```java
package LinkedList;

class JudgeTwoLinkedListIntersection {

    public boolean judgeTwoLinkedListIntersection(Node headA, Node headB) {

        if ( headA == null || headB == null ) {
            return false;
        }

        Node tempA = headA;
        int lengthA = 0;
        while ( tempA.next != null ) {

            lengthA++;
            tempA = tempA.next;
        }

        Node tempB = headB;
        int lengthB = 0;
        while ( tempB.next != null ) {

            lengthB++;
            tempB = tempB.next;
        }

        // 如果只需要判断两个链表是否有交点，那么到这步就结束了；
        // 如果还需要判断交点的位置，就需要下面的步骤
        if ( tempA != tempB ) {
            return false;
        }

        Node startNodeA = headA, startNodeB = headB;
        if ( lengthA >= lengthB ) {
            for ( int i = 0; i < lengthA - lengthB; i++ ) {
                startNodeA = startNodeA.next;
            }
        } else {
            for ( int i = 0; i < lengthB - lengthA; i++ ) {
                startNodeB = startNodeB.next;
            }
        }

        while ( startNodeA != null ) {

            if ( startNodeA == startNodeB ) {
                return true;
            }

            startNodeA = startNodeA.next;
            startNodeB = startNodeB.next;
        }

        return false;
    }
}
```