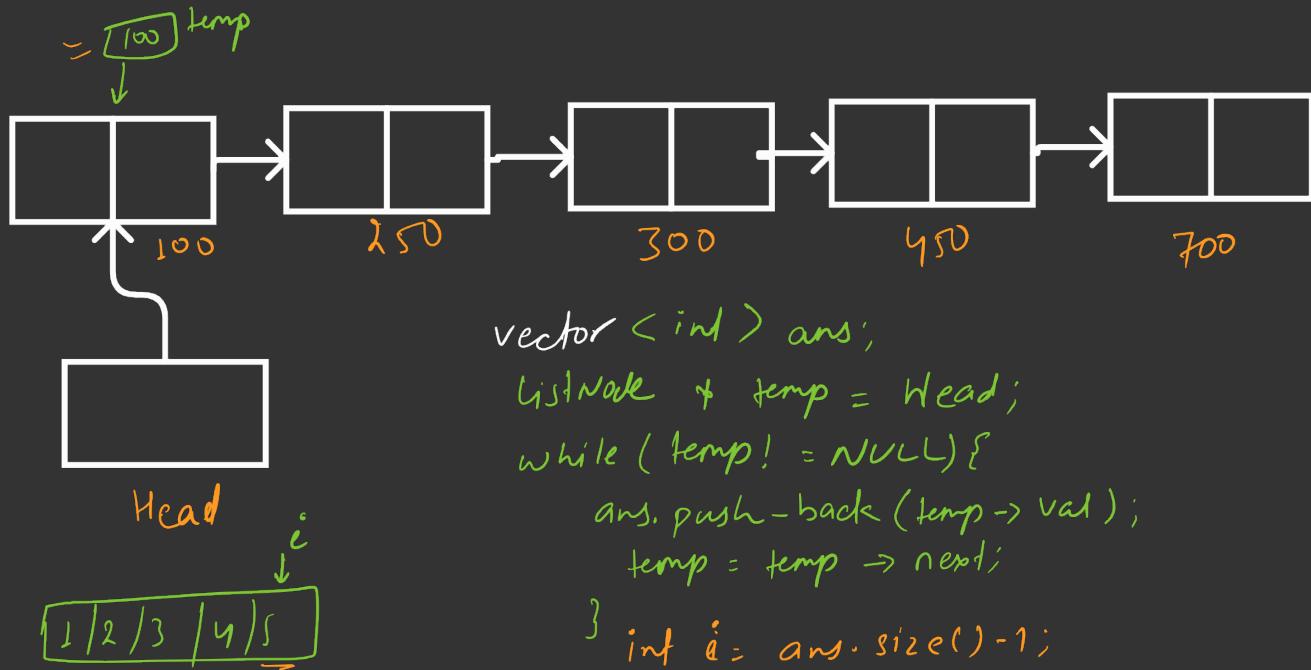
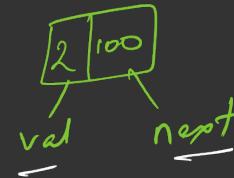


