

```

1  /*Author: Bochen (mddboc@foxmail.com)
2  Last Modified: Tue Apr 10 22:28:44 CST 2018*/
3
4  /*Given a linked list, determine if it has a cycle in it.
5
6  .....Follow up:
7  .....Can you solve it without using extra space?*/
8
9
10 import java.util.*;
11 import java.lang.Math;
12 import java.lang.System;
13 import java.lang.Integer;
14
15
16 public class Main {
17
18     ....public static void main(String[] args) throws ArithmeticException {
19
20         .....TreeNode root = new TreeNode(1);
21         .....root.left = new TreeNode(2);
22         .....root.right = new TreeNode(2);
23         .....root.left.left = new TreeNode(3);
24         .....root.left.right = new TreeNode(4);
25         .....root.right.left = new TreeNode(4);
26         .....root.right.right = new TreeNode(3);
27
28         .....boolean result = new Solution().isSymmetric(root);
29
30         .....System.out.println(result);
31     }
32
33 }
34
35
36 class ListNode {
37     ....int val;
38     ....ListNode next;
39
40     ....ListNode(int x) {
41         .....val = x;
42     }
43 }
44
45
46 class TreeNode {
47     ....int val;
48     ....TreeNode left;
49     ....TreeNode right;
50
51     ....TreeNode(int x) {
52         .....val = x;
53     }
54 }
55
56
57 class Solution {
58     ....public boolean hasCycle(ListNode head) {
59
60         .....if (head == null) {
61             .....return false;
62         }
63
64         .....ListNode slowPointer = head, fastPointer = head;
65
66         .....while (fastPointer != null && fastPointer.next != null) {
67
68             .....fastPointer = fastPointer.next.next;
69             .....slowPointer = slowPointer.next;
70             —————>—————>—————>
71             .....if (fastPointer == slowPointer) {
72                 .....return true;
73             }

```

```
74 .....}
75
76 .....return false;
77 .....}
78 }
```