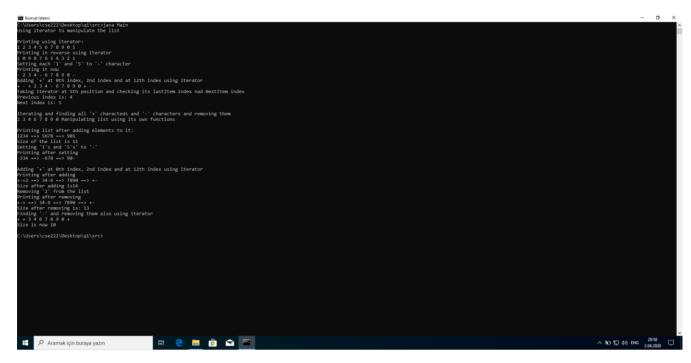# Question 1

## Problem solution approach

I decided to approach this problem by making a structure that would have list functionality. Inside there is a static class of Node that has an array of elements inside. Also there is a private non-static class of iterator that provides iterator functionality and implements listIterator.

## Test cases and & running command and results

They can be checked from the command line. Main class is used for testing. Its outputs can be seen by running it

| Test Scenario | Expected Results | Actual Results |
|---|---|---|
| Adding to list using iterator( empty and non-empty list ) | Should be added | As expected |
| Printing list using iterators and hasNext() and next() functions | Should print the whole list properly | As expected |
| Printing list in reverse using hasPrevious() and previous() methods | Should be printted in reverse | As expected |
| Iterating using hasNext() and next() and finding elements '1' | Should set all 5s and 1s to - character | As expected |

| | | |
|---|---|---|
| and '5' and changing them | | |
| Adding '+' character to 0$^{th}$ index, 2$^{nd}$ index and 12th index | Should add elements at those positions using iterator | As expected |
| Getting iterator at 5$^{th}$ position and checking its previous and next index | Should find 5 and 4 | As expected |
| Iterating and finding all '-' and '+' and removing them | Should remove | As expected |
| Not using iterators | | |
| Adding to empty and non-empty | Should add elements | As expected |
| Setting elements at 0$^{th}$ 2$^{nd}$ and 12$^{th}$ position | Should set them to '-' | As expected |
| printing | Should print | As expected |
| Removing '2' element | Should remove | As expected |
| Printing size after removing | Should print the correct size | As expected |
| Removing '-' element using iterator | Should remove it | As expected |

```
C:\Users\cse222\Desktop\q1\src>java Main
Using iterator to manipulate the list

Printing using iterator:
1 2 3 4 5 6 7 8 9 0 1
Printing in reverse using iterator
1 0 9 8 7 6 5 4 3 2 1
Setting each '1' and '5' to '-' character
Printing it now
- 2 3 4 - 6 7 8 9 0 -
Adding '+' at 0th index, 2nd index and at 12th index using iterator
+ - + 2 3 4 - 6 7 8 9 0 + -
Taking iterator at 5th position and checking its lastItem index nad NextItem index
Previous index is: 4
Next index is: 5

Iterating and finding all '+' characteds and '-' characters and removing them
2 3 4 6 7 8 9 0 Manipulating list using its own functions

Printing list after adding elements to it:
1234 ==> 5678 ==> 901
Size of the list is 11
Setting '1's and '5's to '-'
Printing after setting
-234 ==> -678 ==> 90-

Adding '+' at 0th index, 2nd index and at 12th index using iterator
Printing after adding
+-+2 ==> 34-6 ==> 7890 ==> +-
Size after adding is14
Removing '2' from the list
Printing after removing
+-+ ==> 34-6 ==> 7890 ==> +-
Size after removing is: 13
Finding '-' and removing them also using iterator
+ + 3 4 6 7 8 9 0 +
Size is now 10

C:\Users\cse222\Desktop\q1\src>
```

*Here are run commands and their results*

Class diagram

## - **LinkedArrayList.Node<T>**

**fields**
- elements : T[]
- next : Node
- prev : Node
- current_i : int

**constructors**
- + Node ()
- - Node ( el : T , next : Node<T> , prev : Node<T> )

**methods**
- - add ( el : T ) : boolean
- + getNext () : Node
- + getPrev () : Node
- + setNext ( next : Node ) : void
- + setPrev ( prev : Node ) : void
- + set ( i : int , el : T ) : T
- + capacity () : int
- + size () : int
- + get ( i : int ) : T
- + remove ( index : int ) : T
- + toString () : String

## + **LinkedArrayList<T>** extends AbstractList
implements Iterable

**fields**
- - size : int
- - head : Node
- - tail : Node
- - final  ARR_SIZE : int

**constructors**
- + LinkedArrayList ()

**methods**
- - getNode ( index : int ) : Node<T>
- + get ( i : int ) : T
- + add ( i : int , el : T ) : void
- + add ( el : Object ) : boolean
- + remove ( i : int ) : T
- + set ( i : int , el : T ) : T
- + size () : int
- + toString () : String
- - getHead () : Node<T>
- + listIterator () : LinkedArrayListIter2
- + listIterator ( i : int ) : LinkedArrayListIter2

## **LinkedArrayList.LinkedArrayListIt...**
implements ListIterator

**fields**
- - node_index : int
- - node_index_prev : int
- - node : Node<T>
- - prevNode : Node<T>
- - nextItemT : T
- - lastItemReturnedT : T
- - index : int
- - forward : boolean

**constructors**
- + LinkedArrayListIter2 ()
- + LinkedArrayListIter2 ( i : int )

**methods**
- + hasNext () : boolean
- + next () : T
- + hasPrevious () : boolean
- + previous () : T
- + remove () : void
- + set ( obj : T ) : void
- + add ( obj : T ) : void
- + nextIndex () : int
- + previousIndex () : int