

Linked List Design
COSC 201 - Fall 2012

Things we assume: int size, Node head, Node tail, Node current. Note that each "Case" represents an if-then block and will need to be placed in such when translating from this design to code.

```
Add(Node a, int i){
    //Case 1 - add middle
    Get to index i -
    int counter = 0;
        while(counter != i){
            current = current.next;
            counter++;
        }
    a.prev = current.prev;
    a.next = current;
    current.prev.next = a;
    current.prev = a;
    size++;

    //Case 2 - add front/head
    a.next = head;
    head.prev = a;
    head = a;
    size++;

    //Case 3 - add end/tail
    a.prev = tail;
    tail.next = a;
    tail = a;
    size++;

    //Case 4 - add empty
    head = a;
    tail=a;
    size++;
}
```

```
Remove(int i){

    //Case 1 - remove middle
    // Same loop as add
    current.next.prev = current.prev;
    current.prev.next = current.next;
    current = head;
```

```
size--;  
  
//Case 2 - remove end  
tail = tail.prev;  
tail.next = null;  
size--;  
  
//Case 3 - remove front  
head = head.next;  
head.prev = null;  
size --;  
  
//Case 4 - remove empty  
Print error message  
  
//Case 5 - remove when size == 1  
head = null;  
tail = null;  
size--;  
}
```