

Übungsserie 4

Programmierung mit Prolog

Hinweis: Verwenden Sie für die Übungsaufgaben den SWI-Prolog-Interpreter. Unter <http://www.swi-prolog.org> finden Sie unter anderem eine aktuelle Version für verschiedene Betriebssysteme und eine Dokumentation. Sie können den Interpreter mit dem Befehl `swipl` (in den Computerpools mit `pl`) starten. Man lädt 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.

Aufgabe 13 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(klaus,bernd).
20 parent(klaus,tom).
21 parent(simon,david).
22 parent(simon,klara).
23 parent(karin,david).
24 parent(karin,hugo).
25 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
- `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 14 Schreiben Sie eine Relation welche den Binomialkoeffizienten zweier natürlicher Zahlen $\binom{a}{b}$ bestimmt. Eine Definition finden Sie in Aufgabe 5 der zweiten Übungsserie. Verwenden Sie hierfür eine selbst implementierte Fakultätsrelation. Speichern Sie Ihr fertiges Programm in einer Datei `binom.pl` ab.

Aufgabe 15 Schreiben Sie Relationen zur Berechnung der folgenden Aufgaben in ein Prolog-Programm `rechnen.pl`. Beschreiben Sie darin auch in Kommentaren kurz Ihr Herangehen beziehungsweise wie Ihre Relationen zu benutzen sind.

- Das Quadrat einer Zahl.
- Das Minimum von drei Zahlen.
- Die Summe der natürlichen Zahlen von 1 bis n .
- Die Funktion `sign` ermittelt das Vorzeichen einer Zahl. Ausgabe: -1, 0 oder 1.

Aufgabe 16 Implementieren Sie eine Relation in einer Programmdatei `magisch.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. Hinweis: Ob eine Liste A eine Permutation einer Liste B ist, kann man mit der eingebauten Relation `permutation(A, B)` überprüfen.

Zusatzaufgabe 4 Lösen Sie das folgende Rätsel mit Hilfe einer Faktendatenbank, welche Sie in `raeuber.pl` speichern. Diese Programmdatei soll nur die gegebenen Fakten und die nötigen Regeln zum Lösen des Rätsels, aber keine von Ihnen zusätzlich herausgefundenen Fakten, enthalten! Das Rätsel lautet:

Gestern Nacht wurde der Tresor einer Bank ausgeraubt. Finden Sie unter den fünf Männern, welche die Kombination kennen, denjenigen heraus, der gestern abend allein war.

- Mr. Jones ist dick und fährt einen VW.
- Mr. Smith fährt einen Jaguar und ist Engländer.
- Der dünne Mann fährt einen Fiat.
- Mr. Poole ist Schotte.
- Mr. Kent trägt einen blauen Anzug.
- Der dicke Mann trägt einen schwarzen Anzug.
- Der dünne Mann ist Ire.
- Mr. Bright trägt einen grünen Anzug und fährt einen Renault.
- Der kleine Mann ist Schotte.
- Der dünne Mann trägt einen blauen Anzug.
- Der Mann normaler Statur ist Kanadier.
- Der Mann im grauen Anzug fährt einen Mazda.
- Der Mann im schwarzen Anzug ist Waliser.
- Der große Mann trägt einen braunen Anzug.
- Letzte Nacht war der Waliser mit dem Kanadier zusammen.
- Letzte Nacht war der große Mann mit dem Mazdafahrer zusammen.