

Lab 6

1. null list or null head
2. Special case: head
3. General case

addNode

0.  null

1. 2 if list is null, return -1

if head is null

head

↓
null

head

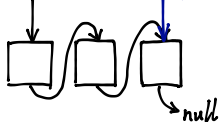
↓


↓
null

3. Loop till currentNode's next is null

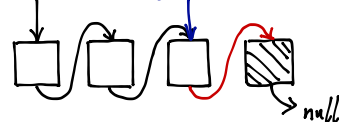
head

current



head

current



addSortedNode

0. 

1. if list is null, return -1

if head is null

head

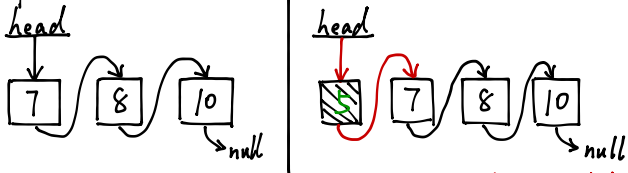
↓
null

head

↓

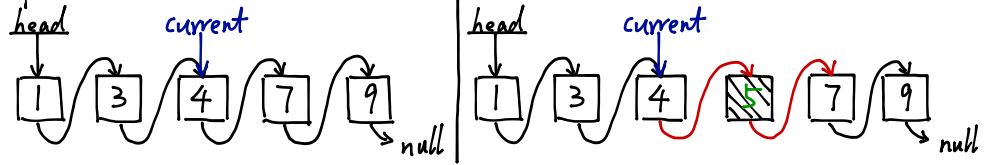

↓
null

2. if head's lon >= newNode's lon

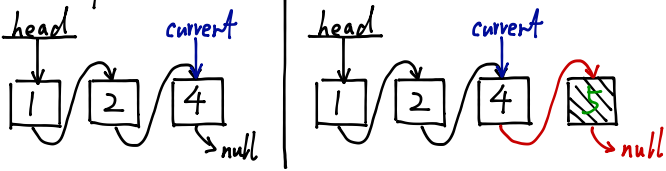


3. Loop till ~~currentNode's next~~ is null Not currentNode. Why?

if currentNode's next's lon >= newNode's lon



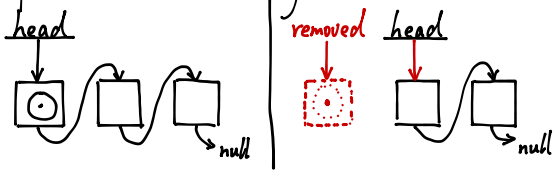
end loop



remNode

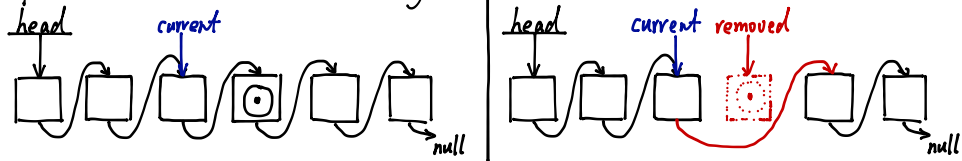
1. if list is null or head is null, return -1

2. if head's name == targetName



3. loop till ~~currentNode's next~~ is null Not currentNode. Why?

if currentNode's next's name == targetName



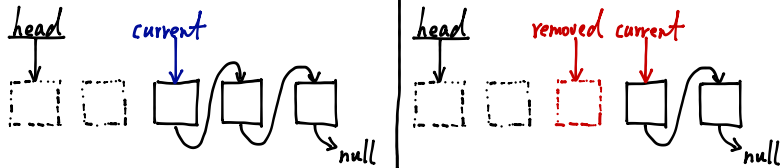
end loop

return -1

clearList

1, 2 if list is null or head is null, return 0

3. loop till currentNode's next is null



end loop

