```
                                CS 61B  Lab 4
                              February 15, 2004

Goal:  This lab will demonstrate how doubly-linked list implementations differ
if a sentinel is, or is not, present.

Copy the Lab 4 directory by starting from your home directory and typing:
  mkdir lab4
  cd lab4
  cp $master/lab/lab4/* .

Getting Started
---------------
Please make sure you have a partner for this lab.

The files in the lab4 directory contain classes for two different types of
doubly-linked list.  The DList1 class does not use a sentinel, whereas the
DList2 class does.  Compile DList1.java and DList2.java (using
"javac -g DList1.java DList2.java".)  Run the resulting test code with

    java DList1    and
    java DList2

The main() methods of DList1 and DList2 include brief test code in the main()
method.  Your task is to implement two removeFirst() methods, one for each list
class.  removeFirst() removes the first node in a list.  Make sure your
implementations work for empty lists, one-node lists, and larger lists.

Part I:  removeFirst in DList1 (2 points)
-----------------------------------------
Write a method called DList1.removeFirst() that removes the first node in
"this" DList1.

Part II:  removeFirst in DList2 (2 points)
------------------------------------------
Write a method called DList2.removeFirst() that removes the first node in
"this" DList2.  Your code should not require separate branches for the one-node
case and the more-than-one-node case.  (You will still need a separate branch
for the zero-node case.)

Check-off
---------
Run the DList1 and DList2 test code for your TA or Lab Assistant.

2 points:  DList1.
2 points:  DList2.
```