



## Musterlösungen zum Übungsblatt 10

### Lösung

#### 1. $\lambda$ -Kalkül: frei und gebunden

$v$  heißt gebunden in  $\lambda v.G$  durch die Bindung  $\lambda v$ , falls  $v$  in  $G$  auftritt.

$v$  heißt (Bezeichner-) frei in  $\lambda v.G$  falls  $v$  nicht in  $G$  auftritt.

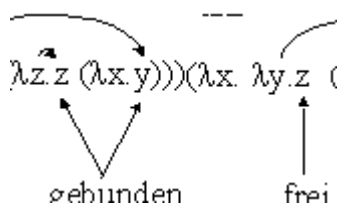
Ein Auftreten von  $v$  in  $G$  heißt in  $\lambda v.G$  gebunden durch die Bindung  $\lambda v$ . Kommt  $v$  in  $F$  außerhalb einer  $\lambda$ -Abstraktion  $\lambda v.G$  vor, so heißt  $v$  (Bezeichner-) frei in  $F$ .

a)

-  $(\lambda x. \lambda y. z (\lambda z. z (\lambda x. y)))$ :  $z$  ist frei, da es außerhalb von  $\lambda z$  vorkommt

-  $(\lambda x. \lambda y. z (\lambda z. z (\lambda x. y)))$ :  $z$  ist gebunden

-  $(\lambda x. \lambda y. z (\lambda z. z (\lambda x. y)))$ :  $y$  ist gebunden



b)

-  $(\lambda x. (\lambda y. x z (y z))) (\lambda x. y (\lambda y. y))$ :  $z$  ist frei, da es außerhalb von  $\lambda z$  vorkommt.

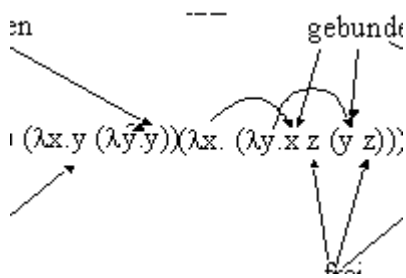
-  $(\lambda x. (\lambda y. x z (y z))) (\lambda x. y (\lambda y. y))$ :  $z$  ist frei, da es außerhalb von  $\lambda z$  vorkommt.

-  $(\lambda x. (\lambda y. x z (y z))) (\lambda x. y (\lambda y. y))$ :  $y$  ist frei, da es im Argument der 1.  $\lambda$ -Abstraktion vorkommt und so nicht von  $\lambda y$  gebunden wird.

-  $(\lambda x. (\lambda y. x z (y z))) (\lambda x. y (\lambda y. y))$ :  $x$  ist gebunden

-  $(\lambda x. (\lambda y. x z (y z))) (\lambda x. y (\lambda y. y))$ :  $y$  ist gebunden

-  $(\lambda x. (\lambda y. x z (y z))) (\lambda x. y (\lambda y. y))$ :  $y$  ist gebunden



## 2.1 -Kalkül: Auswertung

a)

$$(\lambda x. (\lambda y. - y x)) 4 5 \stackrel{(\beta)}{=} (\lambda y. - y x)[4/x] 5 = \beta\text{-Konversion}$$

$$(\lambda y. - y 4) 5 \stackrel{(\beta)}{=} (- y 4) [5/y] = \beta\text{-Konversion}$$

$$- 5 4 =$$

1

b)

$$(\lambda f. (\lambda x. f 4 x)) (\lambda y. (\lambda x. + x y)) 3 \stackrel{(\beta)}{=} (\lambda x. f 4 x) [(\lambda y. \lambda x. + x y)/f] 3 = \beta\text{-Konversion}$$

$$(\lambda x. (\lambda y. (\lambda x'. + x y)) 4 x) 3 \stackrel{(\alpha)}{=} (\lambda x. (\lambda y. (\lambda x'. + x y)) [x'/x] 4 x) 3 = \alpha\text{-Konversion}$$

$$(\lambda x. (\lambda y. (\lambda x'. + x' y)) 4 x) 3 \stackrel{(\beta)}{=} ((\lambda y. (\lambda x'. + x' y)) 4 x) [3/x] = \beta\text{-Konversion}$$

$$(\lambda y. (\lambda x'. + x' y)) 4 3 \stackrel{(\beta)}{=} (\lambda x'. + x' y)[4/y] 3 = \beta\text{-Konversion}$$

$$(\lambda x'. + x' 4) 3 \stackrel{(\beta)}{=} (+ x' 4)[3/x'] = \beta\text{-Konversion}$$

$$+ 3 4 =$$

7

c)

$$(\lambda x. (\lambda x'. + (- x 1)) x 3) 9 \stackrel{(\alpha)}{=} (\lambda x. (\lambda x'. + (- x 1)) [x'/x] x 3) 9 = \alpha\text{-Konversion}$$

$$(\lambda x. (\lambda x'. + (- x' 1)) x 3) 9 \stackrel{(\beta)}{=} ((\lambda x'. + (- x' 1)) x 3) [9/x] = \beta\text{-Konversion}$$

$$(\lambda x'. + (- x' 1)) 9 3 \stackrel{(\beta)}{=} + (- x' 1) [9/x'] 3 = \beta\text{-Konversion}$$

$$+ (-9 1) 3 =$$

$$+ 8 3 =$$

11

d)

$$(\lambda x. (\lambda y. (\lambda x. + x) y) x) 4 \stackrel{(\alpha)}{=} (\lambda x. ((\lambda y. (\lambda x'. + x)[x'/x] y) x)) 4 = \alpha\text{-Konversion}$$

$$(\lambda x. (\lambda y. (\lambda x'. + x') y) x) 4 \stackrel{(\beta)}{=} ((\lambda y. (\lambda x'. + x') y) x)[4/x] = \beta\text{-Konversion}$$

$$(\lambda y. (\lambda x'. + x') y) 4 \stackrel{(\beta)}{=} ((\lambda x'. + x') y) [4/y] = \beta\text{-Konversion}$$

$$(\lambda x'. + x') 4 \stackrel{(\beta)}{=} + x'[4/x'] = \beta\text{-Konversion}$$

