

Übungen zu Einführung in die Informatik: Programmierung und Software-Entwicklung: Lösungsvorschlag

Aufgabe 2-1 EBNF-Grammatik für Spielzeugkisten *Präsenz*

Im Folgenden betrachten wir eine einfache Sprache, mit der man den Inhalt von Spielzeugkisten in einem Kinderzimmer beschreiben kann. Generell wird dabei eine Spielzeugkiste durch das Wort "Kiste" angegeben und anschließend wird in runden Klammern entweder "leer" geschrieben oder es wird der Inhalt der Spielzeugkiste aufgezählt. Der Inhalt einer Spielzeugkiste besteht aus einer Reihe von Gegenständen, wobei zwischen jeweils zwei (aneinander grenzenden) Gegenständen immer angegeben werden soll, ob der erste "neben" oder "auf" dem zweiten liegt. Ein Gegenstand ist entweder ein Spielzeug oder wiederum eine Spielzeugkiste, die ihrerseits (wie oben beschrieben) angegeben wird. Spielzeuge sind "Puppe", "Auto", "Ball" oder "Teddy".

Beispiele für gültige Ausdrücke dieser Sprache sind also:

Kiste (leer)

Kiste (Puppe neben Auto auf Kiste (Teddy neben Auto) neben Ball)

Ungültig sind Konstruktionen, in denen vor oder nach einer Präposition kein Gegenstand steht, also z.B.:

Kiste (neben Auto)

Kiste (Auto neben)

- a) Definieren Sie eine EBNF-Grammatik für die oben beschriebene Sprache. Das Startsymbol soll den Namen `Spielzeugkiste` haben.

Mögliche Lösung:

```
Spielzeugkiste = "Kiste" "(" ( "leer" | Inhalt ) ")"  
Inhalt = Gegenstand { Praeposition Gegenstand }  
Gegenstand = Spielzeug | Spielzeugkiste  
Spielzeug = "Puppe" | "Ball" | "Auto" | "Teddy"  
Praeposition = "neben" | "auf"
```

- b) Leiten Sie mit der oben definierten EBNF-Grammatik folgenden Ausdruck ab:

Kiste (Auto auf Kiste (leer))

Hinweis: Geben Sie sowohl die lange als auch die kurze Ableitung des Ausdrucks an. Geben Sie in jedem Schritt an, ob Sie eine Regel angewendet haben und welche Operatoren Sie angewendet haben!

Lange Lösung:

Spielzeugkiste

```
-(Regel)-> "Kiste" "(" ( "leer" | Inhalt ) ")"
-(Op |)--> "Kiste" "(" Inhalt ")"
-(Regel)-> "Kiste" "(" Gegenstand { Praeposition Gegenstand } ")"
-(Regel)-> "Kiste" "(" ( Spielzeug | Spielzeugkiste )
    { Praeposition Gegenstand }
    ")"
-(Op {})-> "Kiste" "(" ( Spielzeug | Spielzeugkiste )
    Praeposition Gegenstand
    { Praeposition Gegenstand }
    ")"
-(Op {})-> "Kiste" "(" ( Spielzeug | Spielzeugkiste )
    Praeposition Gegenstand
    ")"
-(Op |)--> "Kiste" "(" Spielzeug Praeposition Gegenstand ")"
-(Regel)-> "Kiste" "(" ( "Puppe" | "Ball" | "Auto" | "Teddy" )
    ("neben" | "auf")
    ( Spielzeug | Spielzeugkiste )
    ")"
-(Op |)--> "Kiste" "(" "Auto" "auf" Spielzeugkiste ")"
-(Regel)-> "Kiste" "(" "Auto" "auf" "Kiste" "(" ( "leer" | Inhalt ) ")" ")"
-(Op |)--> "Kiste" "(" "Auto" "auf" "Kiste" "(" "leer" ")" ")"
```

