

```

class ListNode {
    ListNode next;
    int val;
    public ListNode(int v) {
        val = v;
    }
}

public class InsertValueToCircularLL {
    public static ListNode insert(ListNode root, int n) {
        if (root == null) {
            root = new ListNode(n);
            return root;
        } else if (root.next == null) {
            root.next = new ListNode(n);
            root.next.next = root;
            return root.val > root.next.val ? root.next : root;
        } else {
            if (n < root.val) {
                ListNode itr = root;
                ListNode node = new ListNode(n);
                while (itr.next != root) {
                    itr = itr.next;
                }

                node.next = root;
                itr.next = node;
                return node;
            } else {
                ListNode itr = root;
                ListNode node = new ListNode(n);

                while (itr.next.val <= n && itr.next != root) {
                    itr = itr.next;
                }

                node.next = itr.next;
                itr.next = node;
                return root;
            }
        }
    }
}

```