#15 Days of Coding Challenge
#Day 12

Palindrome linked list
#Leetcode 234

Solution:

```
#include<iostream>
class Solution {
public:
bool isPalindrome(ListNode* head) {
  ListNode* slow = head;
  ListNode* fast = head;
  while (fast != nullptr && fast->next != nullptr) {
    slow = slow->next;
    fast = fast->next->next;
  }
  if (fast != nullptr)
    slow = slow->next;
  slow = reverseList(slow);
  while (slow != nullptr) {
    if (slow->val != head->val)
      return false;
    slow = slow->next;
    head = head->next;
  }
  return true;
}
private:
ListNode* reverseList(ListNode* head) {
  ListNode* prev = nullptr;
  while (head != nullptr) {
    ListNode* next = head->next;
    head->next = prev;
    prev = head;
    head = next;
  }
  return prev;
}
};
```

Time Complexity: O(L)

Space Complexity: O(1)