

# מעבדה 3. נושא: רשימה מקושרת

## חד-כיוונית

תאריך הגשה לקבוצה של יום ב': 16/11/2020

תאריך הגשה לקבוצה של יום ד': 18/11/2020

(הגשה בזוגות)

יש לקרוא היטב לפני תחילת העבודה!

### מבוא:

רשימה (List) הוא מבנה נתונים התומך בפעולות המתבצעות על רצף של איברים.

ניתן לממש רשימה בעזרת מערך (כפי שעשינו במעבדה 1) או בעזרת רשימה מקושרת חד-כיוונית.

במעבדה זו נעסוק במימוש של רשימה בעזרת רשימה מקושרת חד-כיוונית.

הערה: שימו לב כי אין להשתמש בספריית אוספים כלשהי ובפרט בספריית האוספים התקנית ב `java.util`.

### שלב 1.

ראשית, ניצור מחלקה של איברי הרשימה (NODES). כל איבר יכיל מפתח (**element**) ומצביע לאיבר הבא ברשימה (**next**). להלן המחלקה:

```
//Facilitator class for the LinkedList class
public class SNode<T> {
    private T element; // element in the list
    private SNode<T> next; // reference to the next element

    public SNode(T element, SNode<T> next) { }

    public T getElement() { }
    public void setElement(T newElem) { }
    public SNode<T> getNext() { }
    public void setNext (SNode<T> nextVal) { }
}
```

שים לב כי `<T>` הוא שם של טיפוס כלשהו (יש להניח כי הטיפוס הוא מחלקה).

צור קובץ בשם `SNode.java`, תעתיק אליו את המחלקה המופיעה לעיל וממש בו את כל המטודות של המחלקה.

## List<T>

התומך בפעולות הבאות:

### void insert ( T newElement )

Precondition: newElement is not null.

Postcondition: Inserts newElement into a list. If the list is not empty, then inserts newElement **after** the cursor. Otherwise, inserts newElement as the **first** (and only) element in the list. **In either case, moves the cursor to newElement.**

### void remove ( )

Precondition: List is not empty.

Postcondition: Removes the element **marked by the cursor** from a list. If the resulting list is not empty, then moves the cursor to the element that followed the deleted element. If the deleted element was **at the end** of the list, then moves the cursor to the beginning of the list.

### void replace ( T newElement )

Precondition: List is not empty and newElement is not null.

Postcondition: Replaces the **element marked by the cursor** with newElement. The cursor remains at newElement.

### void clear ( )

Precondition: None.

Postcondition: Removes all the elements in a list.

### boolean isEmpty ( )

Precondition: None.

Postcondition: Returns true if a list is empty. Otherwise, returns false.

### boolean gotoBeginning ( )

Precondition: None.

Postcondition: If a list is not empty, then moves the cursor to the beginning of the list and returns true. Otherwise, returns false.

### boolean gotoEnd ( )

Precondition: None.

Postcondition: If a list is not empty, then moves the cursor to the end of the list and returns true. Otherwise, returns false.

### boolean gotoNext ( )

Precondition: List is not empty.

Postcondition: If the cursor is not at the end of a list, then moves the cursor to the next element in the list and returns true. Otherwise, returns false.

### boolean gotoPrior ( )

Precondition: List is not empty.

Postcondition: If the cursor is not at the beginning of a list, then moves the cursor to the preceding element in the list and returns true. Otherwise, returns false.

### T getCursor ( )

Precondition: List is not empty.

Postcondition: Returns the element marked by the cursor.

### שלב 3.

כתוב מחלקה בשם

#### **SLinkedList<T>**

**הממשת** את המנשק הנ"ל בעזרת רשימה מקושרת חד-כיוונית ומכילה:

1. מצביע לתחילת הרשימה (**head**)

```
private SNode<T> head // Reference to the beginning of the list
```

ולאיבר הנוכחי המסומן ע"י הסמן (**cursor**):

```
private SNode<T> cursor; // Reference to current cursor position
```

(2). **בנאי** אחד ללא פרמטרים.

(3). מממשת **toString** משלה

### שלב 4. בדיקות

א). צור מחלקה בשם TestSListChar.java שבעזרת tokenizer תומכת בפעולות הבאות ובודקת את עבודתו של SLinkedList **עבור האיברים שהם תווים (characters)**:

פקודה	פעולה
Add x	מוסיף איבר x <u>אחרי</u> הסמן
Del	מוחק איבר המסומן ע"י הסמן
Mv x	מחליף איבר המסומן ע"י הסמן עם איבר x
Cur	מראה את האיבר המוצבע ע"י הסמן
N	מזיז את הסמן לאיבר הבא (next)
P	מזיז את הסמן לאיבר הקודם (previous)
B	מזיז את הסמן לתחילת הרשימה (beginning)
End	מזיז את הסמן לסוף הרשימה (end)
E	שאלתא: האם הרשימה ריקה (empty)
C	מוחק את כל הרשימה (clear)
Print	מדפיס את כל הרשימה
Q	סיים את התוכנית

ב). חזור על סעיף 4א) עבור רשימת מספרים שלמים (integers). יש ליצור קובץ **TestSListInt.java**.

### הגשה:

יש להגיש את הקבצים הבאים:

**SNode.java , List.java, SLinkedList.java, TestSListChar.java  
TestSListInt.java**

## ערעורים:

יש להפנות לד"ר תמר צמח בלבד ע"י שליחת מייל [tamar.zemach@yahoo.com](mailto:tamar.zemach@yahoo.com) תוך שבוע ממועד פרסום הציונים. בכותרת המייל יש לציין: "ערעור מעבדה מס' X ע"י +ת.ז. של הסטודנטים". יש לקחת בחשבון שבעת הערעור העבודה נבדקת מחדש וכתוצאה מהערעור ציון העבודה עלול להשתנות (יכול לעלות או לרדת). התשובה לערעור תהיה סופית ולא ניתנת לערעור נוסף.