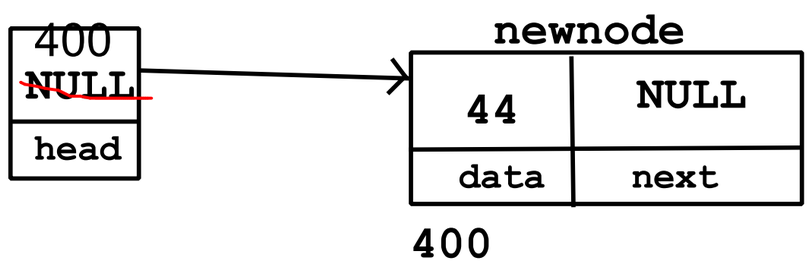
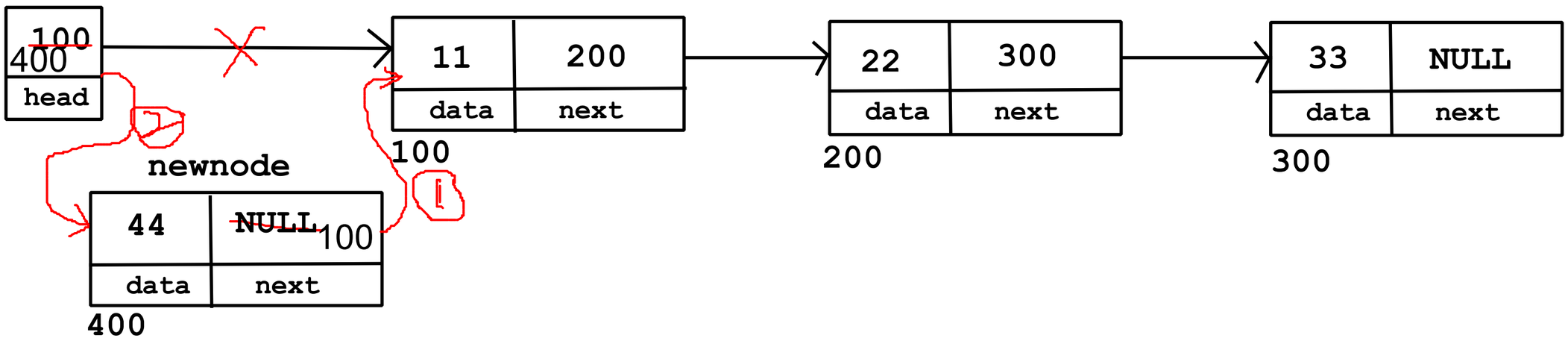
Linked List

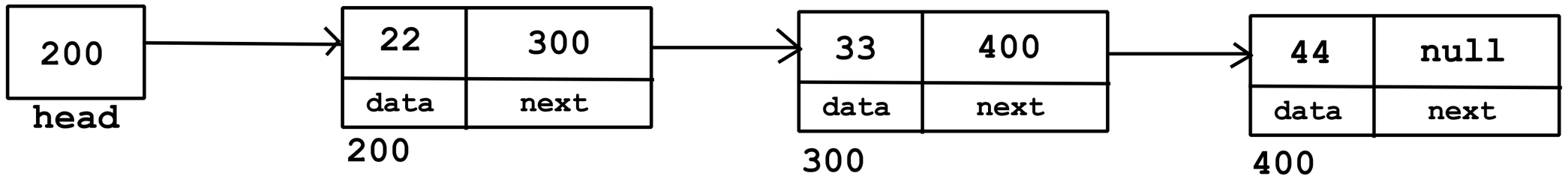
Singly Linear Linked List - Add First



//0. create node with given data
//1. if list is empty
    //a. add newnode into head
//2. if list is not empty
    //a. add head into newnode next
    //b. add newnode into head

