ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΑΣ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ
ΔΟΜΕΣ ΔΕΔΟΜΕΝΩΝ & ΑΝΑΛΥΣΗ ΑΛΓΟΡΙΘΜΩΝ

ΑΣΚΗΣΗ 4

ΣΤΟΙΒΕΣ (Stacks) και ΟΥΡΕΣ (Queues)
Υλοποίηση με απλά Συνδεδεμένη Λίστα

(Βλέπε http://www.iee.ihu.gr/~demos/teaching_GR.html)

Στην άσκηση αυτή θα χρησιμοποιήσετε τις κλάσεις του κόμβου και της απλά συνδεδεμένης λίστας που υλοποιήθηκαν στην προηγούμενη άσκηση 3.

- ***Node class*** η οποία περιγράφει ένα κόμβο μιας απλά συνδεδεμένης λίστας
- ***LinkedList class*** που υλοποιεί το **List interface**

Άσκηση 4.1

Να δώσετε μια υλοποίηση του ***Stack interface*** με χρήση της ***απλά συνδεδεμένης λίστας LinkedList***. Στη συνέχεια να ελέγξετε/τροποποιήσετε το πρόγραμμα της άσκησης 2 για τον έλεγχο παρενθέσεων ώστε να δουλεύει με την καινούργια υλοποίηση της στοίβας. (Δίνεται εκ νέου στη συνέχεια)

*Να γραφεί πρόγραμμα Java για τον έλεγχο της σωστής χρήσης των παρενθέσεων ( ) σε μία αριθμητική παράσταση χρησιμοποιώντας τη δομή δεδομένων στοίβα, η οποία θα υλοποιείται με **απλά συνδεδεμένη λίστα**.. Το πρόγραμμα να επιστρέφει μήνυμα εάν η αριθμητική παράσταση είναι σωστή ή εάν έχει λάθος, να αναφέρει το σημείο που βρήκε το λάθος.*

Άσκηση 4.2

Να δώσετε μια υλοποίηση του ***Queue interface*** με χρήση της της ***απλά συνδεδεμένης λίστας LinkedList.*** Στη συνέχεια να ελέγξετε/τροποποιήσετε το πρόγραμμα της άσκησης **2.7** ώστε να δουλεύει με την καινούργια υλοποίηση της ουράς. (Δίνεται εκ νέου στη συνέχεια)

*Να γραφεί πρόγραμμα Java για την εξυπηρέτηση αυτοκινήτων σε διόδια με την χρήση **ουράς**, η οποία θα υλοποιείται με **απλά συνδεδεμένη λίστα**. Πιο συγκεκριμένα θα εμφανίζεται το παρακάτω μενού:*

ΜΕΝΟΥ

1. *Άφιξη αυτοκινήτου*
2. *Αναχώρηση αυτοκινήτου*
3. *Κατάσταση ουράς*
4. *Έξοδος*

Επιλογή 1*: Θα πληκτρολογούνται τα στοιχεία του αυτοκινήτου π.χ. ο αριθμός αυτοκινήτου και θα τοποθετείται στο τέλος της ουράς.*
Επιλογή 2*: Το αυτοκίνητο που βρίσκεται πρώτο στην ουρά θα διαγράφεται μαζί με ένα ανάλογο μήνυμα επιβεβαίωσης.*
Επιλογή 3: *Θα εμφανίζονται με τη σειρά οι αριθμοί των αυτοκινήτων που παραμένουν στην ουρά για να εξυπηρετηθούν.*
Επιλογή 4: *Το πρόγραμμα θα τερματίζεται.*

Δίνονται εκ νέου:

```
public interface Stack
{
        public int size( );
        public boolean isEmpty( );
        public boolean isFull( );
        public Object top( ) throws StackEmptyException;
        public void push(Object item) throws StackFullException;
        public Object pop( ) throws StackEmptyException;
}
```

```
public interface Queue
{       public int size( );
        public boolean isEmpty( );
        public boolean isFull();
        public Object front( ) throws QueueEmptyException;
        public void add(Object item) throws QueueFullException;
        public Object remove( ) throws QueueEmptyException;
}
```

```
public interface List
{       public boolean isEmpty( );
        public int size( );
        public void insertFirst(Object data);
        public void insertLast(Object data);
        public Object removeFirst( ) throws ListEmptyException;
        public Object removeLast( ) throws ListEmptyException;
}
```