GIT Department of Computer Engineering

CSE 222/505 - Spring 2020

Homework #3 Part 1 Report

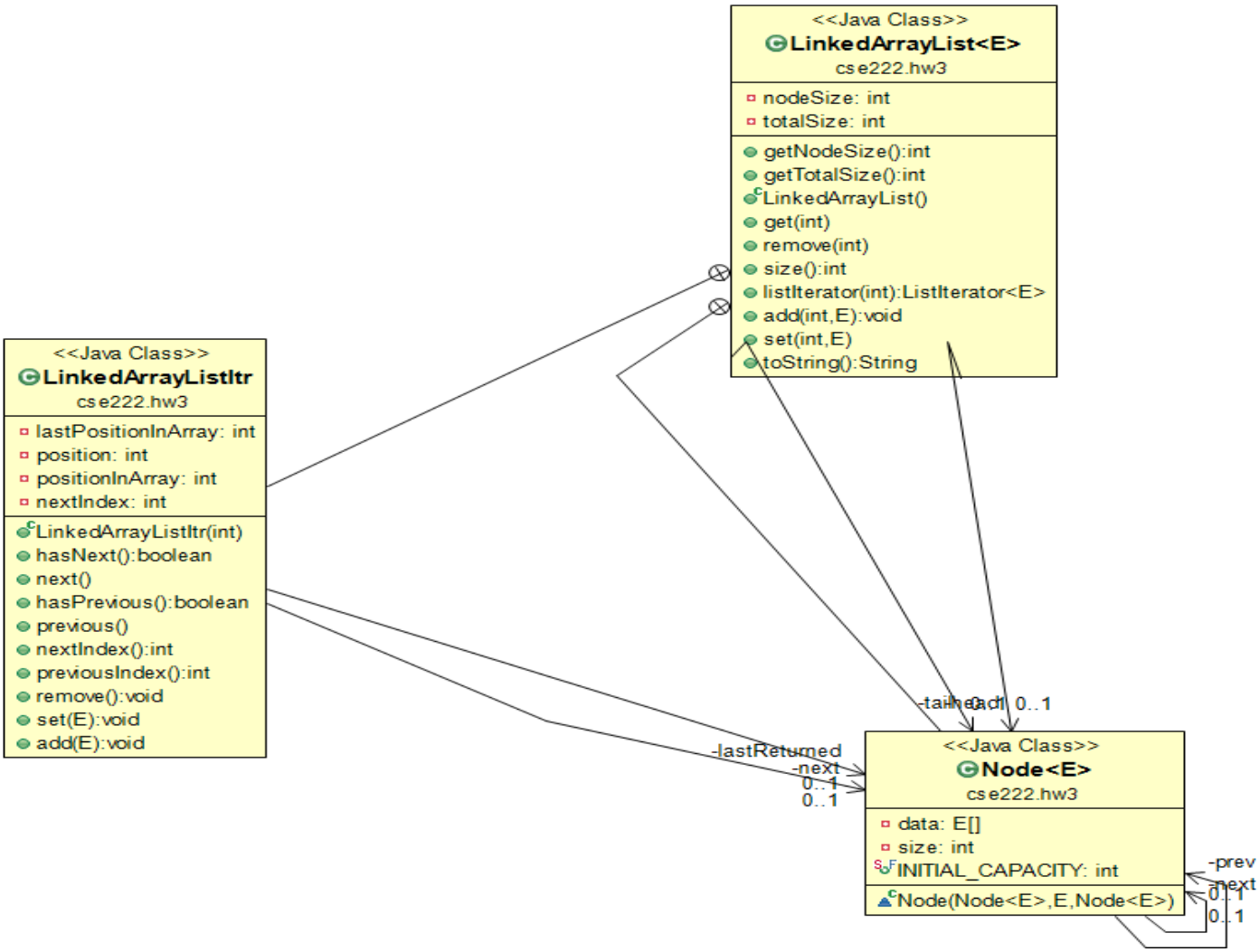
Murat YILDIZ

1801042004

PROBLEM SOLUTION APPROACH

Step 1 – I implemented List interface and extend AbstractList class.
Step 2 – Then create an static inner class named Node which holds generic type array with constant capacity 4, a node to next , a node to previous
Step 2 – Then create an inner iterator class which is implements ListIterator (LinkedArrayListItr).
Step 3 - After implement all required methods inherited ListIterator class in LinkedArrayListItr class, I used these methods to implement all required methods inherited List interface and AbstractList in LinkedArrayList class
Step 4 – After implementations, I try to test all functions implemented and overrided.

