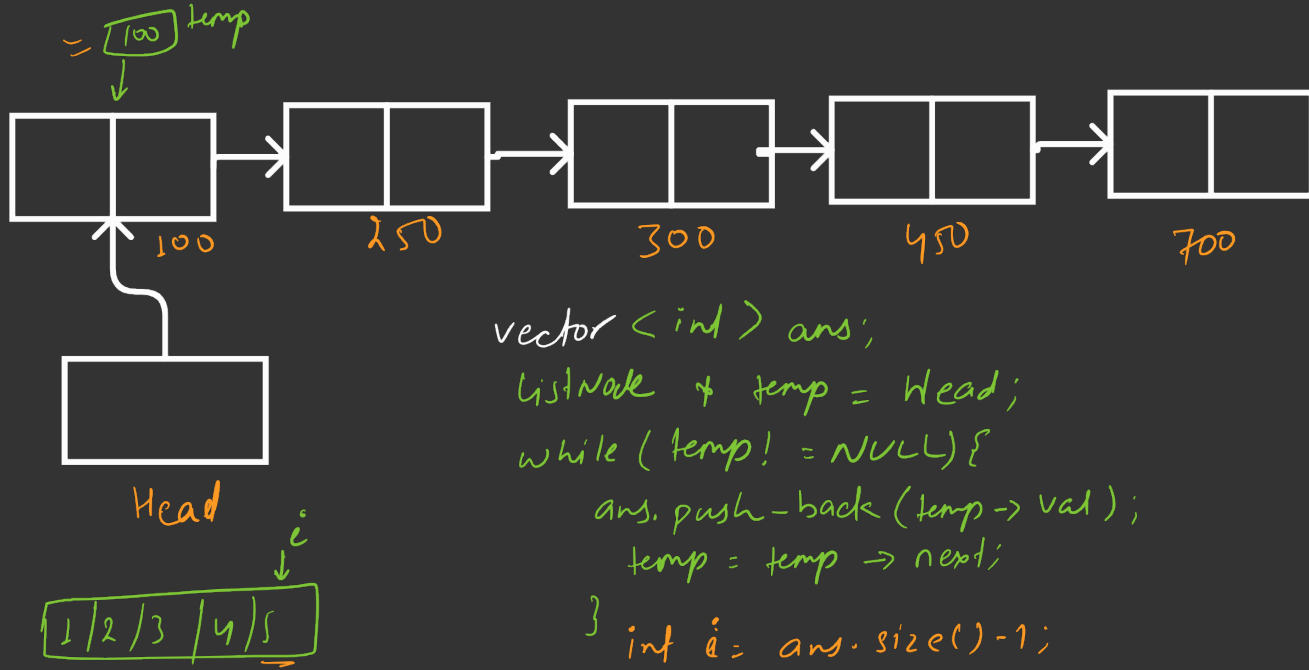


## Reverse a Linked List



```
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;
```

```
        List node* next;
```

