public E removeFirstInstance(E obj) {

Node<E> previous = null, current = head;

if (isEmpty())

return null;

while (current != null &&

((Comparable<E>)obj).compareTo(current.data) != 0) {

previous = current;

current = current.next;

}

if (current == null) // empty

return null;

if (previous == null) // one element

head = head.next;

else if (current == tail) { // last element

previous.next = null;

tail = previous;

}

else

previous.next = current.next;

if(head == null)

tail = null;

currentSize--;

changeCounter++;

return current.data;

}

// Removes and returns last instance of an object, otherwise null

public E removeLastInstance(E obj) {

Node<E> previous = null, current = head, before = null,

target = current;

while (current != null) {

if (((Comparable<E>)obj).compareTo(current.data) == 0) {

before = previous;

target = current;

}

previous = current;

current = current.next;

}

if (target == null) // object not here or is empty

return null;

if (before == null) // if it's in 1st position

head = head.next;

else if (target == tail) { // if it's in last position

before.next = null;

tail = before;

}

else

before.next = target.next;

currentSize--;

changeCounter++;

return target.data;

}