

# Semantik und Pragmatik: Übungen

Universität Bielefeld

10. Juli 2005

**Aufgabe 1** Gibt es unter den folgenden Sätzen Tautologien, Kontradiktionen und konsistente Sätze? Welche sind das jeweils?

- a) Wenn die Koalition die Wahl gewinnt, können Schweine fliegen.  
konsistent
- b) Schweine können fliegen.  
konsistent
- c) Alle Menschen sind sterblich, aber Sokrates ist unsterblich.  
konsistent
- d) Alle Menschen sind sterblich, aber Sokrates ist ein Mensch und trotzdem unsterblich.  
Kontradiktion
- e) Wenn alles ein Ende hat, dann hat auch das Semester ein Ende.  
Tautologie

**Aufgabe 2** In welcher Sinnerelation stehen die Sätze (b) und (c) jeweils zu Satz (a)?

- a) Alle Menschen sind sterblich.
- b) Wenn Sokrates unsterblich ist, dann ist er kein Mensch.  
folgt aus (a)
- c) Sokrates ist unsterblich.  
keine Sinnrelation zu (a)

**Aufgabe 3** Übersetzen Sie die folgenden Sätze intuitiv (also nicht mit Hilfe einer formalen Grammatik) in die Typentheorie. Wenn ein Satz ambig ist, geben Sie alle Lesarten an.

- a) Kein Mensch ist unsterblich.  
 $\neg \exists x(\text{MENSCH}'(x) \wedge \neg \text{STERBLICH}'(x))$
- b) Ein Student kaufte „Anna Karenina“.  
 $\exists x(\text{STUDENT}'(x) \wedge \text{KAUFT}'(\text{ANNA\_KARENINA}')(x))$
- c) Kein Student kaufte kein Buch.  
 $\neg \exists x(\text{STUDENT}'(x) \wedge \neg \exists y(\text{BUCH}'(y) \wedge \text{KAUFT}'(y)(x)))$
- d) Kein Buch kaufte jeder Student.  
 $\neg \exists x(\text{BUCH}'(x) \wedge \forall y(\text{STUDENT}'(y) \rightarrow \text{KAUFT}'(x)(y)))$
- e) Nicht geschlafen hat kein Kind.  
 $\neg \exists x(\text{KIND}'(x) \wedge \neg \text{SCHLIEF}'(x))$

**Aufgabe 4** Gegeben seien die folgenden typentheoretischen Konstanten:

- $R : \langle e, \langle e, t \rangle \rangle$
- $Q : \langle e, t \rangle$
- $a : e$
- $p : t$
- $f : \langle \langle e, t \rangle, \langle e, t \rangle \rangle$

Entscheiden Sie, ob die folgenden Ausdrücke dann wohlgeformt sind. Wenn ja, geben Sie den Typ an.

- a)  $R(p)$  nicht wohlgeformt
- b)  $f(Q) : \langle e, t \rangle$
- c)  $f(R)(a)$  nicht wohlgeformt
- d)  $f(R(a)) : \langle e, t \rangle$
- e)  $Q(a) : t$

**Aufgabe 5** Gegeben sei die folgenden Grammatik (wobei die erste Zeile einer Regel immer die Syntax-Regel angibt und die zweite Zeile die korrespondierende Übersetzungsregel):

1.  $S \rightarrow NP, VP$   
 $S \Rightarrow NP'(VP')$
2.  $NP \rightarrow N_{pr}$   
 $NP \Rightarrow \lambda P(P(N'_{pr}))$
3.  $NP \rightarrow D, N$   
 $NP \Rightarrow D'(N')$
4.  $VP \rightarrow V_i$   
 $VP \Rightarrow V'_i$
5.  $VP \rightarrow V_t, NP$   
 $VP \Rightarrow \lambda x(NP'(\lambda y(V'_t(y)(x))))$
6.  $VP \rightarrow \text{was}, VP[PAS]$   
 $VP \Rightarrow VP[PAS]'$
7.  $VP[PAS] \rightarrow V_t[PAS]$   
 $VP[PAS] \Rightarrow \lambda y \exists z(V_t[PAS]'(y)(z))$
8.  $VP[PAS] \rightarrow V_t[PAS], PP_{by}$   
 $VP[PAS] \Rightarrow \lambda z(V_t[PAS]'(z)(PP'_{by}))$
9.  $PP_{by} \rightarrow \text{by}, N_{pr}$   
 $PP_{by} \Rightarrow N'_{pr}$
10.  $N_{pr} \rightarrow \text{John}$   
 $N_{pr} \Rightarrow \text{JOHN}'$
11.  $N \rightarrow \text{cake}$   
 $N \Rightarrow \text{CAKE}'$
12.  $D \rightarrow \text{some}$   
 $D \Rightarrow \lambda P \lambda Q \exists x(P(x) \wedge Q(x))$
13.  $D \rightarrow \text{every}$   
 $D \Rightarrow \lambda P \lambda Q \forall w(P(w) \rightarrow Q(w))$
14.  $N \rightarrow \text{student}$   
 $N \Rightarrow \text{STUDENT}'$

