementing current.data by 1.

Move current to the next node in the linked list.

Set all the following digits (nodes) to 0. This is done by iterating through the remaining nodes in the linked list and setting their data attribute to 0.

Check if the dummy node remains by comparing dummy.data with 0. If the dummy node remains, it means the original number was all 9s. In this case, return the next node after the dummy node as the new head of the modified linked list.

If the dummy node is removed, return the original head as the new head of the modified linked list.

Define the print_list function to print the linked list by traversing through the nodes and printing their values.

Create a test case by creating a linked list with values 1 -> 2 -> 9 -> 9.

Print the original linked list.

Call the `add_one` function with the head of the linked list to add 1 to the number represented by the linked list.

Print the modified linked list after adding 1.