

Perlen der Informatik 9. Übung

Es folgen ein paar recht einfache Aufgaben zu Prolog. Leider ist anscheinend zur Zeit kein funktionierender Prolog-Interpreter auf der Rechnerhalle installiert (was wir hoffentlich bald nachholen können). Die Installation auf einem eigenen Rechner sollte aber nicht so schwierig sein. Für unsere Perlen-Zwecke ist z.B. *GNU Prolog* geeignet, der auch für Windows verfügbar ist.

Aufgabe G9.1 Prädikate auf Listen

Eine Liste, die aus den Anfangselementen x_1, \dots, x_n sowie der Restliste xs zusammengesetzt ist, wird in Prolog durch die Notation

$$[x_1, \dots, x_n | xs]$$

dargestellt. Schreiben Sie Prolog-Prädikate zum

- Zusammenhängen zweier Listen
- Umdrehen einer Liste
- Testen, ob ein Element in einer Liste enthalten ist

Aufgabe G9.2 Zahldarstellung

Eine Möglichkeit, natürliche Zahlen in Prolog darzustellen besteht darin, einen nullstelligen Konstruktor `zero` sowie einen einstelligen Konstruktor `suc` zu verwenden. Damit kann z.B. die Zahl 4 durch den Term `suc(suc(suc(suc(zero))))` dargestellt werden.

Lösen Sie nun unter Verwendung der obigen Darstellung für natürliche Zahlen folgende Aufgaben:

- Schreiben Sie ein Prädikat `le(X, Y)`, das genau dann erfüllt ist, wenn die natürliche Zahl X kleiner oder gleich Y ist.
- Schreiben Sie ein Prädikat `add(X, Y, Z)`, das genau dann erfüllt ist, wenn die natürliche Zahl Z die Summe der natürlichen Zahlen X und Y ist.
- Schreiben Sie ein Prädikat `sub(X, Y, Z)`, das genau dann erfüllt ist, wenn die natürliche Zahl Z die Differenz der natürlichen Zahlen X und Y ist. Ist X kleiner als Y , so soll die Differenz Null sein.

Aufgabe G9.3 Permutation Sort

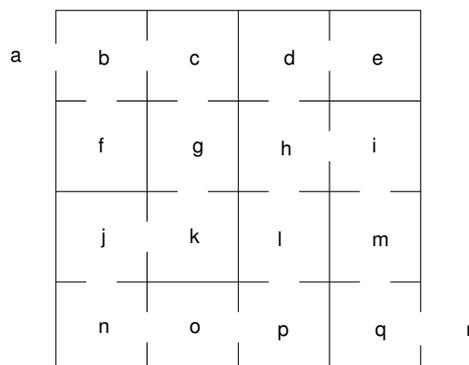
Für die Implementierung des folgenden Sortierverfahrens sollen Listen von natürlichen Zahlen in der obigen Darstellung verwendet werden.

- Schreiben Sie ein Prädikat `permute(Liste1, Liste2)`, das genau dann erfüllt ist, wenn *Liste₂* eine Permutation von *Liste₁* ist.
- Schreiben Sie ein Prädikat `sorted(Liste)`, das genau dann erfüllt ist, wenn *Liste* sortiert ist.
- Schreiben Sie mit Hilfe der Prädikate `permute` und `sorted` ein Prädikat `psort(Liste, SortierteListe)`

Was können Sie über die Effizienz dieses Sortierverfahrens sagen?

Aufgabe G9.4 Suche im Labyrinth

Es soll ein Weg im folgenden Labyrinth von einem Raum zu einem anderen gefunden werden, wobei jeder Raum nur einmal betreten werden darf.



Definieren Sie ein Prolog-Prädikat

`maze(Raum1, Raum2, Weg)`

das den Weg von *Raum₁* nach *Raum₂* berechnet und die Lösung *Weg* als Liste von Räumen liefert. Stellen Sie das Labyrinth von Aufgabe 1 durch mehrere Fakten dar. Die Existenz einer Tür zwischen zwei Räumen soll hierbei durch ein Faktum `door(raum1, raum2)` beschrieben werden (z.B. `door(a, b)`).

Beispiel:

?- maze(a,e,W).

W = [a, b, c, g, k, j, n, o, p, l, h, d, e]