15.  $N \rightarrow \text{book}$   
 $N \Rightarrow \text{BOOK}'$
16.  $V_i \rightarrow \text{walked}$   
 $V_i \Rightarrow \text{WALK}'$
17.  $V_t \rightarrow \text{read}$   
 $V_t \Rightarrow \text{READ}'$
18.  $V_t[\text{PAS}] \rightarrow \text{read}$   
 $V_t[\text{PAS}] \Rightarrow \text{READ}'$
19.  $V_t \rightarrow \text{ate}$   
 $V_t \Rightarrow \text{EAT}'$
20.  $V_t[\text{PAS}] \rightarrow \text{eaten}$   
 $V_t[\text{PAS}] \Rightarrow \text{EAT}'$

Übersetzen Sie mit Hilfe dieser Regeln die folgenden Sätze in die Typentheorie:

a) Some cake was eaten.

$$\begin{array}{c} S \Rightarrow \\ \exists x(\text{CAKE}'(x) \wedge \exists z(\text{EAT}'(x)(z))) \end{array}$$

$$\begin{array}{cc} \text{NP} \Rightarrow & \text{VP}[\text{PAS}] \Rightarrow \\ \lambda Q \exists x(\text{CAKE}'(x) \wedge Q(x)) & \lambda y \exists z \text{EAT}'(y)(z) \end{array}$$

$$\begin{array}{ccc} \text{D} \Rightarrow & \text{N} \Rightarrow & \text{was} \quad V_t[\text{PAS}] \Rightarrow \\ \lambda P \lambda Q \exists x(P(x) \wedge Q(x)) & \text{CAKE}' & \text{EAT}' \\ | & | & | \\ \text{some} & \text{cake} & \text{eaten} \end{array}$$

b) Some cake was eaten by John.

$$S \Rightarrow \exists x(\text{CAKE}'(x) \wedge \text{EAT}'(x)(\text{JOHN}'))$$

$$\text{NP} \Rightarrow \lambda Q \exists x(\text{CAKE}'(x) \wedge Q(x)) \quad \text{VP} \Rightarrow \lambda z \text{EAT}'(z)(\text{JOHN}')$$

$$\begin{array}{ccccc}
 \text{D} \Rightarrow & \text{N} \Rightarrow & \text{was} & \text{VP[PAS]} \Rightarrow & \\
 \lambda P \lambda Q \exists x(P(x) \wedge Q(x)) & \text{CAKE}' & & \lambda z \text{EAT}'(z)(\text{JOHN}') & \\
 | & | & & & \\
 \text{some} & \text{cake} & & & \\
 & & & \text{V}_t[\text{PAS}] \Rightarrow & \text{PP} \Rightarrow \\
 & & & \text{EAT}' & \text{JOHN}' \\
 & & & | & \\
 & & & \text{eaten} & \text{by } \text{N}_{pr} \Rightarrow \\
 & & & & \text{JOHN}' \\
 & & & & | \\
 & & & & \text{John}
 \end{array}$$

c) Every book was read by John.

$$S \Rightarrow \forall w(\text{BOOK}'(w) \rightarrow \text{READ}'(w)(\text{JOHN}'))$$

$$\text{NP} \Rightarrow \lambda Q \forall w(\text{BOOK}'(w) \rightarrow Q(w)) \quad \text{VP} \Rightarrow \lambda z \text{READ}'(z)(\text{JOHN}')$$

$$\begin{array}{ccccc} D \Rightarrow & N \Rightarrow & \text{was} & \text{VP}[\text{PAS}] \Rightarrow & \\ \lambda P \lambda Q \forall w(P(w) \wedge Q(w)) & \text{BOOK}' & & \lambda z \text{READ}'(z)(\text{JOHN}') & \\ | & | & & & \\ \text{every} & \text{book} & & & \\ & & & V_t[\text{PAS}] \Rightarrow & \text{PP} \Rightarrow \\ & & & \text{READ}' & \text{JOHN}' \\ & & & | & | \\ & & & \text{read} & \text{by } N_{pr} \Rightarrow \\ & & & & \text{JOHN}' \\ & & & & | \\ & & & & \text{John} \end{array}$$

d) John read every book.

$$S \Rightarrow \forall w(\text{BOOK}'(w) \rightarrow \text{READ}'(w)(\text{JOHN}'))$$

$$\begin{array}{cc} \text{NP} & \text{VP} \\ | & \\ N_{pr} \Rightarrow & \\ \text{JOHN}' & \\ | & \\ \text{John} & V_t \Rightarrow \\ & \text{READ}' \\ & | \\ & \text{read} \\ & \\ & \text{NP} \Rightarrow \\ & \lambda Q \forall w(\text{BOOK}'(w) \rightarrow Q(w)) \\ & \\ & D \Rightarrow & N \Rightarrow \\ & \lambda P \lambda Q \forall w(P(w) \wedge Q(w)) & \text{BOOK}' \\ & | & | \\ & \text{every} & \text{book} \end{array}$$

