## Question:

You’re given the pointer to the head nodes of two linked lists. Compare the data in the nodes of the linked lists to check if they are equal. If all data attributes are equal and the lists are the same length, return 1. Otherwise, return 0.

**Example**



The two lists have equal data attributes for the first 3 nodes. llist  is longer, though, so the lists are not equal.

Return.

**Function Description**

Complete the *compare\_lists* function in the editor below.

*compare\_lists* has the following parameters:

* *SinglyLinkedListNode llist1:* a reference to the head of a list
* *SinglyLinkedListNode llist2:* a reference to the head of a list

**Returns**

* *int:* return 1 if the lists are equal, or 0 otherwise

|  |
| --- |
| /\*Insert Node at the end of a linked listhead pointer input could be NULL as well for empty listNode is defined asclass Node {int data;Node next;}\*/int CompareLists(Node headA, Node headB) {// This is a "method-only" submission.// You only need to complete this methodwhile (headA != null && headB != null) {if (headA.data != headB.data)return 0;headA = headA.next;headB = headB.next;}if(headA==null && headB!=null)return 0;if(headA!=null && headB==null)return 0;return 1;} |

Output**:**

**Sample Input**

2

2

1

2

1

1

2

1

2

2

1

2

**Sample Output**

0

1