### Reverse a Linked List in groups of given size

Given a linked list of size **N**. The task is to reverse every **k** nodes (where k is an input to the function) in the linked list. If the number of nodes is not a multiple of *k* then left-out nodes, in the end, should be considered as a group and must be reversed (See Example 2 for clarification).

**Example 1:**

**Input:**

LinkedList: 1->2->2->4->5->6->7->8

K = 4

**Output:** 4 2 2 1 8 7 6 5

**Explanation:**

The first 4 elements 1,2,2,4 are reversed first

and then the next 4 elements 5,6,7,8. Hence, the

resultant linked list is 4->2->2->1->8->7->6->5.

**Example 2:**

**Input:**

LinkedList: 1->2->3->4->5

K = 3

**Output:** 3 2 1 5 4

**Explanation:**

The first 3 elements are 1,2,3 are reversed

first and then elements 4,5 are reversed.Hence,

the resultant linked list is 3->2->1->5->4.

**Your Task:**  
You don't need to read input or print anything. Your task is to complete the function **reverse**() which should reverse the linked list in group of size **k**and return the head of the modified linked list.

**Expected Time Complexity**: O(N)  
**Expected Auxilliary Space**: O(1)

**Constraints:**

1 <= N <= 105  
1 <= k <= N

**Company Tags**

[**Paytm**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Paytm) [**VMWare**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=VMWare) [**Accolite**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Accolite) [**Amazon**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Amazon) [**Microsoft**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Microsoft) [**Snapdeal**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Snapdeal) [**Hike**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Hike) [**MakeMyTrip**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=MakeMyTrip) [**Walmart**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Walmart) [**Goldman Sachs**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Goldman%20Sachs) [**Adobe**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=Adobe) [**SAP Labs**](https://practice.geeksforgeeks.org/explore/?company%5b%5d=SAP%20Labs)

import java.util.\*;

import java.lang.\*;

class Node

{

int data;

Node next;

Node(int key)

{

data = key;

next = null;

}

}

class ReverseInSize

{

static Node head;

public static void main (String[] args) {

Scanner sc = new Scanner(System.in);

int t = sc.nextInt();

while(t-- > 0)

{

int n = sc.nextInt();

int a1 = sc.nextInt();

Node head = new Node(a1);

Node tail = head;

for(int i = 1; i < n; i++)

{

int a = sc.nextInt();

// addToTheLast(new Node(a));

tail.next = new Node(a);

tail =tail.next;

}

int k = sc.nextInt();

Solution ob = new Solution();

Node res = ob.reverse(head, k);

printList(res);

System.out.println();

}

}

public static void printList(Node node)

{

while(node != null)

{

System.out.print(node.data + " ");

node = node.next;

}

}

}

class Solution

{

public static Node reverse(Node head, int k)

{

ArrayList<Integer> ar=new ArrayList<>();

while(head!=null){

ar.add(head.data);

head=head.next;

}

int n=ar.size();

// int i=0;

for (int i = 0; i < n; i += k)

{

int left = i;

int right =Math.min(i + k - 1, n-1);

int temp;

while (left < right)

{

temp=ar.get(left);

ar.set(left,ar.get(right));

ar.set(right, temp);

left+=1;

right-=1;

}

}

Node dummy=new Node(-1);

Node curr=dummy;

for(int num : ar){

curr.next=new Node(num);

curr=curr.next;

}

return dummy.next;

}

}