Thursday, January 20, 2022 5:11 PM

Multiply two linked lists

**Easy** Accuracy: 38.67% Submissions: 26091 Points: 2

Given elements as nodes of the two linked lists. The task is to multiply these two linked lists, say $L_1$ and $L_2$.

**Note:** The output could be large take modulo $10^9$+7.

**Input:**

The first line of input contains an integer T denoting the number of test cases. Then T test cases follow, the first line of each test case contains an integer **N** denoting the size of the first linked list (L1). In the next line are the space separated values of the first linked list. The third line of each test case contains an integer **M** denoting the size of the second linked list (L2). In the forth line are space separated values of the second linked list.

**Output:**

For each test case output will be an integer denoting the product of the two linked list.

**User Task**:

The task is to complete the function **multiplyTwoLists**() which should multiply the given two linked lists and return the result.

**Constraints:**

1 <= T <= 100

1 <= N, M <= 100

**Example:**

**Input:**

2

2

3 2

1

2

3

1 0 0

2

1 0

**Output:**

64

1000

**Explanation:**

**Testcase 1:** 32*2 = 64.

**Testcase 2:** 100*10 = 1000.

From <https://practice.geeksforgeeks.org/problems/multiply-two-linked-lists/1#>

```
import java.util.*;
public class MultiplyTwoList {


    public long multiplyTwoLists(Node l1, Node l2) {

        long num1=getNum(l1);
        long num2=getNum(l2);
         long N = 1000000007;

        long product=((num1%N)*(num2%N))%N;


        return product;
    }
    public long getNum(Node head){
 long N = 1000000007;
long num=0;
Node temp=head;
while(temp!=null)
{
    num=((num*10)%N)+temp.data;
    temp=temp.next;
}
return num;
    }



    public static void main(String[] args){
        Scanner sc=new Scanner(System.in);
        System.out.println(" Enter the T : ");
        int t=sc.nextInt();
        while(t-->0){
            Node head=null;
            Node phead=null;
            System.out.println("Enter the n : i.e length of the l1");
            int n=sc.nextInt();
            while(n-->0){
                System.out.println("enter the value of node n1 ");
                int n1=sc.nextInt();
                if(head==null)
                head=new Node(n1);
            else{
                addHead(head,n1);
            }
            }
            System.out.println("Enter the m : i.e length of the l2");
            int m=sc.nextInt();
            while(m-->0){
                System.out.println("enter the value of node m1 ");

                int n1=sc.nextInt();
                if(phead==null)
                phead=new Node(n1);
            else{
                addPhead(phead,n1);
            }
            }
        GfG g=new GfG();
System.out.println(g.multiplyTwoLists(head,phead));
        }
    }
    public static void addHead(Node node,int a) {
          if (node == null)
            return;
        if(node.next==null)
            node.next=new Node(a);
        else
            addHead(node.next,a);
    }
    public static void addPhead(Node node,int a){
         if (node == null)
            return;
        if(node.next==null)
        node.next=new Node(a);
    else
        addPhead(node.next,a);
    }
}
class Node
{
    int data;
    Node next;
    Node(int data) {
        this.data=data;
        this.next = null;
    }
}
class GfG{
  /*You are required to complete this method */

    public long multiplyTwoLists(Node l1, Node l2) {

        int num1=getNum(l1);
        int num2=getNum(l2);
        long product=num1*num2;



        return product;

    }

    public int getNum(Node head){
        if (head==null) {
            return 0;

        }
        String num="";

        Node temp=head;
        while(temp!=null)
        {
            num+=Integer.toString(temp.data);
            temp=temp.next;

        }
        // System.out.println("String Num : "+num);
        int number =Integer.parseInt(num);


        return number;

    }
}
```