

3. Linked List Cycle

(LeetCode 3)

code:

class Solution {

public:

bool hasCycle(ListNode *head) {
 if (head == NULL) return false;

ListNode *slow = head;

ListNode *fast = head;

while (fast != NULL && fast->next != NULL)

slow = slow->next;

fast = fast->next->next;

if (slow == fast) {
 return true;

}

}

return true;

}

}

return false;

}

};

Output:

case 1

Input:

head =

[3, 2, 0, -4]

pos =

1

Output:

true

Expected:

true

case 3

Input:

head =

[1]

pos =

-1

Output:

false

Expected:

false

case 2

Input:

head =

[1, 2]

pos =

0

Output:

true

Expected:

true