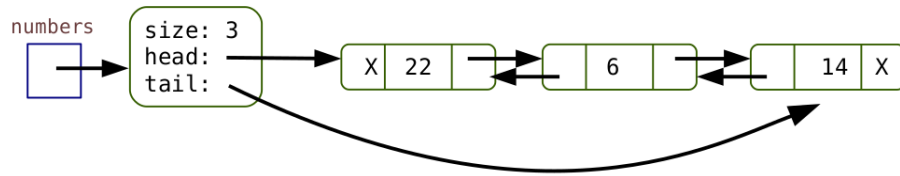


# Doubly-Linked Lists

Java's implementation of `LinkedList` stores two references in each node: one for the *previous*, and one for the *next*. In addition, both the head and the tail are stored in the wrapper class.



## Questions (10 min)

**Start time:**

1. How many operations are required to add an element *at the start* of this list?
2. How many operations are required to add an element *at the end* of this list?
3. How much memory is required for each node? How does that amount compare with using an `ArrayList`?
4. What problems of singly-linked lists do doubly-linked lists solve? (In other words, what do the `previous` and `tail` make possible?)
5. If your program requires a `List` collection, how would you decide which implementation to use? (`ArrayList` vs `LinkedList`)