```
package linkedlist;

// Java program to reverse a linked list in groups of
// given size
class LinkedList2
{
    Node head; // head of list

    /* Linked list Node*/
    class Node
    {
        int data;
        Node next;
        Node(int d) {data = d; next = null; }
    }

    Node reverse(Node head, int k)
    {
        Node current = head;
        Node next = null;
        Node prev = null;

        int count = 0;

        /* Reverse first k nodes of linked list */
        while (count < k && current != null)
        {
            next = current.next;
            current.next = prev;
            prev = current;
            current = next;
            count++;
        }

    /* next is now a pointer to (k+1)th node
        Recursively call for the list starting from current.
        And make rest of the list as next of first node */
        if (next != null)
            head.next = reverse(next, k);

        // prev is now head of input list
        return prev;
    }


    /* Utility functions */

    /* Inserts a new Node at front of the list. */
    public void push(int new_data)
    {
        /* 1 & 2: Allocate the Node &
                  Put in the data*/
        Node new_node = new Node(new_data);

        /* 3. Make next of new Node as head */
        new_node.next = head;

        /* 4. Move the head to point to new Node */
        head = new_node;
    }

    /* Function to print linked list */
    void printList()
    {
        Node temp = head;
        while (temp != null)
```

```
        {
            System.out.print(temp.data+" ");
            temp = temp.next;
        }
        System.out.println();
    }

    /* Driver program to test above functions */
    public static void main(String args[])
    {
        LinkedList2 llist = new LinkedList2();

        /* Constructed Linked List is 1->2->3->4->5->6->
        7->8->8->9->null */
        llist.push(9);
        llist.push(8);
        llist.push(7);
        llist.push(6);
        llist.push(5);
        llist.push(4);
        llist.push(3);
        llist.push(2);
        llist.push(1);

        System.out.println("Given Linked List");
        llist.printList();

        llist.head = llist.reverse(llist.head, 3);

        System.out.println("Reversed list");
        llist.printList();
    }
}
/* This code is contributed by Rajat Mishra */
```