

**Bronx Community College**  
**Department of Mathematics and Computer Science**  
**CSI33 Fall 2019**  
**Programming Assignment 2: worth 20 points**  
**Assigned on September 23, 2019**  
**Due on October 7, 2019**

**Assignment 2, updated**

Add a `last` instance variable to the `LList` class. This `self.last` variable will be a reference to the last node in the list. Be sure that every method maintains the class invariant as described in the document `Linked List Class Invariant, updated`. (Use the version of the class in the file `LList2.py`, the one that uses an `insertathead` function in the constructor.)

Then rewrite the `append` method to take advantage of the `last` variable. This version of `append` will take  $\Theta(1)$  time.

Modify the methods of the class, including the constructor, to ensure that the `last` variable always refers to the last node in the list. Modify all the methods that insert or delete a node to take advantage of the `last` variable when changing the last node in the list when you can, and to ensure that the class invariant is always preserved.

Finally, write an implementation of the list method `count(x)` that returns the number of times the parameter `x` occurs as an element of the list.

Submit the new version of the `LList` class definition to me by email at [sharon.persinger@bcc.cuny.edu](mailto:sharon.persinger@bcc.cuny.edu) by the end of the day on 10/7/2019. The subject line of your email should be `CSI33 Assignment 2`. Please be sure to include your name in the program file as a comment.

Updates:

I changed the implementation of the `_insertathead` method to make it private and to make it responsible for changing the `self.size` variable when an item is added. Even though the initial underscore indicates a private method, the method could be called directly and so it is better to have the method maintain the Linked List class invariant. This is the version in `LList2.py` posted for Sept. 25, 2019.

The methods that need to be changed once the `self.tail` variable is added include:

1. Constructor: Set the variables `self.tail` and `self.length` correctly when the first item is

placed into the list. If you then use the `_insetathead` method to place the remaining items, `self.tail` will not be changed with the later insertions.

2. `append`: This method handles adding an item to the end of the list, and so the `self.tail` variable must be updated. Be sure to deal correctly with appending to an empty list.
3. `_insertathead`: This method should be changed to handle inserting into an empty list.
4. `_delete`: This method should be changed to deal with the case of deleting the last item. `_delete` is used in the other methods that delete an item, such as `pop`.
5. `Insert`: This method should be changed to deal with the case of inserting at position `self.size`, that is, adding a new item at the end of the list, after the existing items.

The unit test module `test_LList2.py` is set up to import the Linked List implementation from `LList2.py`. Please do not change the name of the module.