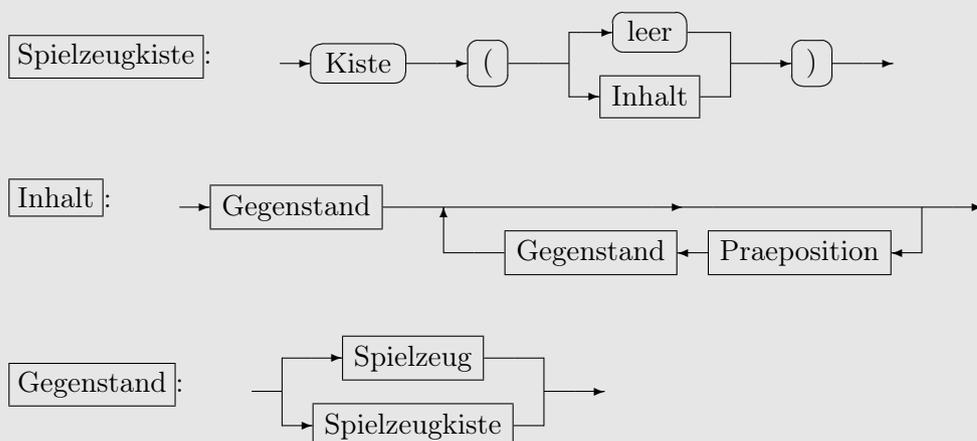
Kurze Lösung:

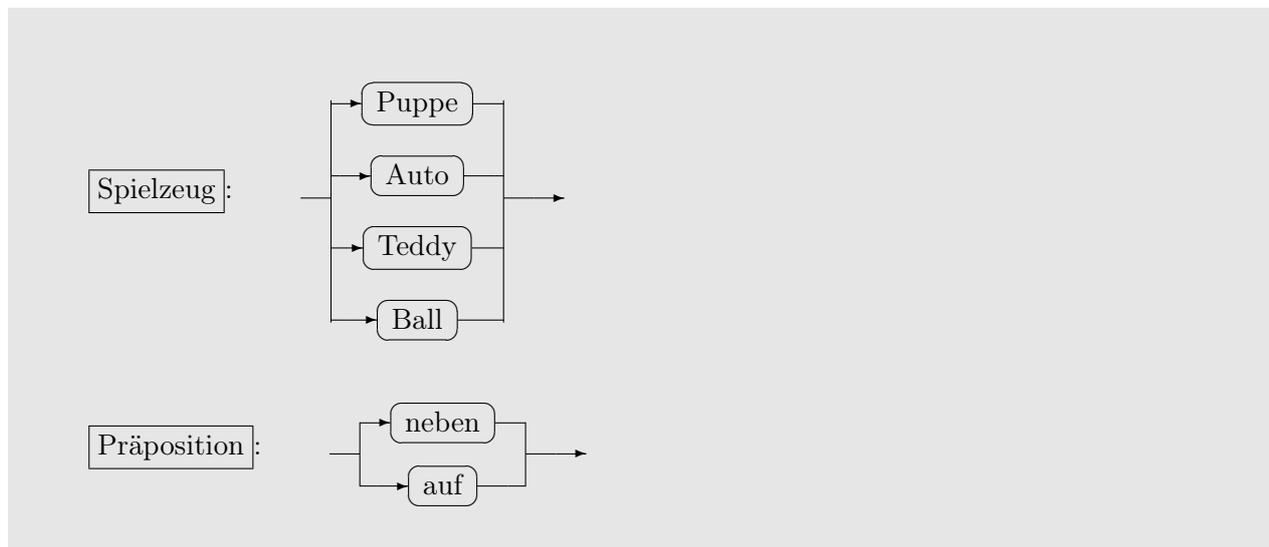
Spielzeugkiste

```
-(Regel, Op |)--> "Kiste" "(" Inhalt ")"
-(Regel, Op {})-> "Kiste" "(" Gegenstand Praeposition Gegenstand
    { Praeposition Gegenstand }
    ")"
-(Op {})-> "Kiste" "(" Gegenstand Praeposition Gegenstand ")"
-(Regel, Op |)--> "Kiste" "(" Spielzeug "auf" Spielzeugkiste ")"
-(Regel, Op |)--> "Kiste" "(" "Auto" "auf" "Kiste" "(" "leer" ")" ")"
```

- c) Erstellen Sie zu der oben definierten EBNF-Grammatik ein äquivalentes Syntaxdiagramm.

