Multiply Two Linked Lists

solved by	Senan
Platform	GeeksForGeeks
difficulty	Easy
tags	Linked List Maths
language	C++
solved on	@October 1, 2024
link	https://www.geeksforgeeks.org/problems/multiply-two-linked-lists/1
Completion	☑

Intuition

The goal is to multiply two large numbers that are represented as linked lists. Each node of the linked list represents a single digit of the number, and the order of nodes matches the order of digits in the number. To solve this problem, we can traverse the linked lists to extract the digits and build the numbers. Since the numbers could be very large, modular arithmetic (modulo $10^9 + 7$) is used to avoid overflow.

Approach

1. **Extracting the number**: We define a helper function `getNumber` that traverses the linked list, digit by digit, and constructs the number by multiplying the current result by 10 and adding the digit stored in the current node.
2. **Modular arithmetic**: To prevent overflow from potentially large numbers, we keep the result of the constructed number modulo $10^9 + 7$ at each step. This ensures that intermediate values stay manageable.
3. **Multiplying the two numbers**: Once we have extracted both numbers from the linked lists, we multiply them together, again taking the result modulo $10^9 + 7$.

Complexity

Time Complexity:

Traversing each linked list takes $O(size)$, where $size$ is the number of nodes in the list. Since we process each list separately, the total time complexity is: $O(n + m)$

Where n and m are the lengths of the two linked lists.

Space Complexity:

We are not using any additional space apart from a few variables to store numbers, so the space complexity is constant: $O(1)$

Code

```
class solution {
  private:
    int mod = 1e9 + 7;
```

```
    long long getNumber(Node* nd){
        long long num = 0;
        Node* temp = nd;

        while(temp){
            num = (10 * num + temp->data) % mod;
            temp = temp->next;
        }

        return num;
    }

  public:
    long long multiplyTwoLists(Node *first, Node *second) {
        long long num1 = getNumber(first);
        long long num2 = getNumber(second);
        return (num1 * num2) % mod;
    }
};
```