# 题目

给定一个排序链表，删除所有重复的元素，使得每个元素只出现一次。

**示例 1:**

输入: 1->1->2

输出: 1->2

**示例 2:**

输入: 1->1->2->3->3

输出: 1->2->3

# 方法一：排序

# 方法二：快慢指针

**思路：**

删除重复的与判断环形（即是否存在两个相等的节点）是同一类问题，这里可以使用快慢指针的方法。

**代码：**

/\*\*

\* Definition for singly-linked list.

\* struct ListNode {

\* int val;

\* ListNode \*next;

\* ListNode(int x) : val(x), next(NULL) {}

\* };

\*/

class Solution {

public:

ListNode\* deleteDuplicates(ListNode\* head) {

ListNode \*dummy = new ListNode(INT\_MAX);

dummy->next = head;

ListNode \*p = dummy;

ListNode \*tmp;

while(p->next)

{

if(p->val == p->next->val)

{

tmp = p->next;

p->next = tmp->next;

delete tmp;

}

else

{

p = p->next;

}

}

return dummy->next;

}

};