

תכנות מונחה עצמים מטלה 0

מטלה זו מוקדשת לחזרה על נושאים שנלמדו בקורסים קודמים של מבוא לחישוב, ומבנה נתונים. בניגוד למטלות הבאות מטלה מוגשת דרך המודל (כל המטלות הבאות יוגשו כפרויקט ב Github).

הנחיות כלליות:

1. את המטלה עושים **בזוגות**, יש להגיש את כל המטלות בזמן לפי הנחיות, על כל איחור לא מוצדק תהיה הורדת ניקוד.
2. המטלות תיבדקנה באופן אוטומטי באספקטים של "העתיקות קוד" אין לבצע שום העתקה של קודים בין קבוצות שונות, מותר לעשות שימוש בקוד פתוח, אבל חובה לציין זאת בפירוש ולהביא את המקור המדויק. למען הסר ספק: שימוש בקוד פתוח (או כל קוד זמין ברשת) שלא יצוין מקור הקוד יחשב כהעתקה!
3. יש לתעד את כל המתודות של הפרויקט שלכם בעזרת **javadoc**.
4. **צריך להגיש קובץ ZIP שם הקובץ - מספר זהות ראשון מקו תחתון מספר זהות שני.** יש לקבץ קבצי JAVA בלבד.

Part 1 – the Problem

Write your own `StringBuilder` class with support for the undo operation. To do this, delegate all methods to the standard `StringBuilder`, and keep a list of all operations to perform `undo()` in your own class. An **`undo()`** method must be written for the following methods:

1. **`public UndoableStringBuilder append(String str)`**
Appends the specified string to this character sequence.
2. **`public UndoableStringBuilder delete(int start, int end)`**
Removes the characters in a substring of this sequence. The substring begins at the specified start and extends to the character at index end - 1 or to the end of the sequence if no such character exists. If start is equal to end, no changes are made.
3. **`public UndoableStringBuilder insert(int offset, String str)`**
Inserts the string into this character sequence.
4. **`public UndoableStringBuilder replace(int start, int end, String str)`**
Replaces the characters in a substring of this sequence with characters in the specified String. The substring begins at the specified start and extends to the character at index end - 1 or to the end of the sequence if no such character exists. First the characters in the substring are removed and then the specified String is inserted at start. (This sequence will be lengthened to accommodate the specified String if necessary).
5. **`public UndoableStringBuilder reverse()`**
Causes this character sequence to be replaced by the reverse of the sequence.

Part 2

Look at the attached code of main, it includes a simple example to test the your code. Be sure to document your code well (javadoc) and add the relevant tests (JUnit).

```
public static void main(String[] args){
    UndoableStringBuilder usb = new UndoableStringBuilder();
    usb.append("to be or not to be");
    System.out.println(usb);
    usb.replace(3, 5, "eat");
    System.out.println(usb);
    usb.replace(17, 19, "eat");
    System.out.println(usb);
    usb.reverse();
    System.out.println(usb);
    System.out.println();
    usb.undo();
    System.out.println(usb);
    usb.undo();
    System.out.println(usb);
    usb.undo();
    System.out.println(usb);
}
```

OUTPUT:

```
to be or not to be
to eat or not to be
to eat or not to eat
tae ot ton ro tae ot

to eat or not to eat
to eat or not to be
to be or not to be
```