Data Structure 2018 Lab 03

Mohammad Sadegh Najafi

Linear List

- A linear list is a list of finite numbers of elements stored in the memory.

- For representation, use a one-dimensional array element[]

- Each element of the Linear List is referred by an index set.

- We can access all these elements with the help of the index set.

Linear List Example

Mike	Rick	christ	Alex	Max	Dave	Roly	Daina
0	1	2	3	4	5	6	7

Link List operation

1. **isEmpty() -** Returns true if the List is empty ,false other wise.

2. **size() -** Give the number of element in the list.

3. **get(theIndex) -** Give the element with gives index.

4. **indexOf(theElement) -** Determines the index of a given element.

5. **remove(theIndex) -** Removes element with a given index and returns it.

6. **add(theIndex,theElement) -** Add a given element so that new element has a specified index.

Linear List Interface

```java
public interface LinearList {
    public boolean isEmpty();
    public int size();
    public Object get(int index);
    public int indexOf(Object elem);
    public Object remove(int index);
    public void add(int index, Object obj);
    public String toString();

}
```

Iterators

The java.util.Iterator interface provides the following methods:

- **boolean hasNext()** - Returns true if the iteration has more elements.

- **E next()** - Returns the next element in the iteration.

- **void remove()** - Removes from the underlying collection the last element returned by the iterator (optional operation).
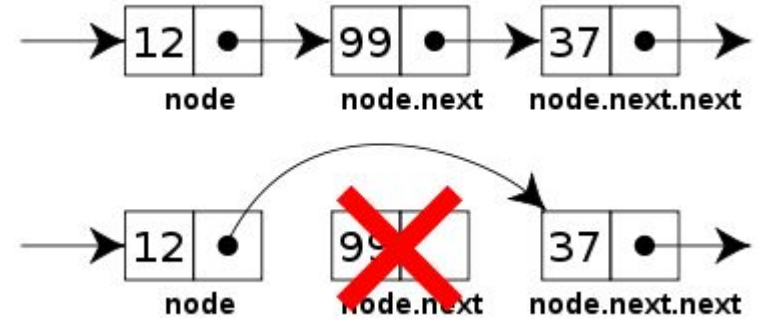
Linked Lists
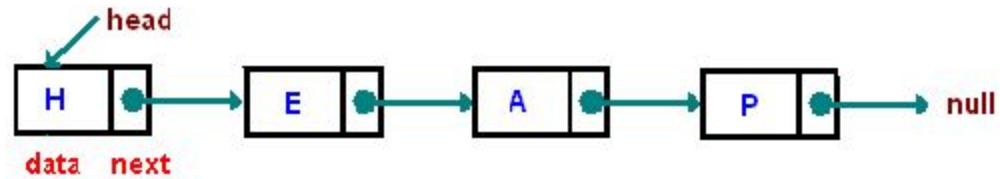
Linked List - add(v)

- if(head == null)
  - Create new node and set head
- if(v < head's v)
  - Create new node
  - Set new node's next to head
  - Set head to new node
- while(iterator's next != null)
  - if(v >= iterator's value && v <= iterator's next node's value) ?
- else?
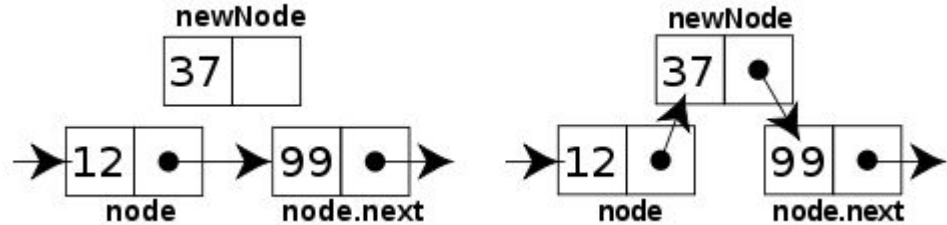
Linked List - remove(v)

- if(head == null)
- if(v == head's v)
- while(iterator's next != null)
  - if(iterator's next's value == v)
- if non?

Write a sorted LinkedList

- Finish writing MyLinkedList that implements all the method from LinkedList interface.

Expected output from Lab03.java

is Empty? true
size is 0
false
Index of 0: -1
2 4 4 5 7 10
size is 6
Index of 2: 0
Index of 7: 4
Index of 10: 5
true
false
1 4 5 10
is Empty? false
size is 4

LinkedList Interface

- public void add(int v);
  - creates new Node object which hold value v and insert into place where list will satisfy to remain sorted
- public boolean remove(int v);
  - Remove Node object that holds value v. If list doesn't have object that holds value v, then return false, otherwise true
- public void print();
  - Print each element in LinkedList
- public boolean isEmpty();
  - Return boolean of either is empty or not
- public int size();
  - Return size of list
- public int indexOf(int v);
  - Return index of Node that holds value v. If there is non, return -1