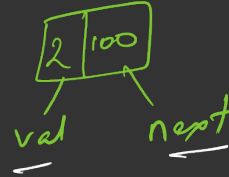
```
        ListNode (int data) {
```

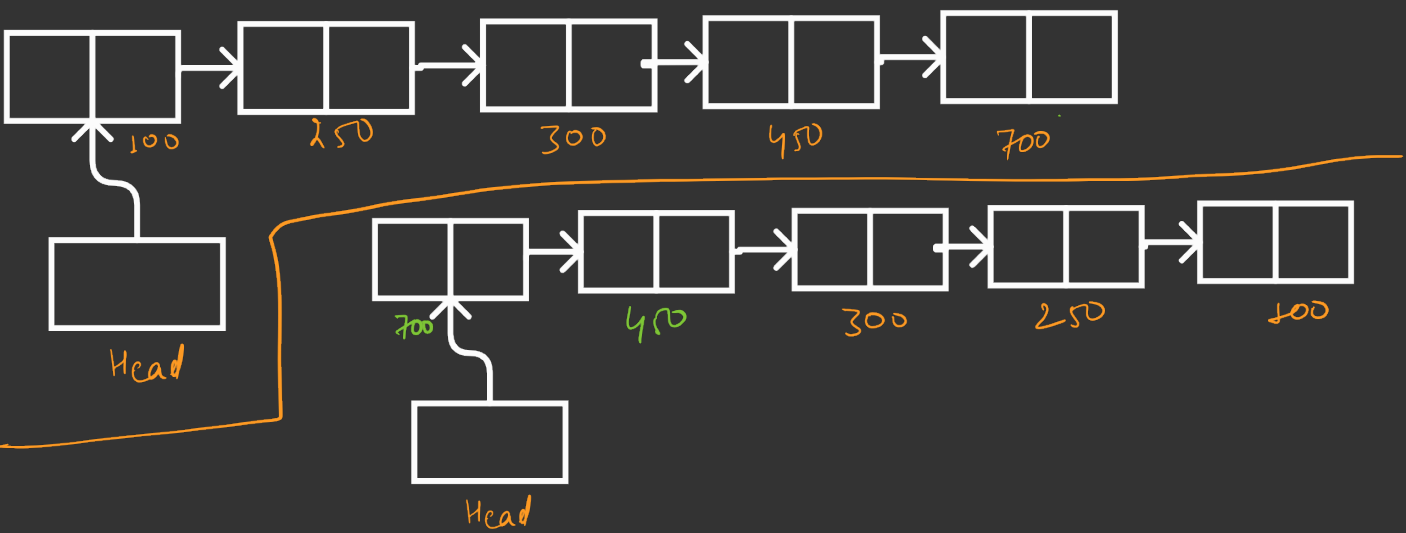
```
            val = data;
```

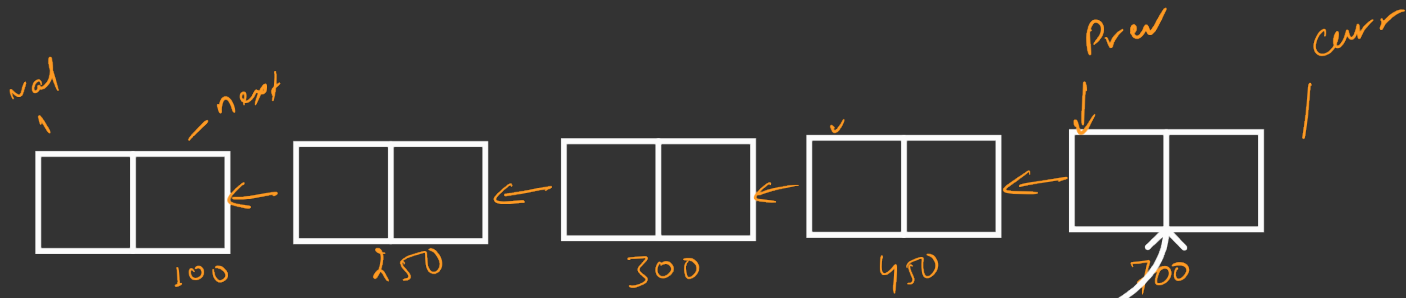
```
            next = NULL;
```

```
        }
```

```
    };
```







Head

```
ListNode *curr = head, *prev = NULL
*next = NULL
while (curr) {
```

```
    fut = curr -> next;
    curr -> next = prev;
```

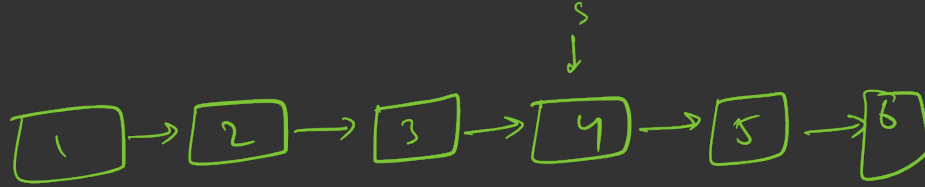
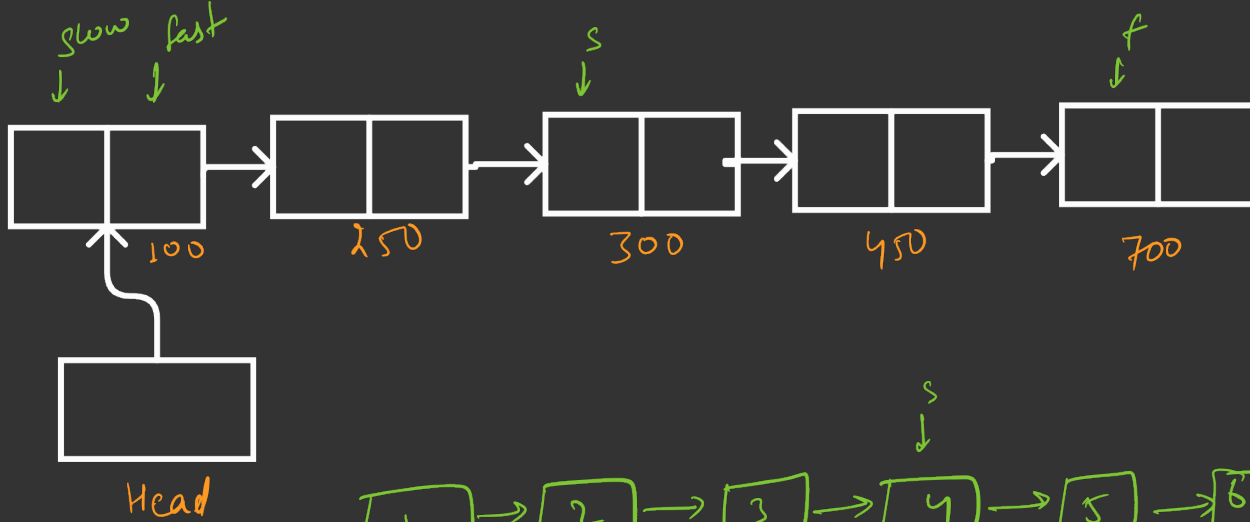
```
    prev = curr;
```

```
    curr = fut;
```

```
}
```

```
Head = prev;
```

## Middle of list



$\text{fast} = \text{null};$   
 $\text{fast} \rightarrow \text{next} = \text{null};$

$\uparrow$   
 $f$











