

**Lehrstuhl für Programmiersprachen
und Übersetzerkonstruktion**

Institut für Informatik

Christian-Albrechts-Universität zu Kiel

Prof. Dr. M. Hanus, S. Fischer



12. Übung zur Vorlesung „Logikprogrammierung“
Sommersemester 2006

Diese Serie enthält Aufgaben zur Klausurvorbereitung und muss nicht abgegeben werden!

Die Klausur findet am 19.7.2006 um 10:15 - 11:45 Uhr statt.

Aufgabe 33

keine Punkte

(a) Gegeben sei das logische Programm:

$$\begin{aligned} & \Rightarrow p(a,b) \\ q(a,b) & \Rightarrow p(b,a) \\ p(X,Y) & \Rightarrow q(Y,X) \end{aligned}$$

und die Auswahlregel „wähle das letzte Literal“. Geben Sie die SLD-Ableitungen einschließlich Resolventen und Unifikatoren eines jeden einzelnen Schritts für die folgenden Anfragen an:

- (1) $p(a,b), q(b,a) \Rightarrow$
- (2) $q(a,b) \Rightarrow$
- (3) $q(b,b) \Rightarrow$

Welche der Ableitungen ist eine SLD-Widerlegung und welche Bedeutung hat diese?

(b) Gegeben sei das logische Programm:

$$\begin{aligned} & \Rightarrow p(f(X), X) \\ & \Rightarrow q(f(Y), f(Y)) \end{aligned}$$

und die Auswahlregel FIRST. Die Anfrage laute $p(Z,Y), q(Y,Y) \Rightarrow$

- (1) Geben Sie eine SLD-Widerlegung an.
- (2) Zeigen Sie, dass falls man bei den SLD-Schritten Klauselvariablen nicht umbenennt, d.h. keine Variantenbildung durchführt, für alle allgemeinsten Unifikatoren keine SLD-Widerlegung existiert.

(c) Gegeben sei das folgende logische Programm P:

$$\begin{aligned} q(Y), r(Y) & \Rightarrow p(X) \\ q(Y) & \Rightarrow q(h(Y)) \\ & \Rightarrow r(g(Y)) \end{aligned}$$

Bestimmen Sie zur Anfrage $p(a) \Rightarrow$ zwei Auswahlregeln, von denen die eine zu einem unendlichen SLD-Baum und die andere zu einem endlichen fehlgeschlagenen SLD-Baum führt. Wie verhält sich das Ergebnis zu der Aussage, dass $P \cup \{p(a) \Rightarrow\}$ genau dann erfüllbar ist, wenn es eine Auswahlregel gibt, für die eine SLD-Widerlegung existiert?

Aufgabe 34

keine Punkte

Überprüfen Sie mit Hilfe des Algorithmus zum Auffinden eines allgemeinsten Unifikators die folgenden Termpaare auf Unifizierbarkeit. Geben Sie, falls die Terme unifiziert werden können, einen allgemeinsten Unifikator an. Weshalb ist andernfalls die Unifikation fehlgeschlagen?

- (a) $z(d(e(F)), a(b(D)), g(H))$ und $z(H, K, g(F))$
- (b) $h(e(0), n, k, f(E), g(L))$ und $h(T, A, N, f(t), E)$
- (c) $f(D, g(A, B))$ und $f(g(B, C), g(u(v(t), B), h(v(C))))$

Aufgabe 35 (Programmieraufgabe)

keine Punkte

Schreiben Sie ein Prädikat `fib`, welches eine Zahl n einliest, in linearer Zeit $fibonacci(n)$ berechnet und diesen Wert ausgibt.

Aufgabe 46 (Programmieraufgabe)

keine Punkte

Zwei Süßwarenläden versuchen nach Ostern die übriggebliebenen Ostereier durch Sonderangebote noch zu verkaufen. Dies nutzen Britta und Peter aus, Britta in dem einen Laden und Peter im anderen.

Am Montag kauft Britta nur Marzipaneier, während Peter sich auf Liköreier beschränkt. Dienstags kaufen beide nur Nougateier, und am Mittwoch kauft Britta nur Liköreier, Peter hingegen nur Marzipaneier. Britta hat an jedem Tag jeweils ein Ei weniger als am Vortag gekauft, während Peter jeden Tag ein Ei mehr gekauft hat als zuvor.

Alle Eier haben einen vollen Cent-Betrag gekostet. In beiden Läden kostete ein Nougatei 20 Cent mehr als ein Marzipanei und ein Likörei 20 Cent mehr als ein Nougatei. Beide haben 28,67 Euro für Ihre Eier bezahlt, was jedoch nicht bedeutet, dass die Preise für die Eiersorten in beiden Läden gleich waren. Die einzelnen Preisunterschiede waren jedoch geringfügig.

Peter hat insgesamt mehr Eier gekauft als Britta. Wie viele mehr?

Aufgabe 47 (Programmieraufgabe)

keine Punkte

Definieren Sie in Prolog ein Prädikat `label(T,LT)`, das in einem beliebigen Term allen Funktoren ein zusätzliches Argument hinzufügt, welches seine Position im Term beschreibt. Zum Beispiel soll durch den Aufruf

```
label(f(a,g(b),h(a,b)),LT).
```

die Variable `LT` an den Wert

```
f([],a([0]),g([1],b([1,0])),h([2],a([2,0]),b([2,1])))
```

gebunden werden.