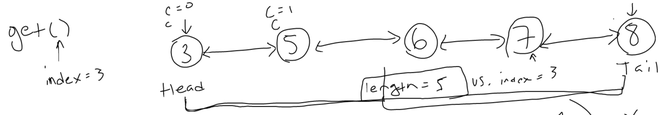


1. if (!this.head || this.length === 1) X
   if (index < 0 || index >= this.length) X

   let count;
   let current; ✓

2. if (index <= this.length/2) {

   count = 0;
   current = this.head
   while (count !== index){
      current = current.next
      count++

   } else {
      current = this.tail
      count = this.length-1
      while (count !== index){
         current = current.prev
         count--
   }

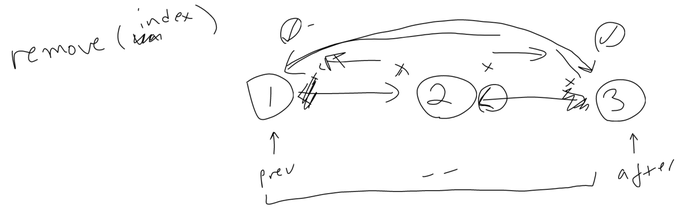
   return current

1. foundNode = this.get(index)

   if (foundNode){
      foundNode.val = val ✓
   }

   return this

1. if (!this.head) this.head = this.tail = newNode
2. if (index === 0) return this.unshift(val)
3. if (index === this.length-1) return this.push(val)
4. if (index < 0 || index >= this.length) return.
5. new Node
6. let prev = this.get(index-1)
   let after = prev.next or this.get(index+1)
7. prev.next = newNode, newNode.prev = prev
8. newNode.next = after, after.prev = newNode
9. this.length++
10. return this;



1) let removed = this.get(index)
   let prev = removed.prev
   let after = removed.next

2. prev.next = after
   after.prev = prev

3. removed.prev = null
   removed.next = null

4. this.length--

5. return this