CLASS DIAGRAM



TEST CASES

Test ID	Scenario	Test Data	Expected Results	Actual Results	Pass/ Fail
Test01	Create an empty Integer list	LinkedArrayList Class constructor	Successfully created	As expected	Pass
Test02	remove(index) method called when list empty.	Index = 0	Exception throwed	As expected	Pass
Test03	add(index,element) method called at start of the list when it is empty	Index = 0 Element = 5	Successfully added	As expected	Pass
Test04	add(index,element) method called at end of the list.	Index = 1 Element = 8	Successfully added	As expected	Pass
Test05	Add some elements at the end	Elements 7,7,15,22,35,5,3	Successfully added	As expected	Pass
Test06	add(index,element) method called at start of the list when it is not empty.	Index = 0 Element = 10	Successfully added	As expected	Pass
Test07	add(index,element) method called with invalid index	Index = 20 Element = 15	Exception throwed	As expected	
Test08	remove(index) method called and removed specified index which is begin of list	Index = 0	Successfully removed	As expected	Pass
Test09	remove(index) method called and removed specified index	Index = 4	Successfully removed	As expected	Pass
Test10	remove(index) method called and removed specified index which is end of list	Index = size – 1	Successfully removed	As expected	Pass
Test11	remove(index) method called with invalid index	Index = 20	Exception throwed	As expected	Pass
Test12	get(index) method called with start index of the list	Index = 0	Successfully Returned data	As expected	Pass
Test13	get(index) method called with specified index	İndex = 5	Successfully Returned data	As expected	Pass
Test14	get(index) method called with last index of the list	Index = size-1	Successfully Returned data	As expected	Pass
Test15	get(index) method called with invalid index	Index = 20	Exception throwed	As expected	Pass
Test16	set(index,element) method called with start index	İndex = 0	Successfully Setted data	As expected	Pass
Test17	set(index,element) method called with specified index	İndex = 5	Successfully Setted data	As expected	Pass
Test18	set(index,element) method called with end index	Index = size-1	Successfully Setted data	As expected	Pass
Test19	set(index,element) method called with invalid index	Index = 20	Exception throwed	As expected	Pass

Test20	listIterator(index) method called	İndex = 0	Successfully returned Iterator	As expected	Pass
Test21	After listIterator(index), printing list with next function until hasNext will return false		Successfully Printed list forward	As expected	Pass
Test22	After printed all with next function, printing list with previous function until hasPrevious will return false		Successfully Printed list backward	As expected	Pass
Test23	After printed all with previous function, printing list with next function until hasNext will return false		Successfully Printed list forward	As expected	Pass
Test24	After printed all with next function, printing list with previous function until hasPrevious will return false		Successfully Printed list backward	As expected	Pass
Test25	Use listIterator(index) function	İndex = size+1	Exception throwed	As expected	Pass
Test26	Use listIterator(index) function	İndex = -1			
Test27	After listIterator(index), use next function	İndex = size+1	Exception throwed	As expected	Pass
Test28	After listIterator(index), use previous function	İndex = 0	Exception throwed	As expected	Pass
Test29	After call listIterator (index), call nextIndex	İndex = 5	Successfully Returned index	As expected	Pass
Test30	After call listIterator (index), call previousIndex	İndex = 5	Successfully Returned index	As expected	Pass
Test31	`After call listIterator(index), before call next or previous, call remove`	İndex = 5	Exception throwed	As expected	Pass
Test32	`After call listIterator(index) and call next, call remove`	İndex = 5	Successfully Removed	As expected	Pass
Test33	`After call listIterator(index), call add`	İndex = 5	Successfully Added	As expected	Pass
Test34	`After call listIterator(index), before call next or previous, call set`	İndex = 5	Exception throwed	As expected	Pass
Test35	`After call listIterator(index) and call next, call set`	İndex = 5	Successfully Changed	As expected	Pass

RUNNING AND RESULTS

Test01

```
LinkedArrayList constructor called with Integer type

LinkedArrayList is created with Integer type
```

Test02

```
Try to delete element from an empty LinkedArrayList with index(0)

java.lang.IndexOutOfBoundsException: Index is not valid
```

Test03

```
Try to add an element(5) to an empty LinkedArrayList

The LinkedArrayList
 5,
```

Test04

```
Try to add an element(8) to end of the LinkedArrayList

The LinkedArrayList
 5, 8,
```

Test05

```
Try to add elements(7,7,15,22,35,5,3) to end of the LinkedArrayList

The LinkedArrayList
 5, 8, 7, 7, 15, 22, 35, 5, 3,
```

