

```
1.  /*
2.  struct ListNode {
3.      int val;
4.      struct ListNode *next;
5.      ListNode(int x) :
6.          val(x), next(NULL) {
7.      }
8.  };*/
9.  class Solution {
10. public:
11.     ListNode* ReverseList(ListNode* pHead)
12.     {
13.         // 反转指针
14.         ListNode* pNode=pHead; // 当前节点
15.         ListNode* pPrev=nullptr; // 当前节点的上一个节点
16.         ListNode* pNext=nullptr; // 当前节点的下一个节点
17.         ListNode* pReverseHead=nullptr; //新链表的头指针
18.
19.         // 反转链表
20.         while(pNode!=nullptr)
21.         {
22.             pNext=pNode->next; // 建立链接
23.
24.             if(pNext==NULL) // 判断 pNode 是否是最后一个节点
25.                 pReverseHead=pNode;
26.
27.             pNode->next=pPrev; // 指针反转
28.             pPrev=pNode;
29.             pNode=pNext;
30.         }
31.         return pReverseHead;
32.     }
33. };
```