

Opgave 2: Inleiding Prolog

Practicum Intelligente Systemen

februari 2014

Lees onze richtlijnen voor het maken van Prolog-programma's op de extra webpagina's bij het vak: http://www.cs.uu.nl/docs/vakken/b3is/richtlijnen_2013_2014.pdf.

Opgaves 1, 2 en 3 gaan alleen over hoofdstukken 1 & 2 van het boek van Blackburn. Het gebruik van recursie, lijsten, cut, etc. (uitgelegd in verdere hoofdstukken) is dus niet nodig, en zal bij deze eerste drie opgaves fout worden gerekend.

Om met de interpreter te leren omgaan raden we jullie aan om allereerst de "Practical Sessions" van hoofdstukken 1 en 2 te maken.

1. Maak exercise 2.4 uit Blackburn.

2. In het vorige practicum hebben we aspecten van de wumpus-wereld gerepresenteerd in eerste-orde logica. Aangezien Prolog gebaseerd is op eerste-orde logica, zouden we de representatie van onderdeel **2** van de vorige opgaves relatief eenvoudig moeten kunnen omvormen naar een kennisbank voor posities van kamers ten opzichte van elkaar. Schrijf een Prolog-kennisbank die een (lege) 4×4 wereld volledig beschrijft.

3. De wumpus-wereld is natuurlijk niet compleet zonder kuilen, een wumpus en het goud. Maak gebruik van je antwoorden bij onderdeel **3** van de vorige opgave voor het schrijven van een Prolog-predikaat `breeze/2` dat gegeven twee coördinaten vertelt of daar een bries waar te nemen is. Schrijf soortgelijke predikaten (`stench/2` en `glitter/2`) voor de stank van de wumpus en het glinsteren van het goud (deze zul je later als feiten aan de kennisbank kunnen toevoegen). Om deze predikaten te testen dien je (tijdelijk) gaten een wumpus en goud toe te voegen aan je kennisbank. Maak gebruik van `wumpus(X,Y)`, `pit(X,Y)` en `gold(X,Y)` om dit te bereiken. Bijvoorbeeld, als je in je kennisbank hebt toegevoegd `pit(2,2)` dan:

```
?- breeze(2,3).
```

```
yes.
```

Echter:

```
?- breeze(1,1).
```

```
no.
```

Nu gaan we werken met enkele basiselementen van Prolog, namelijk lijsten (behandeld in hoofdstukken 4 en 6) en operaties op lijsten (recursie, hoofdstuk 3 en aritmetiek, hoofdstuk 5).

Eerst nog even wat Prolog-conventies doornemen. Als we jullie vragen naar een predikaat `predikaat/3` dan betekent dat dat je een predikaat `predikaat` met ariteit 3 moet schrijven. Wanneer we vragen naar een predikaat `predikaat(+X, -Y, ?Z)` dan betekent dat dat van dit predikaat het eerste argument (X) altijd gegeven wordt bij aanroep, het tweede argument (Y) alleen maar als output zal dienen (en dus nooit gegeven wordt bij aanroep), en dat het derde argument (Z) zowel als input dan wel als output kan dienen (kan gegeven worden bij aanroep, maar hoeft niet).

Wederom is het aan te raden om de Practical Sessions van de genoemde hoofdstukken door te nemen. Ook het maken van enkele opgaven uit Blackburn kan een goede voorbereiding zijn.

4. Schrijf het `travel/3` predikaat, zoals bedoeld in de Practical Session van hoofdstuk 3 van Blackburn. (Online opgave 3.4, boek opgave 3.3.) Het is verstandig om eerst de twee voorgaande opgaves van die Practical Session te maken, hoewel deze niet hoeven te worden ingeleverd.

Je mag, als je dat mooier/makkelijker vindt, voor het derde argument een lijst maken. Deze lijst moet dan natuurlijk wel alle informatie bevatten die is vereist voor het juiste antwoord.

