

9. Übung zur Vorlesung „Prinzipien von Programmiersprachen“  
Wintersemester 2008/2009

---

Abgabe: 27. Januar 2009 in der Vorlesung

**Aufgabe 31**

(Präsenzaufgabe)

Überprüfen Sie mit dem Algorithmus zur Berechnung des allgemeinsten Unifikators (*mgu*) die folgenden Termpaare auf Unifizierbarkeit:

- a)  $f(g(a, X), h(Y, c(Z)), h(X, V))$  und  $f(Y, h(g(Z, c(V)), c(a)), h(c(V), c(c(Z))))$
- b)  $h(f(V, Y), h(Z, f(a, X)))$  und  $h(f(h(X, g(Y)), a), h(Z, g(a)))$
- c)  $f(A, g(C, B))$  und  $f(g(B, D), g(v(t, B), h(D)))$
- d)  $f(A, B, C)$  und  $f(g(X, X), g(A, A), g(B, B))$

Geben Sie die allgemeinsten Unifikatoren an bzw. überlegen Sie sich, warum die Unifikation fehlgeschlagen ist.

**Aufgabe 32**

(Präsenzaufgabe)

Gegeben sei das Prolog-Programm:

```
p(a, b).  
p(b, a) :- q(a, b).  
q(Y, X) :- p(X, Y).
```

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:

- (a)  $?- p(a, b), q(b, a).$       (b)  $?- q(a, b).$       (c)  $?- q(b, b).$

Welche der Ableitungen ist eine erfolgreiche SLD-Ableitung?

**Aufgabe 33**

Gegeben sei das folgende Prolog-Programm:

```
last([X|Xs], Y) :- last(Xs, Y).  
last([X], X).
```

Berechnen Sie die Antworten auf folgende Anfragen mit dem in der Vorlesung vorgestellten Inferenzsystem zur Formalisierung von Prolog:

- (a)  $?- last([1, 2], E).$       (b)  $?- last(L, 5).$

### Aufgabe 34

Mit den Inferenzregeln aus der Vorlesung zur Formalisierung von Prolog ist keine Aussage ableitbar, falls zu der Anfrage unendlich viele Antworten existieren oder wenn eine unendliche Berechnung existiert. Erweitern Sie das Inferenzsystem so, dass höchstens die ersten  $n$  Antworten berechnet werden.

### Aufgabe 35

Definieren Sie folgende Listenoperationen durch Prolog-Klauseln:

- a) Anhängen eines Elements an eine Liste  
(Prädikat `anhang(Liste,Element,ListeUndElement)`)
- b) Umkehrung der Reihenfolge aller Elemente einer Liste.  
(Prädikat `reverse(List,RevList)`)
- c) Überprüfung, ob eine Liste Teilliste einer anderen Liste ist. Dabei ist eine Teilliste von  $[E_1, \dots, E_n]$  ( $n \geq 0$ ) entweder die leere Liste oder eine Liste der Form  $[E_i, \dots, E_j]$  ( $1 \leq i \leq j \leq n$ ). (Prädikat `subList(SubList,List)`)

### Aufgabe 36

Im zweiten Kapitel wurde die natürliche Semantik für arithmetische Terme als Inferenzsystem eingeführt. Implementieren Sie in Prolog ein Prädikat `natsem(s,e,v)`, das erfüllt ist, falls  $s$  die Kodierung eines Zustandes,  $e$  die Kodierung eines arithmetischen Terms und  $v$  das Ergebnis der Auswertung von  $e$  im Zustand  $s$  ist. Die folgende Grammatik beschreibt den Aufbau eines arithmetischen Terms:

$$\begin{array}{ll} \text{Exp} := & n \quad (n \text{ Zahl}) \\ & | \ x \quad (x \text{ Variable}) \\ & | \ \text{Exp } op \ \text{Exp} \quad (\text{mit } op \in \{+, *\}) \\ & | \ \text{let } x = \ \text{Exp in Exp} \quad (x \text{ Variable}) \end{array}$$

Kodieren Sie einen arithmetischen Term durch einen Prolog-Term der Form `zahl(n)`, `var(x)`, `add(e1,e2)`, `mult(e1,e2)` oder `let(x,e1,e2)`. Hierbei ist  $n$  eine Zahl,  $x$  ein Variablenname (Konstante in Prolog) und  $e1$  und  $e2$  sind wieder arithmetische Terme. Ein Zustand wird repräsentiert von einer Liste von Paaren `bind(x,v)`. Die erste Komponente  $x$  ist die Variable und  $v$  ist der Wert, an den  $x$  gebunden ist.

**Hinweis:** Verwenden Sie das vordefinierte Prädikat `Z is X`. Beim Beweis dieses Prädikats wird der arithmetische Ausdruck  $X$  ausgewertet und das Resultat wird mit  $Z$  unifiziert. Zum Beispiel liefert die Anfrage `?- X is 3+4.` die Antwort `X = 7.`

### Hinweis zur Benutzung von Prolog:

Das auf den Instituts-Rechnern installierte Sicstus-Prolog-System lässt sich mit

```
/home/prolog/bin/sicstus
```

starten. Prolog-Programme sollten die Endung `.pl` haben. Das Programm `file.pl` lässt sich mit

```
| ?- [file].
```

in das Prolog-System laden. Weitere Informationen, auch zu frei verfügbaren Prolog-Systemen, finden Sie auf der Website zur Vorlesung.