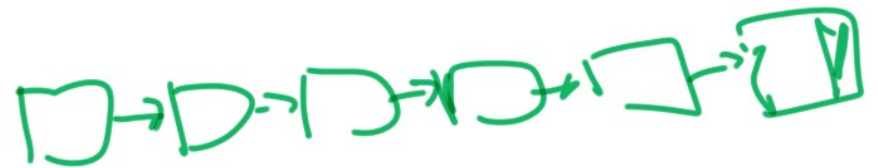
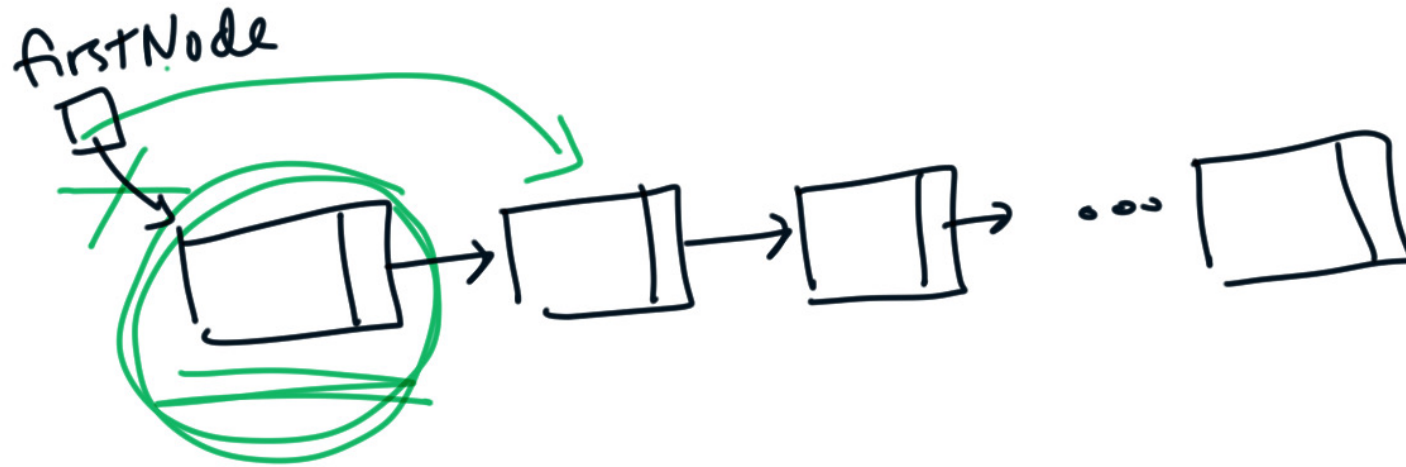


Removing a node from a non-empty list

1) Remove first node

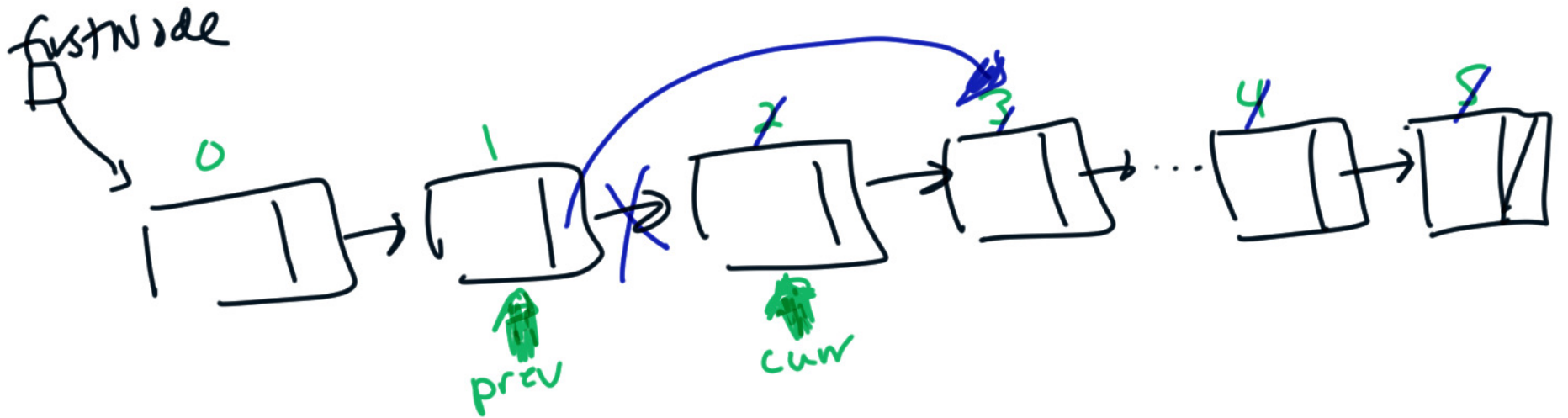
2) Remove a node other than first
(middle or last)





Remove the firstNode

$firstNode = firstNode.next$



Remove non-first Node

- 1) Reference to node to remove 6 node before it
- 2) Update the chain by skipping the node to remove

```

145 // Returns a reference to the node at a given position.
146 // Precondition: The chain is not empty;
147 // 0 <= givenPosition <= numberOfEntries - 1.
148 private Node getNodeAt(int givenPosition) {
149     Node currentNode = firstNode;
150     // Traverse the chain to locate the desired node
151     // (skipped if givenPosition is 0)
152     for (int counter = 0; counter < givenPosition; counter++)
153         currentNode = currentNode.getNext();
154     return currentNode;
155 } // end getNodeAt

```

```
36 //List should now contain 10 15 25 20 35 45 55
37
38 myList.remove(0);
39 //Removing first,
40 //List should now contain 15 25 20 35 45 55
41
42 myList.remove(5);
43 //Removing last, List should now contain 15 25 20 35 45
44
45 myList.remove(3);
46 //Removing middle, List should now contain 15 25 20 45
47
48 myList.replace(2, "25");
49 //Replacing middle, List should now contain 15 25 25 45
50
```

