

```

1.  /**
2.   * Definition for singly-linked list.
3.   * class ListNode {
4.   *     public int val;
5.   *     public ListNode next;
6.   *     ListNode(int x) { val = x; next = null; }
7.   * }
8.   */
9.  public class Solution {
10.     public int IPalin(ListNode A) {
11.         int size=0;
12.         ListNode t=A;
13.         int mid=0;
14.         while(t!=null){
15.             size++;
16.             t=t.next;
17.         }
18.         if(size%2==0)
19.             mid=size/2;
20.         else
21.             mid=(size/2)+1;
22.
23.         t=A;
24.         for(int i=1;i<mid;i++)
25.         {
26.             t=t.next;
27.         }
28.         ListNode newH=t.next;
29.         ListNode t1=reverseList(newH);
30.         t=A;
31.         int flag=1;
32.         if(size%2==0)
33.         {
34.             for(int i=1;i<=mid;i++)
35.             {
36.                 if(t.val!=t1.val){
37.                     flag=0;
38.                     break;
39.                 }
40.                 t=t.next;
41.                 t1=t1.next;
42.             }
43.             return flag;
44.         }
45.         else{
46.             for(int i=1;i<mid;i++)
47.             {
48.                 if(t.val!=t1.val){
49.                     flag=0;
50.                     break;
51.                 }
52.                 t=t.next;

```

```
53.         t1=t1.next;
54.     }
55.     return flag;
56. }
57.
58.
59. }
60. public ListNode reverseList(ListNode h) {
61.     ListNode curr=h;
62.     ListNode previous=null;
63.     ListNode nex=null;
64.     while(curr!=null) {
65.         nex=curr.next;
66.         curr.next=previous;
67.         previous=curr;
68.         curr=nex;
69.     }
70.     return previous;
71. }
72. }
```

Problem Link: [palindrome-list-InterviewBit](#)
Tutorial Link: [Check if palindrome-take U forward](#)