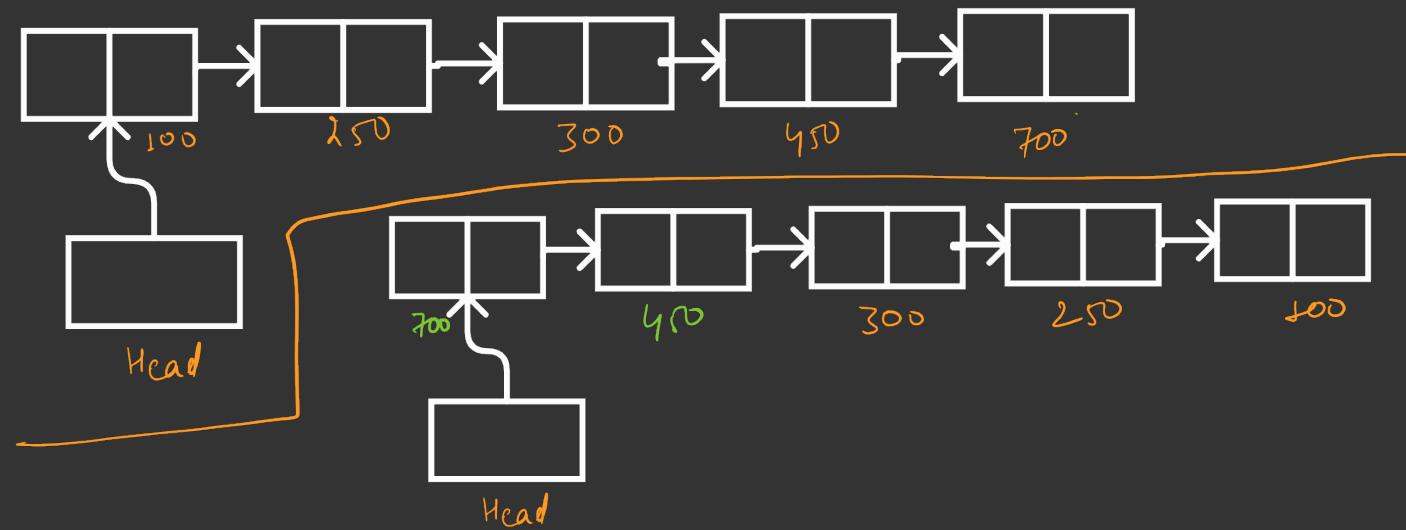
Reverse a Linked List

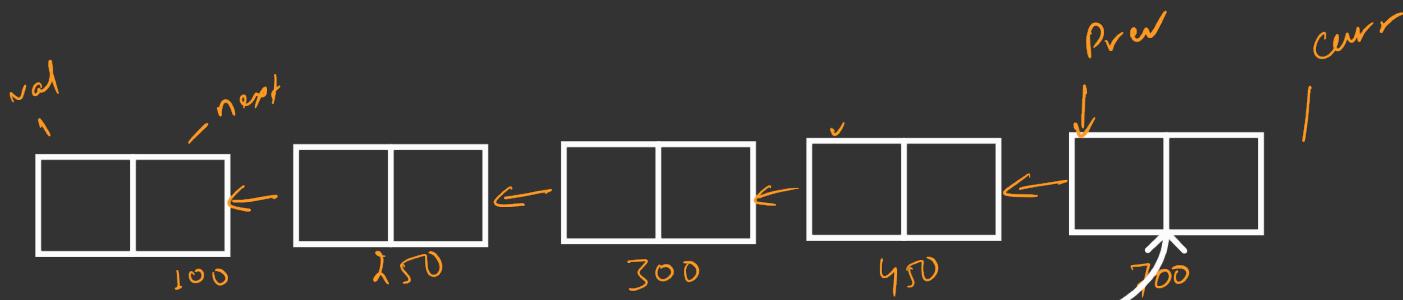


```
vector<int> ans;  
listNode * temp = Head;  
while (temp != NULL){  
    ans.push_back(temp->val);  
    temp = temp->next;  
}  
int i = ans.size() - 1;  
temp = Head;  
while (temp){  
    temp->val = ans[i];  
    i--;  
    temp = temp->next;  
}
```

```
class ListNode {
public:
    int val;
    ListNode* next;
    ListNode (int data) {
        val = data;
        next = NULL;
    }
};
```







Head

$\text{ListNode}^* \text{curr} = \text{head}$, $^*\text{prev} = \text{NULL}$
 $^*\text{fut} = \text{NULL}$

while (curr) {

$\text{fut} = \text{curr} \rightarrow \text{next};$

$\text{curr} \rightarrow \text{next} = \text{prev};$

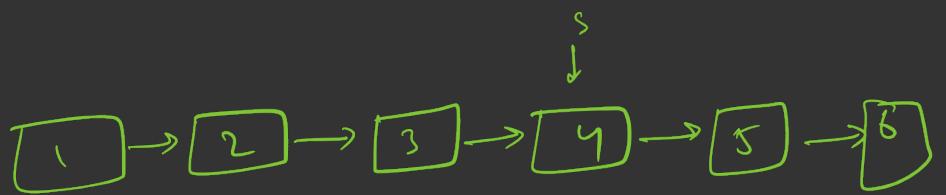
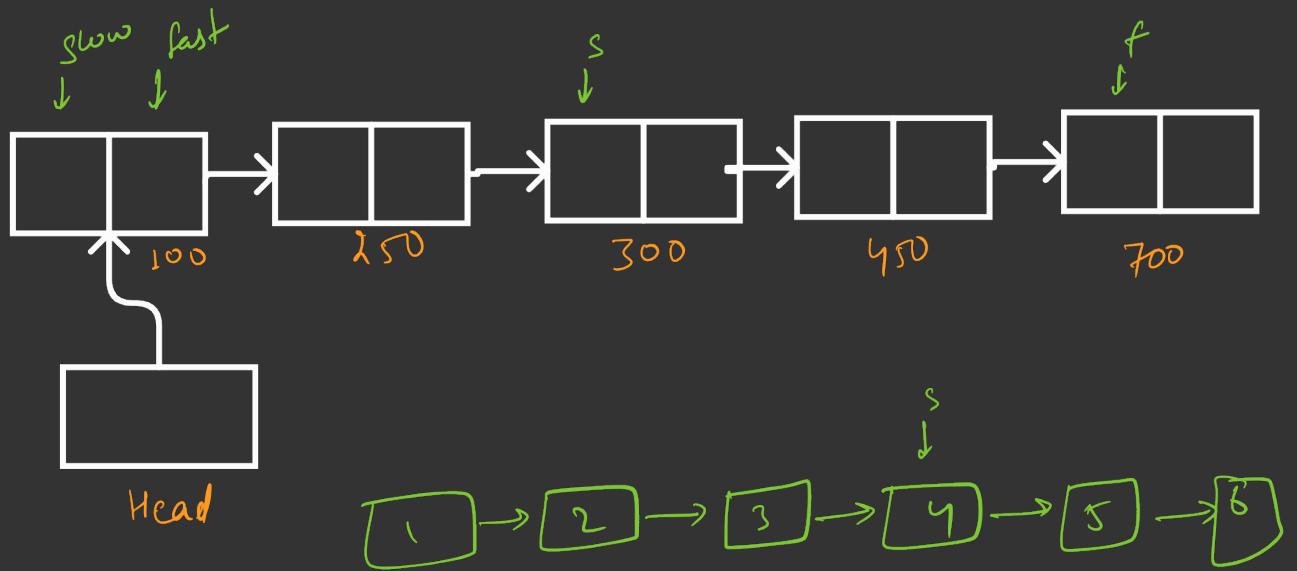
$\text{prev} = \text{curr};$

$\text{curr} = \text{fut};$

}

$\text{Head} = \text{prev};$

Middle of List



fast = null;
fast->next = null;

