Seminar 20. Mai

0.1 Syntax/Semantik

Syntax, Form, Zeichenketten, Umformen, Ableiten, Axiome, Schlußregeln, $\vdash$
Semantik, Bedeutung, Modelle, Wahrheit, Folgern, $\models$
Zweck des Vollständigkeitssatzes: bei passendem Regelsystem (Hilbert, Modus Ponens) ist $\vdash$ das gleiche wie $\models$.

0.2 Verschiedene Regelsysteme

System	Grundbereich
Hilbert und Modus Ponens	Formeln
Aufg. 52: $uxw \vdash uyw$	Wörter
`make` (Unix)	Files
Curry-Howard	Typen

0.3 Curry-Howard-Isomorphie

$a, b, \ldots$ sind Typvariablen.
$x :: a$ heißt: x hat Typ a. Beispiele: `65 :: Int`, `'A' :: Char`.
$a \to b$ ist der Typ "Funktionen von a nach b". Beispiel: `ord :: Char -> Int`.
Schlußregel für Typ-Aussagen ist Modus ponens: $\dfrac{x :: a, f :: a \to b}{f(x) :: b}$
Das wird sehr interessant bei Funktionen höherer Ordnung
(Funktionales Programmieren, Hauptstudium)

0.4 Üben Hilbertscher Axiome

Leite $x \to x$ ab. Benutze 14, 15, 3.

```
http://www.informatik.uni-leipzig.de/~joe/edu/ss99/logik/hilbert/
F1 := sub H14 [A := x]  =  x -> not (not x)
F2 := sub H15 [A := x]  =  not (not x) -> x
F3 := sub H3 [A := x, B := not (not x), C := x]
  =  (x -> not (not x)) -> ((not (not x) -> x) -> (x -> x))
F4 := mopo F1 F3  =  (not (not x) -> x) -> (x -> x)
claim := mopo F2 F4  =  x -> x
```

0.5 Deduktionstheorem

für Folgern: $(X \cup \{A\} \models B) \;\Rightarrow\; (X \models A \to B)$.
Beweis (straightforward): zu zeigen ist: unter der gegebenen Voraussetzung: für alle Belegungen m: wenn $m \models X$, dann $m \models A \to B$. Fallunterscheidung $m \models A$ oder $m \not\models A$.
für Ableiten: $(X \cup \{A\} \vdash B) \;\Rightarrow\; (X \vdash A \to B)$.

> **Wichtig** Wiederholung $\vdash, \vdash_a$. Hier brauchen wir den Unterschied!
>
> $X \vdash_a B$ heißt: B ist mit mod. pon. aus X ableitbar.
>
> $X \vdash B$ heißt $X \cup H \vdash_a B$, dabei $H =$ Hilbert-Axiome. D. h. B ist aus X und Axiomen ableitbar.

Beweis (jetzt völlig anderer Beweisgang als bei $\models$): durch Induktion nach Ableitungslänge.
Zu zeigen ist: für alle $n \geq 0$: $(X \cup \{A\} \vdash^n B) \;\Rightarrow\; (X \vdash A \to B)$.
Induktionsanfang: $n = 0$, also $X \cup \{A\} \vdash^n B$. Fallunterscheidung

1. $B = A$. Benutze $H \vdash A \to A$. Siehe obige Bemerkung. Wenn wir H nicht hätten (z. b. $X =$), dann nicht $X \vdash A \to A$.

2. $B \in X \cup H$. Benutze $H \vdash B \to (A \to B)$.

Induktionsschritt: Behauptung für n gezeigt. Betrachte $X \cup \{A\} \vdash^{n+1} B$. Der $n+1$te Schritt war ein Modus Ponens. Also gibt es ein C mit

$$X \cup \{A\} \vdash^{\leq n} C, X \cup \{A\} \vdash^{\leq n} C \to B$$

Nach Induktionsannahme folgt $X \vdash A \to C, X \vdash A \to (C \to B)$
Wir müssen zeigen $H \vdash (A \to C) \to ((A \to (C \to B)) \to (A \to B))$.
Das ist mühevoll, aber möglich.

0.6 Bemerkung zum (Automatischen) Ableiten

Ableitungen lassen sich automatisch prüfen (siehe mein Applet). Es ist viel schwerer, sie automatisch zu finden. (siehe Logische Programmierung, Automatisches Beweisen, Hauptstudium).
Für Aussagenlogik braucht man das aber "eigentlich" nicht: durch Wahrheitstabellen hat man ein einfaches Entscheidungsverfahren.
Aber bei Prädikatenlogik ist damit Essig, weil die Modellmengen dann viel komplizierter sind. Deswegen ist es schon richtig, das Ableiten jetzt schon zu üben.