

Probeklausur Einführung in die symbolische Programmierung 2010 Vorlesung

Aufgabe 1: zur Sortierung

10 Punkte

1.1) Schreiben Sie eine Prozedur um zwei sortierte Listen in eine dritte Liste zu sortieren.

(benutzen Sie keine speziellen Built-In Sortierungsprädikate)

?- merge([2,5,6,6,8],[1,3,5,9],L).

L=[1,2,3,5,5,6,6,8,9]

1.2) Eliminieren Sie nun doppelte Einträge aus der Liste.

1.3) Schreiben Sie ein Programm, das auch unsortierte Listen sortieren kann

1.4) Die in der Vorlesung vorgestellte Variante von Quicksort funktioniert schlecht (Zeitkomplexität für bereits sortierte Listen.) Begründen Sie warum.

Aufgabe 2: zum Programmablauf

10 Punkte

Betrachten Sie folgendes Beispielprogramm:

p(4).

p(7) :- !.

p(9).

Schreiben Sie alle Antworten von Prolog (Backtracking mit ';'auslösen, nachdem eine Antwort gegeben wurde). Achten Sie auf die richtige Reihenfolge in der Prolog die Antworten gibt.

2.1) ?- p(X).

2.2) ?- p(X), p(Y).

2.3) ?- p(X),!, p(Y).

Aufgabe 3: was ist ein Prologprogramm

15 Punkte

3.1) woraus besteht ein Prologprogramm

3.2) erklären Sie die grundlegende Abarbeitungsweise eines Prologprogramms

3.3) erklären Sie anhand von append die Wirkungsweise eines Akkumulators

3.4) erklären Sie anhand der Negation die Wirkungsweise des Cut Prädikats.

Aufgabe 4: Grammatik

15 Punkte

Gegeben Sei die folgende Grammatik

s --> np, vp.

np --> det, n.

vp --> v, np.

det --> [a].

det --> [the].

n --> [cat].

n --> [mouse].
v --> [scares].
v --> [hates].

4.1) Zeigen Sie dass folgender Satz von der Grammatik akzeptiert wird: [the, mouse, scares, the, mouse]. Beschreiben Sie welche Prologregeln wann aufgerufen werden. Erklären Sie wie es zur Abarbeitung der Terminale kommt.

Aufruf ist: ?- s([the, mouse, scares, the, mouse],[]).

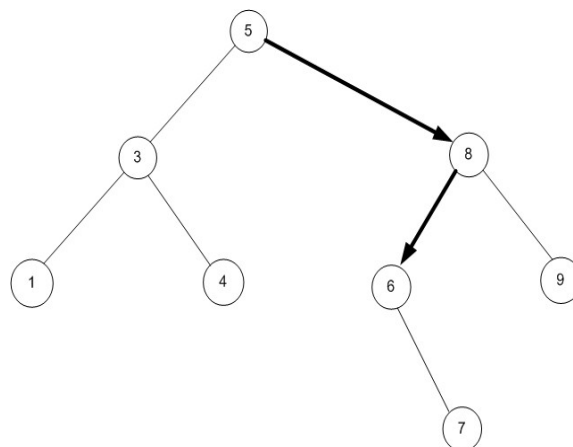
4.2) Benutzen Sie nun nur die Nominalphrase der Grammtik einschliesslich ihrer Elemente mit einem Beispielterminal nämlich [mouse]. Geben Sie die Schreibweise ohne DCG Pfeile im internen Format von Prolog an (DCG Regeln in Standardprolog aufgelöst)

4.3) Implementieren Sie nun den Plural für die obige Grammatik (mouse – mice, cat -- cats, scares – scare, hates – hate) geben Sie die notwendigen ergänzenden Grammatikregeln an. Führen Sie in das DCG-Format eine Variable ein, die die Numerus Kongruenz sicherstellt, also sicherstellt wenn das Nomen Plural ist, dass auch das Verb Plural ist (Sätze wie [the, mice scares, the, cat] sollen also ungrammatisch sein).

Zusatzaufgabe 5: Datenstrukturen

10 Punkte

Gegeben ist ein binärer Baum $D = t(\text{Left}, X, \text{Right})$, der von links nach rechts sortiert ist. Dies bedeutet alle Knoten im linken Subbaum sind kleiner als X und alle im rechten Subbaum sind grösser als X . Beide Subbäume sind wieder sortiert.



Gegeben Sei folgendes Prologprogramm:

```
in(X,t(_,X,_)).  
in(X,t(Left,Root,Right)) :-  
    Root > X,  
    in(X,Left).  
in(X,t(Left,Root,Right)) :-  
    Y > Root,  
    in(X,Right).
```

5.1) erklären Sie kurz was dieses Programm macht.

5.2) geben Sie die aufgerufenen Regeln + Variablenbelegungen für folgenden Aufruf an: ?- in(6,D). Wobei D der oben gezeichnete Baum mit der Wurzel 5 sein soll.