

Introduction to Data and Knowledge Engineering

8. Tutorium



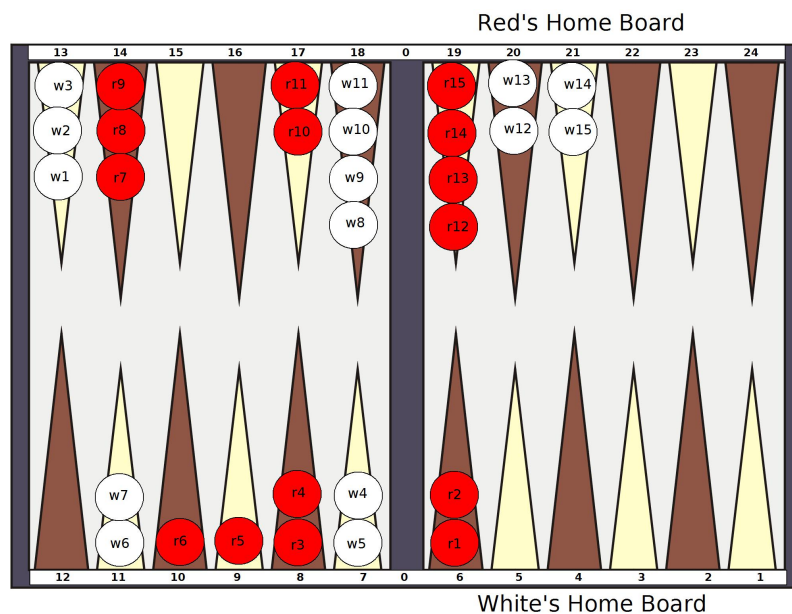
Tutorium 8: Prolog

18. Juni 2010

Aufgabe 8.6 Backgammon

In dieser Aufgabe wollen wir die Modellierung von Backgammon aus dem letzten Tutorium ergänzen, indem wir neue Prädikate hinzufügen. Machen Sie sich zunächst die Bedeutung der folgenden Prädikate aus Aufgabe 7.5 nochmal klar. Diese waren `player(P,C)`, `color(S,C)`, `point(S,O)`, `home(S)`, `player_stone(P,S)`, `can_beaten(S)`, `can_move(A,P)`, `start_move_out(P)`, und `player_win(P)`.

Auf der Homepage der Veranstaltung befindet sich eine Prolog Programm, das die in der folgenden Abbildung dargestellte Spielsituation sowie die Definition der schon besprochenen Prädikate enthält. Points, Stein, Spieler und Farben sind dort durch bestimmte Konstanten vertreten: Points als Zahlen $0 \dots 24$ wobei die 0 die Bar repräsentiert, die Steine mit einer geteilten Nummerierung $r1 \dots r15$ und $w0 \dots w15$, die Spieler heißen `barabara` und `george` und die Farben sind `red` und `white`.



- Schreiben Sie ein Prädikat `dice(P,X,Y)`, das die Augenzahlen von dem Wurf des Spielers `P` modelliert.
- Jeder Spieler setzt seine Steine entsprechend den gewürfelten Augenzahlen. Die Augenzahlen werden entweder zusammengezählt oder einzeln gesetzt. Schreiben Sie ein Prädikat, das die möglichen Züge für Stein `S` zurückgibt. Verwenden Sie das Prädikat `can_move(A,P)`.
- Definieren Sie ein Prädikat `all_possible_moves(P,S)`, das alle Steine zurückgibt, die der gegebene Spieler `P` ziehen kann.
- Definieren Sie ein Prädikat `beat_opponent(P,S)`, das alle Steine vom gegebenen Spieler `P` zurückgibt, die mindestens einen gegnerischen Stein schlagen können.

Aufgabe 8.7 DataLog vs. Prolog

Gegeben sei das folgende Programm.

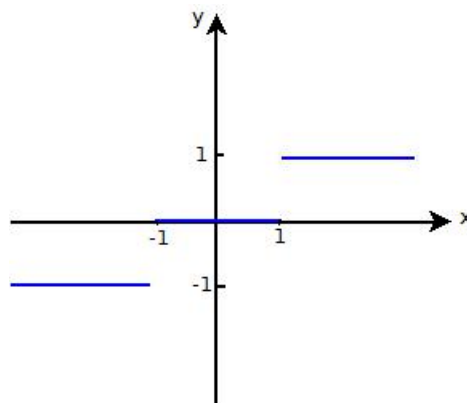
```
fatigue(ana).
fatigue(joe).
headache(john).
body_temperature(ana,38).
body_temperature(joe,37).
body_temperature(john,37).
hypothesis(Patient,flu):- hypothesis(Patient,fever), fatigue(Patient).
hypothesis(Patient,flu):- hypothesis(Patient,fever), headache(Patient).
hypothesis(Patient,fever):- body_temperature(Patient,X) , X > 37.5.
```

Sie können beim Lösen der Aufgaben die folgenden Abkürzungen für die Namen der vorkommenden Prädikate verwenden: $fa(X)$, $he(X)$, $bt(X,T)$, $hy(X, D)$.

- Betrachten Sie das erstmal als ein Datalog-Programm und bestimmen Sie den Fixpunkt, indem Sie angeben welche Fakten nach jeder EPP-Iteration hinzukommen.
- Wie würde Prolog die Queries $hypothesis(ana,flu)$ und $hypothesis(john,flu)$ abarbeiten? Skizzieren den SLD-Suchbaum, der auch die fehlgeschlagenen Alternativen enthält.
- Zeichnen Sie den Beweisbaum für die Query $hypothesis(ana,flu)$.

Aufgabe 8.8 Cut in Prolog

Betrachten Sie die folgende Funktion:



Welches der folgenden Programme in Prolog beschreibt die obige Funktion. Begründen Sie Ihre Antwort.

```
f(X,1) :- 1 < X.
f(X,0) :- X < 1.
f(X,-1) :- X < -1.
```

```
f(X,1) :- 1 < X, !.
f(X,0) :- X < 1, !.
f(X,-1) : X < -1, !.
```

```
f(X,1) :- 1 < X, !.
f(X,-1) :- X < -1, !.
f(X,0) :- X < 1, !.
```