

**Proseminar Logische Programmierung
Elmar Eder, Universität Salzburg**

Blatt 1, Aufgaben für 8.3.2021

Aufgabe 1 In einer Familie haben Bert und Anna die Kinder Hans und Inge. Lisa hat die Eltern Dirk und Dora. Kurt ist Sohn von Hans und Lisa.

- (a) Stellen Sie die Verwandtschaftsbeziehungen graphisch dar!
- (b) Schreiben Sie ein Prolog-Programm zur Beschreibung der Verwandtschaftsbeziehungen und des Geschlechts von Personen, in dem Sie Prädikate `kind/2`, `maennlich/1` und `weiblich/1` definieren!
- (c) Fragen Sie Prolog, wessen Kind Inge ist, wer Kind von Anna ist, wer Kind von wem ist, wer Kind von sich selbst ist, wer Vater und Mutter von von wem sind, und wer Großmutter väterlicherseits von wem ist! Lassen Sie sich jeweils durch wiederholte Eingabe eines Strichpunkts alle Antworten ausgeben!

Aufgabe 2 Drücken Sie die folgenden Aussagen über verschiedene Tiergruppen und ihre bevorzugten Fortbewegungsmethoden in Form eines Prologprogramms aus:

- (a) Löwe, Rind, Wal und Fledermaus sind Säugetiere.
- (b) Star, Strauß und Wasseramsel sind Vögel.
- (c) Löwe, Rind und Strauß gehen gerne.
- (d) Forelle, Wasseramsel und Wal schwimmen gerne.
- (e) Fledermaus, Star und Wasseramsel fliegen gerne.

Fragen Sie Prolog

- (f) nach einem Säugetier, das gerne schwimmt
- (g) nach einem Vogel, der gerne fliegt
- (h) nach einem Vogel, der gerne fliegt und gerne schwimmt

Schreiben Sie diese Anfragen als Kommentare in das Programm hinein unter das eigentliche Programm! Die letzten drei Zeilen sollten also jeweils mit `% ?-` beginnen und mit einem Punkt enden. Achten Sie darauf, weder Umlaute noch scharfes s zu verwenden, stattdessen `ae`, `oe`, `ue`, `ss`, `Ae`, `Oe`, `Ue`!

Aufgabe 3 Gegeben ist das Prolog-Programm

```
zahl(7).  
zahl(2).  
zahl(5).  
zahl(3).
```

Stellen Sie an Prolog eine Anfrage, die Prolog veranlasst, die Antwort

A = 2
B = 3
C = 5
D = 7

zu geben, also die Variablen A, B, C und D zu instantiieren zu den oben gegebenen Zahlen, sortiert in aufsteigender Reihenfolge! Verwenden Sie bei dieser Aufgabe keine eingebauten Prädikate außer < und keine Datentypen außer Zahlen (also auch keine Listen)!

Aufgabe 4 In Prolog ist jeder Term entweder ein atomarer Term (englisch: atomic), eine Variable (variable) oder ein zusammengesetzter Term, auch Struktur genannt (compound term, structure). Ein atomarer Term ist entweder ein Prolog-Atom (atom) oder eine Zahl (number). Eine Zahl ist entweder eine ganze Zahl (integer) oder eine Fließkommazahl (floating point number). Zum Testen des Typs eines Terms gibt es die eingebauten Prädikate `atomic/1`, `var/1`, `nonvar/1`, `compound/1`, `atom/1`, `number/1`, `integer/1` und `float/1`. Wenn der gegebene Ausdruck kein Term ist, liefern diese Prädikate eine Fehlermeldung. Beispiel: Die Anfrage “?- `compound(f(a,b))`.” liefert die Antwort `yes`. Ein zusammengesetzter Term hat die Form $f(a_1, \dots, a_n)$, wobei f der *Funktor* genannt wird, n die *Stelligkeit* und a_1, \dots, a_n die *Argumente* des Terms. Um den Funktor, die Stelligkeit und die Argumente eines Terms zu extrahieren, gibt es die eingebauten Prädikate `functor/3` und `arg/3`. Beispiel: Die Anfrage “?- `functor(f(a,b),F,N)`.” liefert den Funktor `f` und die Stelligkeit `2`. Die Anfrage “?- `arg(2,f(a,b),A)`.” liefert das 2-te Argument `b` des Terms `f(a,b)`. Probieren Sie es aus!

Rufen Sie in Prolog jedes der Typ-Test-Prädikate mit jedem der folgenden Ausdrücke auf! Schauen Sie sich Prologs Antwort an und geben Sie dann von jedem dieser Ausdrücke an, ob es sich um einen Term handelt und von welchem/n Typ(en) er ist! Erklären Sie, soweit Sie es können, aus welchem Grund er von diesem Typ ist! Rufen Sie die Prädikate `functor/3` und `arg/3` mit jedem der Terme auf und geben Sie dann von jedem der Terme an, was der Funktor, die Stelligkeit und die Argumente sind! Die Prädikate `functor/3` und `arg/3` funktionieren nicht nur mit zusammengesetzten Termen, sondern auch mit manchen anderen. Mit welchen? Hier sind die Ausdrücke, die Sie testen sollen:

```
f(a,g(b,c))   h(X,2)   32   -5.3e-12   sin(x)   6-2   20-3*5   20-3-9   Meyer
Hans Meyer   hans_meyer   hans-meyer   Hans_Meyer   'Hans Meyer'   [a,b,c]
```