Prof. Dr. Stefan Brass
Institut für Informatik
Universität Halle

12.06.2006

Deduktive Datenbanken und Logische Programmierung — Blatt 4: Operatorsyntax, Aussagenlogischer Modellprüfer —

Aufgabe 8 — 6 Punkte

Ein rätselwütiger König stellt seine Gefangenen vor die Wahl zwischen zwei Türen, die entweder in die Freiheit oder in den Tod führen. An den Türen stehen Hinweise, von denen aber nur einer richtig ist, der andere ist falsch. Zum Beispiel steht an der linken Tür:

> "Diese Tür führt in die Freiheit, und die andere in den Tod."

An der rechten Tür steht:

> "Eine Tür führt in die Freiheit und eine Tür führt in den Tod."

Dies muß explizit gesagt werden, weil der König sich vorbehält, daß auch beide Türen in die Freiheit oder beide in den Tod führen können (tatsächlich ist er aber doch relativ fair).

Ihre Aufgabe besteht darin, ein Prolog-Programm zu schreiben, das Rätsel dieser Art löst (der König denkt sich ja jeden Tag ein neues aus). Es soll die Hinweise auf den beiden Schilderen einlesen, und dann die Lösung(en) des Rätsels ausgeben (es ist möglich, daß die Aussagen mehrere Lösungen offenlassen).

Natürlich müssen die Hinweise in einer formalen Syntax notiert werden, und zwar der Sprache der Aussagenlogik. Schreibt man "`l`" für "die linke Tür führt in die Freiheit" und "`r`" für "die rechte Tür führt in die Freiheit", so könnte man die Inschrift des ersten Schildes (linke Tür) so formulieren:

```
l und (nicht r).
```

Dabei sind "`und`", "`oder`", "`nicht`" die bekannten Operatoren $\wedge$, $\vee$, $\neg$ der Aussagenlogik. Die zweite Aussage lautet formal:

```
(l und (nicht r)) oder ((nicht l) und r).
```

Wenn Sie die Bindungsstärken der Operatoren richtig definieren ("`nicht`" stärker als "`und`", und dies wieder stärker als "`oder`"), so können Sie die Aussage auch ohne Klammern notieren.

Das Rätsel stammt aus dem Buch "Dame oder Tiger? — Logische Denkspiele und eine mathematische Novelle über Gödels große Entdeckung" von Raymond Smullyan (Krüger Verlag). In seiner Geschichte wartet hinter der Tür, die in die Freiheit führt, eine heiratswillige junge Dame, und hinter der anderen Tür ein hungriger Tiger.

a) Definieren Sie Operatoren "`und`", "`oder`", "`nicht`" in Prolog, so daß obige Eingaben möglich sind. Der Operator "`nicht`" soll am stärksten binden, "`und`" am zweitstärksten, und "`oder`" am schwächsten. Doppelte Negation soll ohne Klammern möglich sein, und die beiden binären Operatoren sollen linksassoziativ sein (Ketten ohne Klammern werden von links gebunden). Sie können Ihre Operator-Deklarationen mit folgendem Prädikat testen:

```
test :- write('Bitte Aussage eingeben: '),
        read(X),
        nl,
        write('Standardsyntax:' ),
        display(X).
```

b) Schreiben Sie nun ein Prädikat "`eval(A, L, R, W)`", das gilt, wenn die Aussage `A` den Wahrheitswert (wahr, falsch) `W` hat, sofern `L` und `R` die Wahrheitswerte für `l` und `r` sind. Dabei ist `A` ein Term mit den Funktoren `und`, `oder`, `nicht` und den Konstanten `l` und `r`. Wenn Sie meinen, daß ein anderes Prädikat für die Lösung der Rätsel besser wäre, müssen Sie nicht unbedingt `eval` wie oben definieren.

c) Schreiben Sie nun ein Hauptprogramm, das die Inschriften der Schilder an den beiden Türen (als aussagenlogische Formeln) abfragt, und dann die Modelle ausgibt (also jeweils "Freiheit" oder "Tod" für die beiden Türen). Es sollen dabei alle Modelle ausgegeben werden. Z.B. können Sie nach der Ausgabe von einem Modell das Prädikat "`fail`" aufrufen, das Backtracking erzwingt. Da diese Regel dann aber immer fehlschlagen wird, sollten Sie für das betreffende Prädikat noch eine weitere Regel angeben, die anschließend ausgeführt wird, und erfolgreich ist.

Wenn Sie anstatt der Symbole `l` und `r` lieber `diese_freiheit`, `diese_tod`, `andere_freiheit`, `andere_tod` verwenden wollen, ist das auch möglich (aber aufwändiger). Sie könnten auch weitere Junktoren hinzufügen.

Abgabetermin:

Bitte geben Sie Ihre Lösung bis zum Montag, den **19. Juni 2006**, ab.