5. Schrijf een predikaat `somlijst/2` dat gegeven een numerieke lijst de som van de elementen geeft (`somlijst(+X, ?Y)`). Dus `?- somlijst([1, 2, 3, 4, 5], N)` . geeft `N = 15`.

6. Schrijf een predikaat `permutatie/2` dat gegeven een lijst een permutatie van die lijst teruggeeft (`permutatie(?X, ?Y)`¹). Dus

```
?- permutatie([a,b,c],P).
```

```
P = [a,b,c] ;
```

```
P = [b,a,c] ;
```

```
P = [b,c,a] ;
```

```
P = [a,c,b] ;
```

```
P = [c,a,b] ;
```

```
P = [c,b,a] ;
```

```
No
```

7. Schrijf een predikaat `sublijst(?X, +Y)` dat gegeven twee lijsten kijkt of de eerste lijst een sublist is van de tweede lijst. Echter, *niet* op de manier zoals in het voorbeeld op de college-slides (zie aldaar), maar zodanig dat alle *elementen* uit de eerste lijst in die volgorde, maar niet noodzakelijk aaneengesloten, voorkomen in de tweede lijst. Het predikaat kan bijvoorbeeld gebruikt worden om alle mogelijke sublists van een lijst te geven:

¹Op z'n minst is één van de twee argumenten gegeven!

```
?- sublijst(L, [a,b,c]).
```

```
L = [a,b,c] ;
```

```
L = [a,b] ;
```

```
L = [a,c] ;
```

```
L = [a] ;
```

```
L = [b,c] ;
```

```
L = [b] ;
```

```
L = [c] ;
```

```
L = [] ;
```

```
No
```

Echter: `sublijst([b,a],[a,b,c])` faalt.

8. Schrijf een predikaat `subsom/3` (`subsom(+X,+Y,?Z)`) dat alle sub-lijsten van een numerieke lijst met de exacte som `N` teruggeeft. Dus:

```
?- subsom([1,2,5,3,2],5,L).
```

```
L = [1,2,2] ;
```

```
L = [2,3] ;
```

```
L = [5] ;
```

```
L = [3,2] ;
```

```
No
```

9. Jullie hebben bij de eerste drie opgaven wellicht gemerkt dat Prolog een probleem heeft met variabele matching als je, bijvoorbeeld, `1 =< X` vraagt (probeer eens een lijst van alle kamers op te vragen met jouw uitwerking van opgave 2, als je hier ook `=<` operatoren hebt gebruikt). Om dit op te lossen gaan we een predikaat `tussen/3` schrijven (`tussen(+X,+Y,?Z)`), zodat `tussen(N1,N2,X)` door middel van recursie en backtracking `X` matcht met alle getallen tussen `N1` en `N2` (zodanig dat $N1 \leq X \leq N2$). Bijvoorbeeld:

```
?- tussen(1,4,X).
```

```
X = 1 ;
```

```
X = 2 ;
```

```
X = 3 ;
```

```
X = 4 ;
```

```
No
```

Puntenverdeling bij deze opgave; 1:10, 2:10, 3:10, 4:10, 5:5, 6:15, 7:15, 8:5, 9:10

Inleveren

- Aan de hand van je antwoorden op de bovenstaande opgaven bepalen we het tweede punt dat meetelt voor je algehele practicumcijfer.
- Inleveren dient te geschieden via 'submit', lever **ALLEEN** de (niet gezipte) .pl-file in!
- De deadline voor inleveren staat vast op 14-03-2014, voor 23.59 uur.
- Lees de richtlijnen voor de programma's op http://www.cs.uu.nl/docs/vakken/b3is/richtlijnen_2013_2014.pdf
- Zorg ervoor dat de opgave voorzien is van zowel namen als studentnummers. Neem dit als commentaar op **bovenaan** in je .pl file. Commentaar kan in .pl files worden opgenomen door middel van het '%' symbool.
- **Zorg dat je op tijd inlevert, we accepteren geen overschreidingen van de deadline.**