Prof. Dr. Manfred Schmidt-Schauß
Künstliche Intelligenz/Softwaretechnologie

Fachbereich Informatik (20)
Johann Wolfgang Goethe-Universität Frankfurt am Main

Deduktionssysteme: Grundlagen und Anwendungen

Sommersemester 1999

Übungsblatt Nr. 9

Abgabe: Donnerstag 10. Juni 1999 vor der Vorlesung

Aufgabe 1 (15 Punkte)

Programmieren Sie in Prolog ein Prädikat `permutation/2`, welches für zwei Listen wahr ist, wenn sie Permutationen voneinander sind.

Aufgabe 2 (20 Punkte)

Es sei eine Menge von Grundfakten `R(a1,a2)`, ... gegeben. Programmieren Sie ein terminierendes Prädikat `TR/2`, das wahr ist wenn seine Argumente in der transitiven Hülle von `R/2` liegen.

Aufgabe 3 (15 Punkte)

Differenzlisten Manchmal ist es günstig, Listen als sogenannte Differenzlisten darzustellen und zu verarbeiten, denn diese ermöglichen ein effizienteres Aneinanderhängen von Listen. Eine schöne Syntax ist die Darstellung mit Minus. Man stellt z.B. `[1,2,3]` dar als `[1,2,3`X]-X—. Dadurch erhält man Zugriff auf das Ende einer Liste, was das effiziente Anhängen erlaubt. Differenzlisten sollen immer *allgemeinst* sein, d.h. `[1,2,3,4]-[4]` ist zu stark instantiiert.

Das Aneinanderhängen zweier Differenzlisten kann man dann so programmieren:

```
append-dl(H - Tail1, Tail1 - Tail2, H - Tail2)
```

1. Erklären Sie wie Differenzlisten funktionieren. Erklären Sie zunächst `append-dl/3`.
2. Was sind die Vor-, was die Nachteile von Differenlisten?
3. Programmieren Sie ein Prädikat `length-dl/2`, das wahr ist, wenn das zweite Argument die Länge der Differenliste ist, die als erstes Argument übergeben wird.