Mögliche Lösung:





Aufgabe 2-2

EBNF-Grammatik für Backrezepte

Hausaufgabe

In einem Backrezept wollen wir beschreiben können, welche Zutaten und wie viel davon für ein Gebäck gebraucht werden. Dazu geben wir an, wie viele Teelöffel "TL" oder Esslöffel "EL" einer Zutat benötigt werden. Die Anzahl der Tee- oder Esslöffel muss eine positive ganze Zahl ungleich 0 sein. Als Zutaten können wir "Salz", "Zucker", "Mehl" und "Milch" verwenden. Ein Backrezept besteht dann aus beliebig vielen, aber mindestens einer solchen Mengenangabe.

Beispiele für gültige Backrezepte sind also:

5 EL Zucker 10 EL Mehl 20 TL Milch

1 TL Salz 1 TL Salz

Ungültig sind beispielsweise folgende Konstruktionen:

0 TL Salz

5 TL 3 EL Milch

10 EL Mehl Zucker

- a) Definieren Sie eine EBNF-Grammatik für die oben beschriebene Sprache. Das Startsymbol soll den Namen `Backrezept` haben. Sie können folgende Regeln verwenden:

```

Zahl = ZifferOhne0 { Ziffer }
ZifferOhne0 = "1" | "2" | ... | "9"
Ziffer = "0" | ZifferOhne0
  
```

Mögliche Lösung:

```

Backrezept = Mengenangabe { Mengenangabe }
Mengenangabe = Zahl Masseinheit Zutat
Masseinheit = "TL" | "EL"
Zutat = "Salz" | "Zucker" | "Mehl" | "Milch"
  
```

- b) Leiten Sie mit der oben definierten EBNF-Grammatik folgenden Ausdruck ab:

1 TL Zucker 3 EL Mehl

Hinweis: Geben Sie sowohl die lange als auch die kurze Ableitung des Ausdrucks an. Geben Sie in jedem Schritt an, ob Sie eine Regel angewendet haben und welche Operatoren Sie angewendet haben!

Lange Lösung:

Backrezept

```

-(Regel)-----> Mengenangabe { Mengenangabe }
-(Regel)-----> Zahl Masseinheit Zutat { Mengenangabe }
-(Regel)-----> ZifferOhne0 { Ziffer }
                    ("TL" | "EL")
                    ("Salz" | "Zucker" | "Mehl" | "Milch")
                    { Mengenangabe }
-(Op {}, Op |)-> ZifferOhne0 "TL" "Zucker" { Mengenangabe }
-(Regel)-----> ("1" | "2" | ... | "9") "TL" "Zucker" { Mengenangabe }
-(Op |)-----> "1" "TL" "Zucker" { Mengenangabe }
-(Op {})-----> "1" "TL" "Zucker" Mengenangabe { Mengenangabe }
-(Regel)-----> "1" "TL" "Zucker" Zahl Masseinheit Zutat { Mengenangabe }
-(Regel)-----> "1" "TL" "Zucker"
                    ZifferOhne0 { Ziffer }
                    ("TL" | "EL")
                    ("Salz" | "Zucker" | "Mehl" | "Milch") { Mengenangabe }
-(Op |)-----> "1" "TL" "Zucker" ZifferOhne0 { Ziffer } "EL" "Mehl"
                    { Mengenangabe }
-(Op {})-----> "1" "TL" "Zucker" ZifferOhne0 "EL" "Mehl"
                    { Mengenangabe }
-(Regel)-----> "1" "TL" "Zucker" ("1" | "2" | ... | "9") "EL" "Mehl"
                    { Mengenangabe }
-(Op |)-----> "1" "TL" "Zucker" "3" "EL" "Mehl" { Mengenangabe }
-(Op {})-----> "1" "TL" "Zucker" "3" "EL" "Mehl"

