### SortedListReferenceBased

## **Topics**

- Throughout this lab, we want to practice abstract data type (ADT).
- In the previous lab/assignment, you were asked to implement a SortedListArrayBased class, while this lab asks you to implement two classes: "ListReferenceBased.java" and "SortedListReferenceBased.java".
- After finishing this lab, you will be more confident in manipulating reference.

### **Activities**

# <<interface>> ListInterface + isEmpty(): Boolean + **size()**: int + get(index: int): Object + removeAll(): void + add(index: int, item: Object): void + remove(index: int): void ListReferenceBased - head: Node - numItems: int + ListReferenceBased() + isEmpty(): Boolean + **size()**: int + get(index: int): Object + removeAll(): void + add(index: int, item: Object): void + remove(index: int): void SortedListReferenceBased + SortedListReferenceBased() + add(item: Object): void + remove(item: Object): void + locateIndexToAdd(item: Object): int

+ locateIndexToRemove(item: Object): int

- Start with the following supportive files ("ListException.java", "ListIndexOutOfBoundsException.java", "ListInterface.java", "Node.java", "SortedListDriver.java" "ListReferenceBased.java", and "SortedListReferenceBased.java").
- The first five java files are complete, thus use them as they are. You're asked to show your work on the two java files, "ListReferenceBased.java" and "SortedListReferenceBased.java".
- In Ch. 5, we discussed the required implementation details of "<u>ListReferenceBased.java</u>". Also, please refer to your "<u>SortedListArrayBased.java</u>" in the previous lab/assignment for "<u>SortedListReferenceBased.java</u>".
- You should keep the sorted (say, non-decreasing) order of items, while adding and deleting an item using "add()" and "delete()", respectively. Keeping the sorted order of the contents in the list is the key in this lab.
- Note that index starts from 0, not 1. Therefore, the valid index ranges from 0 to size()-1.

## **Running Example**

0. apples	Removing juice	Removing sausage
1. black beans	0. apples	0. apples
2. bread	<ol> <li>black beans</li> </ol>	1. banana
3. butter	2. bread	2. black beans
4. chicken	3. butter	3. bread
5. eggs	4. chicken	4. butter
6. flour	5. eggs	5. chicken
7. milk	6. flour	6. eggs
8. pecans	7. milk	7. flour
9. rice	8. pecans	8. milk
10. sausage	9. rice	9. pecans
numItems is now: 11	10. sausage	10. rice
	numItems is now: 11	numItems is now: 11
Adding juice		
item 5 is: eggs	Adding banana	Removing cheese
0. apples	0. apples	0. apples
1. black beans	1. banana	1. banana
2. bread	2. black beans	2. black beans
3. butter	3. bread	3. bread
4. chicken	4. butter	4. butter
5. eggs	5. chicken	5. chicken
6. flour	6. eggs	6. eggs
7. juice	7. flour	7. flour
8. milk	8. milk	8. milk
9. pecans	9. pecans	9. pecans
10. rice	10. rice	10. rice
11. sausage	11. sausage	numItems is now: 11
numItems is now: 12	numItems is now: 12	Press any key to continue

#### What to Hand in

• Turn in your programs, "ListReferenceBased.java" and "SortedListReferenceBased.java", via Blackboard.