

Übungen zu Einführung in die Informatik

Aufgabe 2-1 Datum (29./30. 10. 07)

Ein Datum soll durch Angabe des Monats, gefolgt von einem Bindestrich und der Angabe des Tages dargestellt werden. Beispiel: 7. November wird als 11-07 dargestellt. Bitte beachten sie, dass z.B. 02-30 und 20-10 keine gültigen Daten sind. Der Einfachheit halber nehmen wir an, dass der Monat Februar immer 28 Tage hat.

- a) Geben Sie eine EBNF-Grammatik an, die die Menge der korrekten Daten beschreibt.
- b) Geben Sie Ableitungen für die Daten 11-07 und 03-23 an.

Aufgabe 2-2 Java Kommentare in BNF (29./30. 10. 07)

In Java gibt es drei Arten von Kommentaren:

- **Einzeilen-Kommentare** : sie werden durch zwei Schrägstriche (/) eingeleitet und enden durch ein Zeilenendsymbol.
 - **Block-Kommentare** : sie werden durch Schrägstrich (/) und Stern (*) eingeleitet und durch Stern und Schrägstrich beendet. Dazwischen können beliebige¹ Zeichenketten stehen.
 - **Javadoc-Kommentare** : sie werden durch Schrägstrich (/) und zwei Sterne (*) eingeleitet und verhalten sich sonst wie Block-Kommentare².
- a) Modellieren Sie in BNF-Notation das Nicht-Terminal **Kommentar**, welches alle Arten von Java-Kommentaren akzeptiert. Gerne dürfe Sie weitere Hilfs-Nicht-Terminale einführen. Betrachten Sie die Nicht-Terminale **Buchstabe**, **Ziffer**, **Symbol**, **Leerzeichen** sowie **Zeilenumbruch** als gegeben, d.h. verwenden Sie diese Nicht-Terminale in Ihren BNF-Regeln.
 - b) Geben Sie für den Kommentar “ `/* Version 2.0 */` ” die Ableitung an, die Ihre BNF Regeln von oben liefern.

Aufgabe 2-3 Arithmetische Ausdrücke (5./6. 11. 07)

- a) Entwerfen Sie eine BNF-Grammatik für arithmetische Ausdrücke über den natürlichen Zahlen und den (binären) Operatoren + und * mit beliebiger, aber korrekt geschachtelter Klammerung. Dabei darf die in der Vorlesung vorgestellte BNF-Grammatik für natürliche Zahlen verwendet werden.
- b) Geben Sie eine Ableitung für "4" "2" "+" "3" "*" "(" "9" "9" "+" "1" ")" "+" "2" an.

Aufgabe 2-4 Umkehrung einer Zeichenkette (5./6. 11. 07)

Programmieren Sie eine Klasse mit der folgenden Funktionalität:

```
public static String reverseString(String toReverse)
```

Dabei soll eine beliebige Zeichenkette übergeben und als Resultat der übergebene String in umgekehrter Reihenfolge zurückgegeben werden.

Beispiel:

¹Ansich sollte man sagen, dass beliebige Zeichenketten ohne */ vorkommen dürfen – denn diese beiden Zeichen beenden ja den Kommentar. Diese Ausnahme braucht allerdings nicht explizit von Ihnen behandelt zu werden.

²Im Grunde genommen gibt es nur 2 (disjunkte) Arten von Kommentaren, wenn man bedenkt, dass Javadoc-Kommentare selber alle auch Blockkommentare sind.

Übergabeparameter: Hallo

Rückgabewert: ollaH

Hinweis: Die Methode `char charAt(int index)` der Klasse `String` gibt das `index`-te Zeichen einer Zeichenkette zurück. Beispiel: Falls der Wert vom `String s` gleich "abc" ist, dann ist der Wert von `s.charAt(0)` gleich 'a' und `s.charAt(1)` ist gleich 'b'.

Siehe auch [http://java.sun.com/j2se/1.5.0/docs/api/java/lang/String.html#charAt\(int\)](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/String.html#charAt(int)) .

Aufgabe 2-5

Palindrom

(5./6. 11. 07)

Laden Sie die Klasse `PalindromDemo` von unserer Web-Seite herunter und ergänzen Sie sie wie folgt:

- a) Implementieren Sie die Methode `public static String convertString(String toConvert)` der Klasse `PalindromDemo`. Dieser Methode wird eine Zeichenkette als Parameter übergeben. Aus der übergebenen Zeichenkette sollen alle Zeichen entfernt werden, die keine Buchstaben sind. Großbuchstaben sollen durch entsprechende Kleinbuchstaben ersetzt werden. Der daraus resultierende `String` soll zurückgegeben werden.

Beispiel:

Übergabeparameter: `Dentist: "Sit, Ned."`

Rückgabewert: `dentistsitned`

Hinweis

[http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Character.html#isLetter\(char\)](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Character.html#isLetter(char)),

[http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Character.html#isUpperCase\(char\)](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Character.html#isUpperCase(char)),

und

[http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Character.html#toLowerCase\(char\)](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Character.html#toLowerCase(char))

- b) Implementieren Sie die Methode `public static boolean isPalindrom(String toTest)` der Klasse `PalindromDemo`. Der übergebene Parameter soll zunächst durch `convertString` konvertiert werden. Daraufhin soll überprüft werden, ob es sich bei dieser Zeichenkette um ein Palindrom handelt. Bei positivem Ergebnis soll `true` zurückgegeben werden, sonst `false`.

Beispiele:

- Übergabeparameter: `Reliefpfeiler`
Rückgabewert: `true`, da `Reliefpfeiler` ein Palindrom ist, wenn man die Groß/Kleinschreibung außer Acht läßt.
- Übergabeparameter: `Dentist: "Sit, Ned."`
Rückgabewert: `true`, da `dentistsitned` ein Palindrom ist.
- Übergabeparamter: `kein Palindrom`
Rückgabewert: `false`. Sie wissen ja warum.

Hinweis Wir definieren auch die leere Zeichenkette als Palindrom.

- c) Testen Sie Ihr Programm. Auf <http://www.palindromelist.com/> finden Sie viele Palindrome.