

## 24. Swap Nodes in Pairs

Given a linked list, swap every two adjacent nodes and return its head.

You may not modify the values in the list's nodes, only nodes itself may be changed.

### Example:

Given 1->2->3->4, you should return the list as 2->1->4->3.

- pointer 兩兩交換，存前一組的結尾 (prev) ，再接到當下這組

```
1  /**
2   * Definition for singly-linked list.
3   * struct ListNode {
4   *     int val;
5   *     struct ListNode *next;
6   * };
7   */
8
9  struct ListNode* swapPairs(struct ListNode* head)
10 {
11     if (head == NULL)
12         return NULL;
13     struct ListNode *cur = head;
14     struct ListNode *prev = head, *next;
15
16     if (cur->next != NULL)
17         head = cur->next;
18
19     while (cur != NULL && cur->next != NULL) {
20         prev->next = cur->next;
21
22         next = cur->next->next;
23         cur->next->next = cur;
24         cur->next = next;
25
26         prev = cur;
27         cur = cur->next;
28     }
29
30     return head;
31 }
```