- 11.1 Circular Linked Lists
  - a circular list has the tail point back to the head instead of NULL
  - affects insertion, deletion & traversals
    - traversals cannot use "while not NULL"
      - also can't use "while not head"
        - because you start at the head
      - use a do-while
        - set ptr to heed
        - do
          - traversal action
          - set ptr to head's next
        - while ptr is not heed
    - insertion
      - head insertion
        - tail must be updated to point to new head
      - empty list insertion
        - creates one element list
        - element is both head & tail
          - must point back to self
    - deletion
      - must check for ore element list to transition back to empty list
      - cannot rely on head = head->getNext()
        - "next" of one element list is itself
- 11.3 Doubly Linked Lists
  - Adds a previous pointer to the list node
    - points back to the element before this one
  - Adds a "tail" pointer so List class
  - Can traverse list forward & back now
  - Also makes deletion easier because you don't have to find-previous
  - Insertion
    - head insertion
      - set new node's previous to NULL
      - set new node's next to head
      - set head's previous to new node
      - set head to new node
    - other insertions-inserting after prev
      - Create var called next that is pointing to prev->getNext()
      - So have vars: prev, next, new node
      - set prev's next to new node
      - if prev is tail
        - set tail to new node
      - else
        - set next's prev to new node
      - set new node's next to next
      - set new node's prev to prev
  - Deletion

```
head deletion
    set tmp to head
    set head to head->getNext()
    set head's prev to NULL
    delete tmp
tail deletion
    set tmp to tail
    set tail to tail->getPrev()
    set tail's next to NULL
    delete tmp
other deletions
    set tmp to node to delete
    set next to node->getNext()
    set prev to node->getPrev()
    set prev's next to next
    set next's prev to prev
    delete tmp
```