Assignment Prefix: Lab05

Due Date: Tuesday, Feb. 20th @ 11:59pm

Points: 100

This is an individual assignment.

**Restrictions:**

**You cannot use any predefined Java classes in writing this lab.**

**You CAN import the java.util.Random class.**

**Create a NetBeans project named Lab05 and save it to a location like the desktop or your flash drive. In the project you will do the following:**

In this assignment, you are to create a complete DoublyLinkedList Class.

Begin by copying your SinglyLinkedList from one of your previous assignments. This codes should include Code Fragments 3.14 and 3.15 from the textbook pages 126 to 127.

Add the following method to the SinglyLinkedList class:

* public String toString( )
* this method should return a String that includes
  + the class name
  + the size of the list
  + the contents of each element in the list

Now create a new class named DoublyLinkedList. This class should work like the Doubly Linked List described in the textbook. Using the SinglyLinkedList as a guide modify the DoublyLinkedList so that:

* The nested node class includes:
  + a previous (prev) pointer
  + a getPrev() method
  + a setPrev() method
* The DoublyLinkedList has header and trailer instance variables that point to sentinel nodes (i.e. nodes that anchor the ends of the list but never reference elements)
* Include all the corresponding methods from the SinglyLinkedList class and add the following:
  + Private methods:
  + private void addBetween( E e, Node<E> predecessor, Node<E> successor )
  + private E remove( Node<E> node )
  + Public methods:
  + public E removeLast( )
  + public String toString( )
* HINTS:
  + your addFirst and addLast methods should call your addBetween method
  + your removeFirst and removeLast methods should call you remove method.

Testing your code:

Part 1:

Create a SinglyLinkedList of Integers:

* Add ten values ( 1 through 10 ) to your list in ascending order.
* Use the toString method to print the contents of the list.
* Call remove first and display the value removed.
* Use the toString method to print the contents of the list.

Create a DoublyLinkedList of Integers:

* Add ten values ( 1 through 10 ) to your list in ascending order.
* Use the toString method to print the contents of the list.
* Call remove first and display the value removed.
* Call remove last and display the value removed.
* Use the toString method to print the contents of the list.

Part 2:

Start with an empty SinglyLinkedList and time how long it takes to add 100,000,000 Integers to the list.

Start with an empty DoublyLinkedList and time how long it takes to add 100,000,000 Integers to the list.

Print your results in a nicely formatted fashion, e.g.:

SinglyLinked N = 100,000,000 time(msec) = 12,345

DoublyLinked N = 100,000,000 time(msec) = 12,345

**Things to turn in:**

* Open a Microsoft Word document
* Copy and Paste the source code of the **Client Class, SinglyLinkedList** **Class, and DoublyLinkedList Class** (make sure to use *Ctrl + A* to select all the source code of the program, *Ctrl + C* to copy, and Ctrl + V to paste.).
* Copy and paste the output of the client program
* Next, zip the Project folder.
* Finally on blackboard, submit both your Word document and project zipped file.