+ 4 =>

4

---

### 3. Klammergebirge und Klammertiefe

#### Lösung Teilaufgabe 1:

Die Sprache  $S_1$  besteht aus einer beliebigen Anzahl von Klammerfolgen der Tiefe 1, also nur Aneinanderreihungen des Wortes  $,()$ . Die leere Zeichenfolge (Tiefe 0) darf laut Voraussetzung nicht in  $S_1$  vorkommen.

Beispielwörter:  $()$ ,  $()()$ ,  $()()()$

Die Sprache  $S_2$  besteht aus einer beliebigen Anzahl von Klammerfolgen der Tiefe 1 oder 2. Es sind also alle Wörter aus  $S_1$  in  $S_2$  enthalten, jedoch zusätzlich noch Wörter mit der Klammertiefe 2.

Beispielwörter:  $()$ ,  $()()$ ,  $(())$ ,  $(())()$ ,  $((())())$

Die Sprache  $S_3$  besteht aus einer beliebigen Anzahl von Klammerfolgen der Tiefe 1, 2 oder 3. Es sind also alle Wörter aus  $S_1$  und  $S_2$  in  $S_3$  enthalten, jedoch zusätzlich noch Wörter mit der Klammertiefe 3.

Beispielwörter:  $()$ ,  $()()$ ,  $(())$ ,  $(())()$ ,  $((())())$ ,  $((()))$ ,  $((()))()$ ,  $((()))((()))$

#### Lösung Teilaufgabe 2:

$$T_i = \{ [()^i]^m, m > 0 \}$$

$$G_i = (\{()\} \{S\}, P_i, S)$$

$$P_i = \{ S \rightarrow ()^i S \mid ()^i \}$$

---

### 4. Klammergebirge als regulärer Ausdruck und in BNF-Notation

#### Teilaufgabe 1:

$R_2$ : Aus jedem beliebigen Wort  $w$  aus  $S_1$  lässt sich ein Wort  $u$  mit Klammertiefe 2 aus  $S_2$  erzeugen, indem man  $w$  in Klammern setzt:  $u = (w)$ .

$$R_2 = \{ ( \{R_1\}^* ) \}^+$$

(Es muss mindestens das Wort  $"()$ " durch  $R_2$  beschrieben werden, daher die äußere Klammer  $\{ \}^+$ .)

$R_3$ : Aus jedem beliebigen Wort  $w$  aus  $S_2$  lässt sich ein Wort  $u$  mit Klammertiefe 3 aus  $S_3$  erzeugen, indem man  $w$  in Klammern setzt:  $u = (w)$ .

$$R_3 = \{ ( \{R_2\}^* ) \}^+$$

(Es muss mindestens das Wort  $"()$ " durch  $R_3$  beschrieben werden, daher die äußere Klammer  $\{ \}^+$ .)

#### Teilaufgabe 2:

Induktive Definition für  $S_i, i \in \mathbb{N}$

$$S_1 = R_1 = \{ () \}^+$$

$$S_{i+1} = R_{i+1} = \{ ( \{R_i\}^* ) \}^+$$

Der reguläre Ausdruck  $R_i$  beschreibt nach Induktionsvoraussetzung alle Wörter mit einer Klammertiefe  $\leq i$ .

Setzt man um diesen Ausdruck nun Klammern, so werden Wörter der Klammertiefe  $\leq i+1$  erzeugt. Klammertiefe 1 wird durch die Klammerung  $\{ \}^*$  von  $R_i$  möglich, da  $R_i$  dann auch weggelassen werden kann und gerade die Wörter aus  $S_1$  erzeugt werden.

Die Sprache  $S$  kann nicht als regulärer Ausdruck definiert werden, da sie eine Typ 2 Sprache ist. Sie ist Obermenge von der Sprache  $(\ )^n$ , von der bereits gezeigt (vorhergehende Übungsblätter) wurde, dass sie eine Typ 2 Sprache ist.

Reguläre Ausdrücke können jedoch nur genau die Menge der CH-3 Sprachen beschreiben. Dies liegt daran, dass mit Operatoren wie  $*$  oder  $+$  die Anzahl von öffnenden und schließenden Klammern in einem Ausdruck nicht vergleichbar werden kann ( $(\ )^n$ , wohl aber mit einem Ausdruck wie  $(\ )^n$ , was jedoch kein regulärer Ausdruck ist).

Die Sprachen  $S_i$  sind CH-3 Sprachen, da hier die Anzahl der öffnenden und schließenden Klammern endlich ist. Solche Sprachen können mit CH-3 Grammatiken (mit gewissem Aufwand) realisiert werden, indem man für jede Klammerebene ein neues Nichtterminalzeichen einführt.

### Teilaufgabe 3:

Die Backus-Naur-Form lässt sich fast direkt aus den Ausdrücken für  $R_i$  und  $R_{i+1}$  ableiten. Folgende Aussage muss in BNF umgesetzt werden:

Entweder es wird ein Wort aus  $S_1$  erzeugt, oder ein bestehendes Wort aus der Sprache  $S_1$  wird zu einem Wort aus der Sprache  $S_{i+1}$ , indem man Klammern darum setzt.

Somit ergibt sich folgender Ausdruck:

$\langle S \rangle ::= \{ (\ ) \}^+ \mid \{ (S) \}^+$