This assignment gives you practice with a simplified implementation of linked lists. Reverse the elements of a singly linked list using an iterative method and a recursive method.

You must reverse the linked list in place. You may not copy the elements to an array or an array list.

In Codecheck, you are given a linked list class `AbbreviatedLinkedList` that has a private inner `Node` class, a private field `first`, and a public method `addFirst()`. There is no iterator class.
Draft

You will add the iterative `reverse()` method:

```java
public void reverse()
```

Sample output:

```
Original list: [A B C D]
Iteratively reversed: [D C B A]
```

Codecheck URL:
http://codecheck.it/codecheck/files/15072108363747y0fo08ar6knsd11a5wjak

Canvas: Homework 9 Draft
Due: Monday, July 20 at 11:59 PM

Final

Replace your iterative method with your recursive method:

```java
private Node reverse(Node p)
```

where parameter `p` refers to the starting node of the list where reversal is to begin. There is a public driver method for the recursive `reverse()` method:

```java
public void reverseRecursive()
{
    first = reverse(first);
}
```

Sample output:

```
Original list: [A B C D]
Recursively reversed: [D C B A]
```

Codecheck URL:
http://codecheck.it/codecheck/files/1507210846ccvdl3kluu0qfmolsx1y2h1i

Canvas: Homework 9 Final
Due: Monday, July 27 at 11:59 PM