

Einführung in die Programmierung
WS 2012/13

Tutorium 2: Strings

Besprechung: 10.01.2013

Ende der Abgabefrist: Keine Abgabe.

Hinweise:

- Im Tutorium besprochene Aufgaben sind prüfungsrelevant für Nebenfachstudierende mit 60 ECTS.
- Eine Abgabe und Korrektur des Tutoriums findet **nicht** statt.

Aufgabe 2-1 *String vs. StringBuilder/StringBuffer*

Als angehende Akademiker sollten Sie stets bereit sein, die Äußerungen Ihres Dozenten zu hinterfragen. Kann es wirklich sein, dass die Konkatenation von Strings problematisch ist? Das ist doch eine Operation, die ständig vorkommt und sogar mit einem eigenen Operator unterstützt wird!

Zur Überprüfung der Effizienz von `String`-Konkatenationen gegenüber der Verwendung von `StringBuilder` und `StringBuffer` schreiben Sie in einer Klasse `StringEffizienz` drei Methoden

- (a) `public static String buildString(int n)`
- (b) `public static String buildStringBuilder(int n)`
- (c) `public static String buildStringBuffer(int n)`

die jeweils einen `String` zurückgeben, der die Zahl 10^n für $n \geq 0$ darstellt, also einen `String` bestehend aus dem Zeichen '1' n -mal gefolgt von dem Zeichen '0'. Die n -fache Wiederholung soll in jeder der drei Methoden gleichartig als `for`-Schleife implementiert sein, nur wird der `String` in den verschiedenen Methoden unterschiedlich gebildet:

- `buildString` verwendet fortgesetzte `String`-Konkatenation
 - `buildStringBuilder` verwendet einen `StringBuilder`, um die Zeichenkette aufzubauen
 - `buildStringBuffer` verwendet einen `StringBuffer`, um die Zeichenkette aufzubauen
- (d) Implementieren Sie in der `main`-Methode der Klasse einen Effizienz-Test zur Untersuchung des Laufzeitverhaltens der Methoden. Dazu können Sie natürlich Anleihen aus der Klasse `Laufzeit` vom letzten Übungsblatt nehmen. Pro Durchlauf der Klasse `StringEffizienz` soll jeweils nur eine der drei Methoden getestet werden, d.h. in der `main`-Methode darf jeweils nur eine Methode aufgerufen werden.

