

Übungsserie 5a

Programmierung mit Prolog

Verwenden Sie für die Übungsaufgaben den SWI-Prolog-Interpreter. Der Interpreter kann von der Website <http://www.swi-prolog.org> für verschiedene Betriebssysteme heruntergeladen werden, ebenso ist an dieser Stelle eine Dokumentation verfügbar. In Prolog lädt man eine Datei durch `consult('filename')`. und mit `halt.` verlässt man den Interpreter. Achten Sie auf den abschließenden Punkt! Weitere Hilfestellung erhalten Sie, wenn Sie `help.` eingeben.

Bitte beachten Sie den Abgabezeitpunkt in Moodle. Serien per E-Mail oder verspätete Abgaben können nicht gewertet werden. Die Textaufgaben bitte per PDF mit den Programmieraufgaben zusammen in ein Zip-Archiv bündeln.

Aufgabe 1 (10 Punkte) Gegeben ist die folgende Datenbank:

```
1 male(tom).
2 male(bernd).
3 male(klaus).
4 male(simon).
5 male(david).
6 male(hugo).
7 female(anna).
8 female(susi).
9 female(karin).
10 female(klara).
11 /* likes(X, Y) - X mag Y */
12 likes(tom, dogs).
13 likes(anna, tom).
14 likes(hugo, wine).
15 likes(susi, dogs).
16 likes(simon, dogs).
17 likes(simon, anna).
18 /* parent(X, Y) - X ist Elternteil von Y
    */
19 parent(anna, bernd).
20 parent(klaus, tom).
21 parent(klaus, hugo).
22 parent(klaus, susi).
23 parent(simon, david).
24 parent(simon, klara).
25 parent(karin, david).
26 parent(karin, hugo).
27 parent(karin, susi).
28 parent(karin, klara).
```

Modellieren Sie folgende Regeln in Prolog und speichern Sie diese zusammen mit der Datenbank in einer Datei `family.pl`.

- `mother(X, Y)`: X ist die Mutter von Y
- `father(X, Y)`: X ist der Vater von Y
- `parents(X, Y)`: X und Y haben ein Kind
- `brother(X, Y)`: X ist der Bruder von Y
- `sister(X, Y)`: X ist die Schwester von Y
- `halfSibling(X, Y)`: X ist Halbgeschwister von Y
- `son(X, Y)`: X ist der Sohn von Y
- `daughter(X, Y)`: X ist die Tochter von Y
- `likes(X, Y)`: Jeder mag seine Kinder und Eltern
- `likes(klaus, Y)`: Klaus mag jeden, der Hunde mag

Aufgabe 2 (5 Punkte) Schreiben Sie ein Prologprogramm `minlist.pl` welches ein minimales Element einer Liste von Ganzzahlen ermittelt und ausgibt. Verwenden Sie dabei *nicht* das bereits vorhandene Prologprädikat `min_list`, sondern implementieren Sie es selbst. Beachten Sie dabei auch leere Listen. Überprüfen Sie Ihre Regel mit Beispielanfragen.

Aufgabe 3 (5 Punkte) Implementieren Sie eine Relation in einer Programmdatei `magic.pl`, welche eine Liste mit 3 Listen der Länge 3 als Einträgen, also eine 3×3 Matrix, daraufhin überprüft, ob sie die Zahlen 1 bis 9 als Permutation enthält und sich wie ein magisches Quadrat verhält. Das heißt, dass alle Spalten, Zeilen und beide Diagonalen der Länge 3 dieselbe Summe ergeben. Geben Sie ein Beispiel für eine solche Zahlenfolge an und schreiben Sie sie in den Programmcode als Kommentar.

Hinweis: Ob eine Liste A eine Permutation einer Liste B ist, kann man mit der eingebauten Relation `permutation(A, B)` überprüfen.