

2. Consider the following incomplete declaration of a `LineEditor` class that allows insertions and deletions in a line of text. The line of text is stored internally as a `String`. The `insert` operation takes a `String` and inserts it into the line of text at the given index. The `delete` operation takes a `String` parameter and removes the first occurrence (if any) of that string from the line of text. The `deleteAll` operation removes all occurrences (if any) of a given `String` from the line of text, including any that are formed as a result of the deletion process.

```
public class LineEditor
{
    private String myLine;

    /** Inserts str into myLine at position index;
     *   no characters from myLine are overwritten.
     *   @param str the string to be inserted
     *   @param index the position at which to insert str
     *   Precondition:  $0 \leq \text{index} \leq \text{myLine.length}()$ 
     */
    public void insert(String str, int index)
    { /* to be implemented in part (a) */ }

    /** If str is found in myLine the first occurrence of str has been
     *   removed from myLine; otherwise myLine is left unchanged.
     *   @param str the string to be removed
     */
    public void delete(String str)
    { /* to be implemented in part (b) */ }

    /** Removes all occurrences of str from myLine;
     *   myLine is otherwise unchanged.
     *   @param str the string to be removed
     */
    public void deleteAll(String str)
    { /* to be implemented in part (c) */ }

    // There may be instance variables, constructors, and methods that are not
    // shown.
}
```

- (a) Write the `LineEditor` method `insert` as described at the beginning of the question. The following tables show the result of several different calls to `insert`.

<u>Method call: <code>insert("A.P.", 0)</code></u>	
<u>myLine before the call</u>	<u>myLine after the call</u>
"Computer Science"	"A.P.Computer Science"

<u>Method call: <code>insert(" is best", 16)</code></u>	
<u>myLine before the call</u>	<u>myLine after the call</u>
"Computer Science"	"Computer Science is best"

<u>Method call: <code>insert("Java", 4)</code></u>	
<u>myLine before the call</u>	<u>myLine after the call</u>
"Computer Science"	"CompJavauter Science"

Complete method `insert` below. Assume that the other class methods work as specified.

```
/** Inserts str into myLine at position index;
 * no characters from myLine are overwritten.
 * @param str the string to be inserted
 * @param index the position at which to insert str
 * Precondition:  $0 \leq \text{index} \leq \text{myLine.length}()$ 
 */
public void insert(String str, int index)
```

- (b) Write the `LineEditor` method `delete` as described at the beginning of the question. The following table shows the result of several different calls to `delete`.

Method call: `delete("Com")`

<u>myLine before the call</u>	<u>myLine after the call</u>
"Computer Science"	"puter Science"

Method call: `delete("ter Sc")`

<u>myLine before the call</u>	<u>myLine after the call</u>
"Computer Science"	"Compuience"

Method call: `delete("c")`

<u>myLine before the call</u>	<u>myLine after the call</u>
"Computer Science"	"Computer Sience"

Method call: `delete("Java")`

<u>myLine before the call</u>	<u>myLine after the call</u>
"Computer Science"	"Computer Science"

Complete method `delete` below.

```
/** If str is found in myLine the first occurrence of str has been
 * removed from myLine; otherwise myLine is left unchanged.
 * @param str the string to be removed
 */
public void delete(String str)
```

- (c) Write the `LineEditor` method `deleteAll` as described at the beginning of the question. The following table shows the result of several different calls to `deleteAll`.

Method call: <code>deleteAll("ing")</code>	
<u>myLine before the call</u>	<u>myLine after the call</u>
"string programming"	"str programm"

Method call: <code>deleteAll("r")</code>	
<u>myLine before the call</u>	<u>myLine after the call</u>
"string programming"	"sting pogamming"

Method call: <code>deleteAll("aba")</code>	
<u>myLine before the call</u>	<u>myLine after the call</u>
"abababa"	"b"

Method call: <code>deleteAll("oh-la")</code>	
<u>myLine before the call</u>	<u>myLine after the call</u>
"ooh-lah-lah"	"h"

Method call: <code>deleteAll("zap")</code>	
<u>myLine before the call</u>	<u>myLine after the call</u>
"pizza pie"	"pizza pie"

In writing `deleteAll`, you may call any of the methods in the `LineEditor` class, including `insert` and `delete` from parts (a) and (b). Assume that these methods work as specified, regardless of what you wrote in parts (a) and (b).

Complete method `deleteAll` below.

```
/** Removes all occurrences of str from myLine;
 * myLine is otherwise unchanged.
 * @param str the string to be removed
 */
public void deleteAll(String str)
```