**Assignment 3**

**CMSC204**

**Spring 2016**

Concepts tested by this program:

Exceptions

New concepts tested by this program

Generic Classes

Doubly Linked List

Ordered Doubly Linked List

Iterators

Comparators

Your assignment is to write a generic doubly-linked list class and a generic sorted doubly-linked list class that inherits from your generic doubly-linked class. There is no additional GUI required for this assignment. Your list classes will be tested with Junit tests.

**BasicDoubleLinkedList class**

This generic doubly-linked list relies on a head (reference to first element of the list) and tail (reference to the last element of the list). Both are set to null when the list is empty. Both point to the same element when there is only one element in the list. A node structure has only three fields: data and the prev and next references. The class must only define the following entities: an inner class Node, an inner class that implements ListIterator (for the iterator method), head and tail references and an integer representing the list size. However only the next(), hasNext(), previous() and hasPrevious() methods of the ListIterator that you are required to implement. The rest of the methods can return can throw the UnsupportedOperationException, such as:

public void remove() throws UnsupportedOperationException{

throw new UnsupportedOperationException();}

All the entities are defined as protected so they can be accessed by the subclass. Follow the Javadoc that is provided.

**SortedDoubleLinkedList class**

A generic sorted doubly-linked list constructed using a provided Comparator. It extends BasicDoubleLinkedList class. Follow the Javadoc that is provided.

**Exception Handling**

UnsupportedOperationException – this exception is a Java library exception and will be returned by the addtoFront and addToEnd implementations of the SortedLinkedList class and by the remove method of the iterator.

NoSuchElementException – this exception is a Java library exception and will be returned by the next function within the iterator class when there are no more elements in the linked list.

Deliverables:

Java files - The src folder with your driver (javafx application), data element, data manager and Junit Test (.java) files

Javadoc files - The doc folder with your javadoc for student generated files

UML Class Diagram (an image, not the proprietary format, must be a .jpg)

Deliverable format: The above deliverables will be packaged as follows. Two compressed files in the following formats:

LastNameFirstName\_AssignmentX\_**Complete**.zip [a compressed file containing the following]

UML.jpg

Assignment 2 Checklist (filled in with YES or NO or ?)

doc [a directory] *please include the entire doc folder with the javadoc for student*

*generated files*

file1.html (example)

file2.html (example)

src [a directory] *contains your driver (javafx application), enumerated class, data*

*element, data manager and Junit Test (.java) files*

File1.java (example)

File2.java (example)

File\_Test.java (example)

LastNameFirstName\_AssignmentX\_**Moss**.zip [a compressed file containing only the following]

*contains .java file which includes the driver (javafx application), enumerated*

*class, data element, data manager and Junit Test (.java) files – NO FOLDERS!!*

File1.java (example)

File2.java (example)

**Assignment 3 Grade Sheet – Spring 2016**

Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Blackboard Date/Time:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

## DOCUMENTATION 25 points

# CheckList for Assignment 2 is included and completed 1 pt \_\_\_\_\_

# Class documentation - Javadoc for all user created classes 5 pt \_\_\_\_\_

Class Description, @author

Method description, @param, @return

Test Cases 10 pt \_\_\_\_\_

JUnit Test Class

Implement STUDENT test for BasicDoubleLinkedListTest

and SortedDoubleLinkedTest

UML Diagram 4 pt \_\_\_\_\_

Lessons Learned 5 pt \_\_\_\_\_

In 3+ paragraphs, highlight your lessons learned and learning experience from working on this project. How did you do? What have you learned? What did you struggle with? How will you approach your next project differently?

## PROGRAMMING 75 points

Internal class documentation (within source code) 5 pt \_\_\_\_\_

Class description using Javadoc

Author’s Name, Class, Class Time, @author

Methods commented using Javadoc, @param, @return

Compiles and Runs without runtime errors or warnings 10 pts \_\_\_\_\_

Accuracy

Public tests – JUnit test given to you 6 pts \_\_\_\_\_

Your JUnit Tests – STUDENT test for BasicDoubleLinkedListTest

and SortedDoubleLinkedTest 4 pts \_\_\_\_\_

Private tests – other tests run by the instructor 10 pts \_\_\_\_\_

Program Detail

BasicDoubleLinkedListStudent **20 points**

Implements all methods of BasicDoubleLinkedList

addToEnd, addToFront 3 pt \_\_\_\_\_

getFirst, getLast 2 pt \_\_\_\_\_

getSize 1 pt \_\_\_\_\_

iterator -inner class implements ListIterator 5 pt \_\_\_\_\_

remove 3 pt \_\_\_\_\_

retrieveFirstElement, retrieveLastElement 3 pt \_\_\_\_\_

toArray 3 pt \_\_\_\_\_

SortedDoubleLinkedList **12 points**

Implements all methods of the SortedDoubleLinkedList

add 8 pt \_\_\_\_\_

addToEnd, addToFront throw exceptions 2 pt \_\_\_\_\_

iterator – uses superclass iterator method 2 pt \_\_\_\_\_

Exception handling 8 pt \_\_\_\_\_

java.lang.UnsupportedOperationException

java.lang.NoSuchElementException

Total 100 pt \_\_\_\_\_

Comments: