**Delete Middle of Linked List**

Given a singly linked list, delete **middle**of the linked list. For example, if given linked list is 1->2->**3**->4->5 then linked list should be modified to 1->2->4->5.  
If there are **even** nodes, then there would be **two middle**nodes, we need to delete the **second middle element**. For example, if given linked list is 1->2->**3**->**4**->5->6 then it should be modified to 1->2->3->5->6.  
If the input linked list has **single** node, then it should return **NULL.**

**Example 1:**

**Input:**

LinkedList: 1->2->3->4->5

**Output:**1 2 4 5

**Example 2:**

**Input:**

LinkedList: 2->4->6->7->5->1

**Output:**2 4 6 5 1

**Your Task:**  
The task is to complete the function **deleteMid**() which takes head of the linkedlist  and return head of the linkedlist with **middle element deleted** from the linked list. If the linked list is **empty** or contains **single** element then it should return **NULL**.

**Expected Time Complexity:**O(n).  
**Expected Auxiliary Space:**O(1).

**Constraints:**  
1 <= n <= 105  
1 <= value[i] <= 109

class Solution{

public:

Node\* deleteMid(Node\* head)

{

if(head==NULL || head->next==NULL) return NULL;

Node\* prev =NULL;

Node\* slow =head;

Node\* fast =head;

while(fast->next && fast->next->next){

prev =slow;

slow = slow->next;

fast =fast->next->next;

}

if(fast->next){

prev =slow;

slow =slow->next;

}

prev->next =slow->next;

slow->next=NULL;

return head;

}

};

Link : <https://www.geeksforgeeks.org/problems/delete-middle-of-linked-list/1>