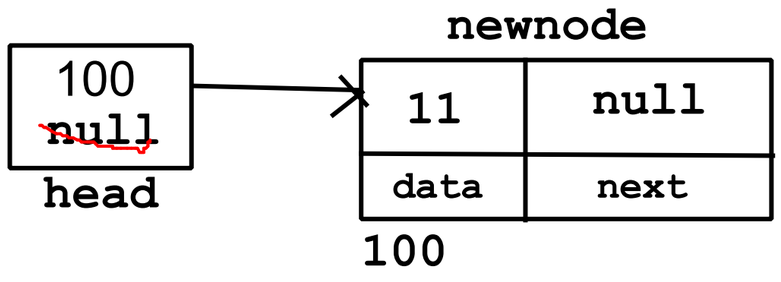


Singly Linear Linked List - Traverse



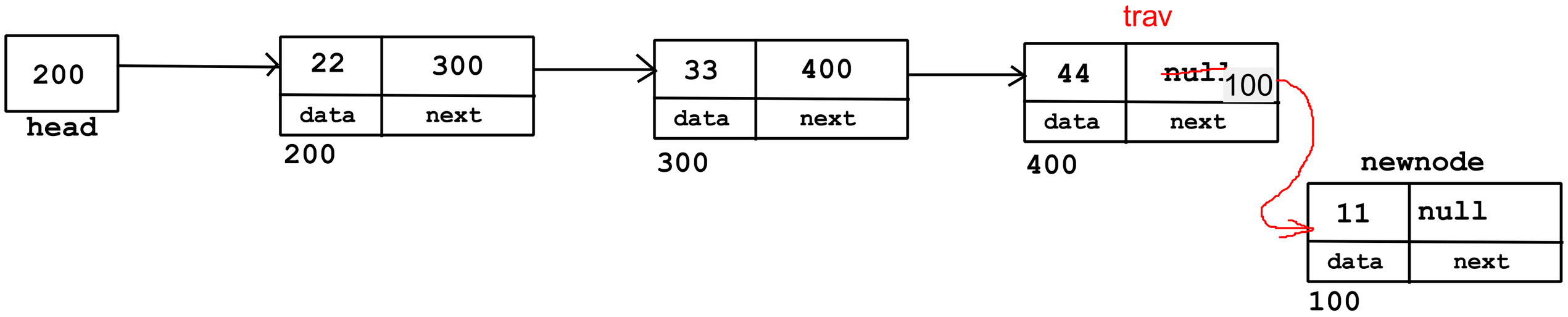
//1. if list is empty
    //a. print msg as list is empty
//2. if list is not empty
    //a. start at head
    //b. print data of that node
    //c. go on next node
    //d. repeat step b and c till trav != NULL

Singly Linear Linked List - Add Last



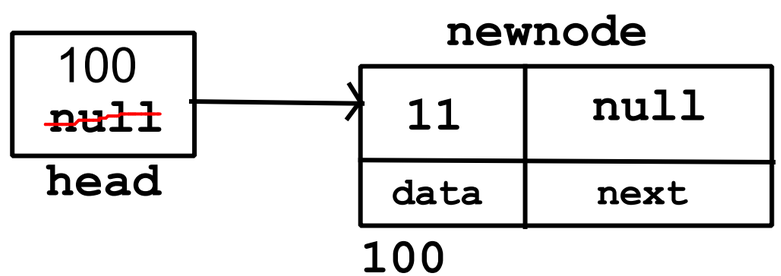
//0. create node with given data
//1. if list is empty
    //a. add newnode into head
//2. if list is not empty
    //a. traverse till last node
    //b. add newnode into next of last node

while(trav->next != NULL)
trav = trav->next



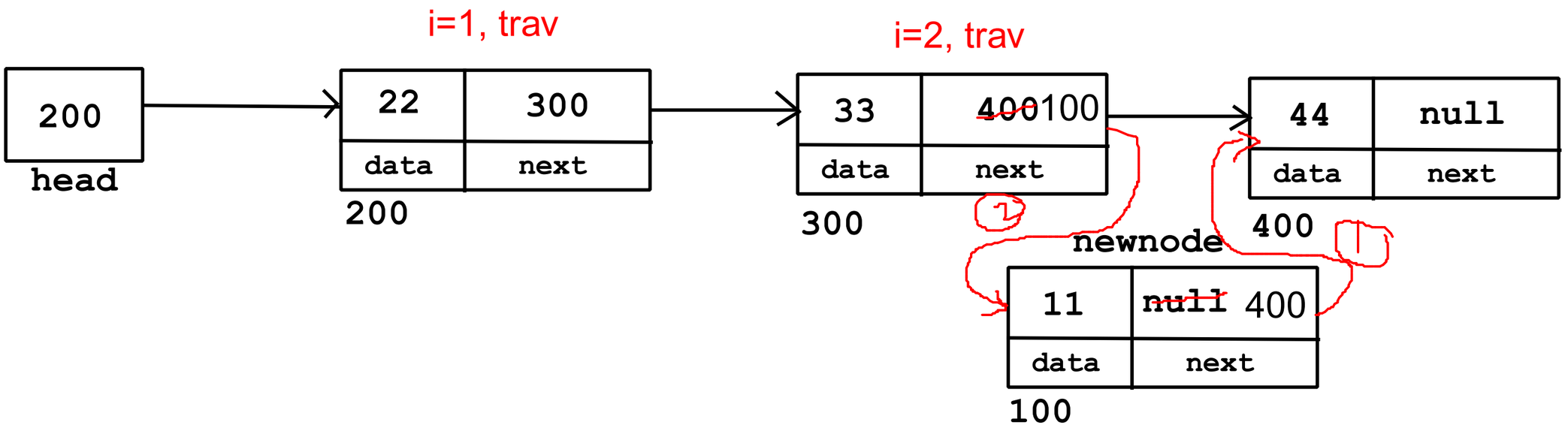
Singly Linear Linked List - Insert at pos