```

Kurze Lösung:

Backrezept

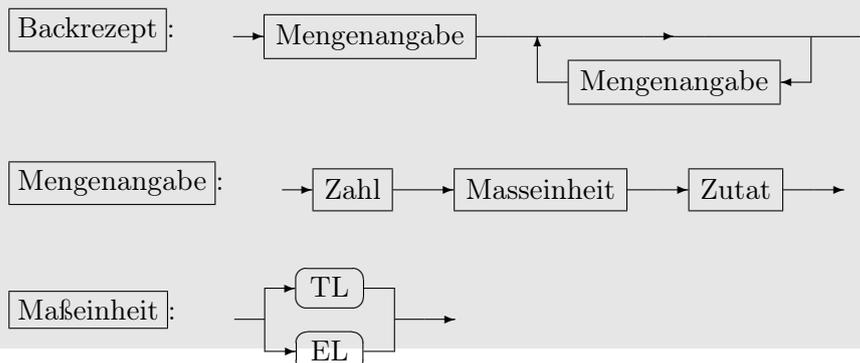
```

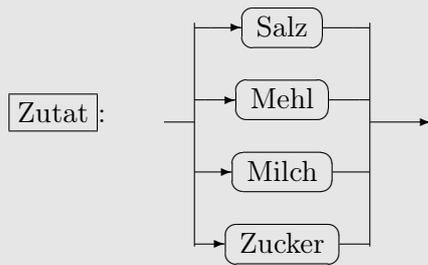
-(Regel, Op {})-> Mengenangabe Mengenangabe { Mengenangabe }
-(Op {})-> Mengenangabe Mengenangabe
-(Regel)-----> Zahl Masseinheit Zutat Zahl Masseinheit Zutat
-(Regel, Op {}, Op |)-> ZifferOhne0 "TL" "Zucker" ZifferOhne0 "TL" "Mehl"
-(Regel, Op |)-> "1" "TL" "Zucker" "3" "EL" "Mehl"

```

- c) Erstellen Sie zu der oben definierten EBNF-Grammatik ein äquivalentes Syntaxdiagramm. Dabei brauchen Sie die Regeln für `Zahl`, `ZifferOhne0` und `Ziffer` nicht beachten.

Mögliche Lösung:





Besprechung der Präsenzaufgaben in den Übungen ab 27.10.2017. Abgabe der Hausaufgaben bis Mittwoch, 08.11.2017, 14:00 Uhr über UniWorX (siehe Folien der ersten Zentralübung).

- *Sie können Ihre Lösungen zu den Übungsaufgaben **freiwillig** zur Korrektur abgeben. Die Aufgaben haben keinen Einfluss auf Klausurzulassung oder Noten. Die Aufgaben dienen dazu, Ihnen eine intensive Beschäftigung mit dem Stoff der Vorlesung und eine gute Betreuung zu ermöglichen. Die Abgabe zur Korrektur erfolgt über das UniWorX-System. Hinweise zur Anmeldung finden Sie auf den Folien zur ersten Zentralübung.*
- *Erstellen Sie alle nötigen Dateien in einem eigenen Ordner (für jedes Übungsblatt ein eigener Ordner) und geben Sie den gesamten Ordner als ZIP-Archiv ab. In dem Ordner dürfen nur Dateien mit der Endung .java, .pdf, .jpg oder .txt enthalten sein. **Word-Dokumente werden nicht korrigiert!***
- *Unter Windows kann ein ZIP-Archiv wie folgt erstellt werden: rechter Mausklick auf den Ordner, Auswahl von **Senden an** -> **ZIP-komprimierter Ordner**. Unter Mac OS hingegen: rechter Mausklick auf den Ordner, Auswahl von **Komprimieren/Compress**.*