Test06

```
Try to add an element(10) to begin of non empty LinkedArrayList

The LinkedArrayList
 10, 5, 8, 7, 7, 15, 22, 35, 5, 3,
```

Test07

```
Try to add an element(10) Indexed which is 20 out of bounds to LinkedArrayList

java.lang.IndexOutOfBoundsException: Index is not valid

The LinkedArrayList
 10, 5, 8, 7, 7, 15, 22, 35, 5, 3,
```

Test08

```
The LinkedArrayList
 10, 5, 8, 7, 7, 15, 22, 35, 5, 3,

Try to delete an element(10) indexed(0) from LinkedArrayList

The LinkedArrayList
 5, 8, 7, 7, 15, 22, 35, 5, 3,
```

Test09

```
The LinkedArrayList
 5, 8, 7, 7, 15, 22, 35, 5, 3,

Try to delete an element(15) indexed(4) from LinkedArrayList

The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5, 3,
```

Test10

```
The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5, 3,

Try to delete an element(3) indexed(7) from LinkedArrayList

The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5,
```

Test11

```
The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5,

Try to delete an element indexed(20) which is out of bounds from LinkedArrayList

java.lang.IndexOutOfBoundsException: Index is not valid
```

Test12

```
The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5,

***** USE OF GET INDEX 0: 5
```

Test13

```
The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5,

***** USE OF GET INDEX 5: 35
```

Test14

```
The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5,

***** USE OF GET INDEX SIZE-1: 5
```

Test15

```
The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5,

***** USE OF GET INDEX 20:

java.lang.IndexOutOfBoundsException: Index is not valid
```

Test16

```
The LinkedArrayList
 5, 8, 7, 7, 22, 35, 5,

***** USE OF SET INDEX 0: OLD VALUE : 5  NEW VALUE : 11

The LinkedArrayList
 11, 8, 7, 7, 22, 35, 5,
```

Test17

```
The LinkedArrayList
 11, 8, 7, 7, 22, 35, 5,

***** USE OF SET INDEX 5: OLD VALUE : 35  NEW VALUE : 11

The LinkedArrayList
 11, 8, 7, 7, 22, 96, 5,
```

Test18

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 5,

***** USE OF SET INDEX SIZE-1: OLD VALUE : 5  NEW VALUE : 11

The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,
```

Test19

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,

***** USE OF SET INDEX 20:

java.lang.IndexOutOfBoundsException: Index is not valid
```

Test20

```
Create LinkedListArrayItr with index(0)
```

Test21

```
Testing listIterator next function as printing forward until hasNext function will return false

11, 8, 7, 7, 22, 96, 65,
```

Test22

```
After tested next function Testing listIterator previous function as printing backward until hasPrevious function will return false

65, 96, 22, 7, 7, 8, 11,
```

Test23

```
After tested previous function Testing listIterator next function as printing forward until hasNext function will return false

11, 8, 7, 7, 22, 96, 65,
```

Test24

```
After tested next function Testing listIterator previous function as printing backward until hasPrevious function will return false

65, 96, 22, 7, 7, 8, 11,
```

Test25

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,

Call listIterator function with index size+1

java.lang.IndexOutOfBoundsException: Index is not valid
```

Test26

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,

Call listIterator function with index -1

java.lang.IndexOutOfBoundsException: Index is not valid
```

Test27

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,

After call listIterator function with index size, call next function

java.util.NoSuchElementException: There is no next element
```

Test28

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,

After call listIterator function with index 0, call previous function

java.util.NoSuchElementException: There is no previous element
```

Test29

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,

After call listIterator function with index 5, call nextIndex
nextIndex = 5
```

Test30

```
After call listIterator function with index 5, call previousIndex
previousIndex = 4
```

Test31

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,

After call listIterator function with index 5 , call remove

java.lang.IllegalStateException
```

Test32

```
The LinkedArrayList
 11, 8, 7, 7, 22, 96, 65,

After call listIterator function with index 5 and call next, call remove

The LinkedArrayList
 11, 8, 7, 7, 22, 65,
```

Test33

```
The LinkedArrayList
 11, 8, 7, 7, 22, 65,

After call listIterator function with index 5 , call add(89)

The LinkedArrayList
 11, 8, 7, 7, 22, 89, 65,
```

Test34

```
The LinkedArrayList
 11, 8, 7, 7, 22, 89, 65,

After call listIterator function with index 5 , call set(69)

java.lang.IllegalStateException
```

Test35

```
The LinkedArrayList
 11, 8, 7, 7, 22, 89, 65,

After call listIterator function with index 5 and call next, call set(69)

The LinkedArrayList
 11, 8, 7, 7, 22, 69, 65,
```