pos = 3

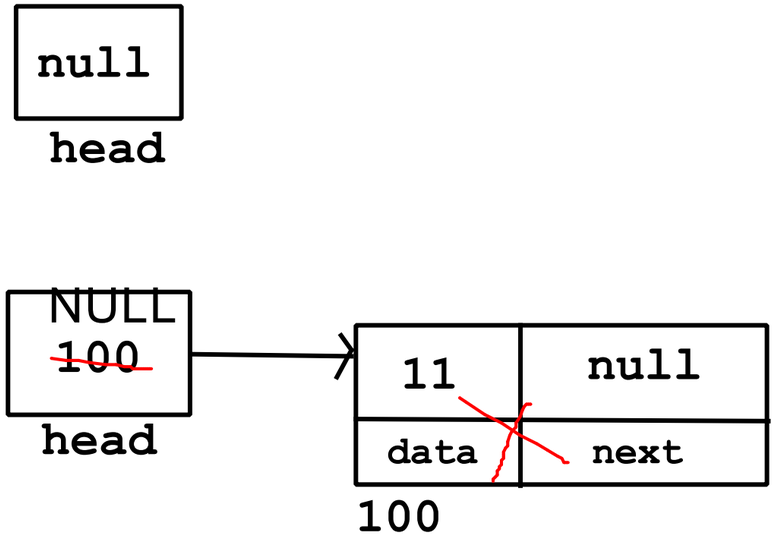


//0. create node with given data
//1. if list is empty
    //a. add newnode into head
//2. if list is not empty
    //a. traverse till pos -1 node
    //b. add pos node into newnode
    //c. add newnode into pos-1 node

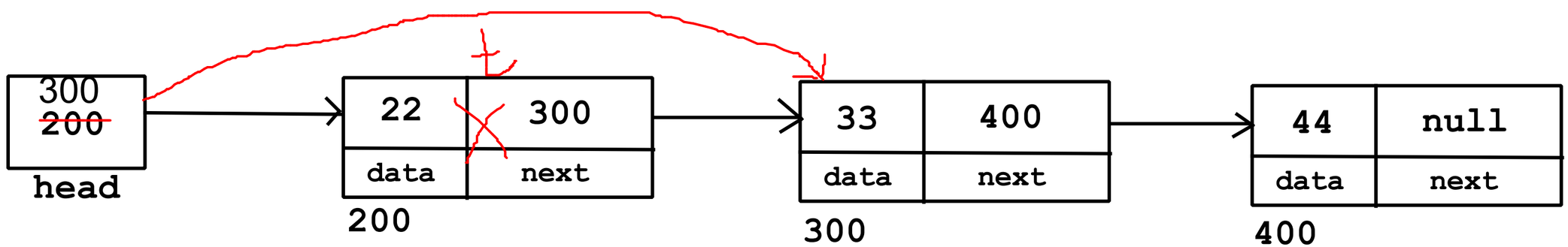
i=1
while(i != pos -1)



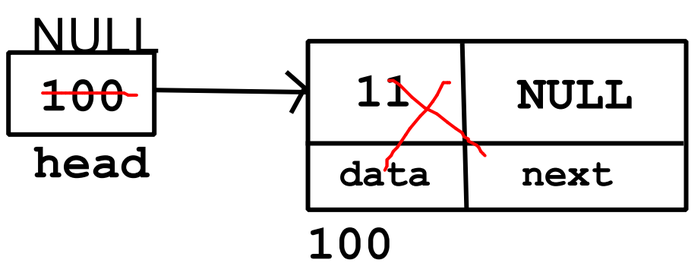
Singly Linear Linked List - Delete First




```
//1. if list is empty
        //print msg
//2. if list has only one node
        // delete head node only
//3. if list has multiple nodes
        //a. take backup of first node
        //b. add second node into head
        //c. delete backuped node
```



Linked List - Delete Node Last

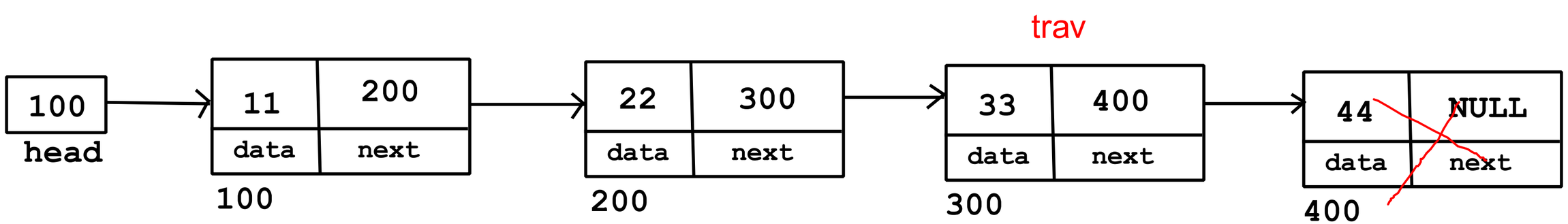



```
//1. if list is empty
        //print msg
//2. if list has one node
        //delete head node only
//3. if list has multiple nodes
        //a. traverse till second last node
        //b. delete last node
        //c. add NULL into next of second last node
```

trav != NULL --> trav = NULL

trav->next != NULL --> trav is on last node

trav->next->next != NULL --> trav is on second last node



Linked List - Delete Node at Position