

Prolog toets 1: 2 Maart 1998

Beoordeling: aantal punten /100

vraag	aantal punten
1a	15
1b	15
2a	15
2b	15
3	20
4	20

Vraag 1: Representatie in Prolog

We willen Prolog gebruiken om een database op te bouwen over artikelen. De volgende gegevens moet in de database van artikelen komen te staan:

- Titel,
- Auteurs (met naam en voornaam),
- Keywords,
- als het een artikel in een tijdschrift betreft moeten de volgende gegevens opgeslagen worden: naam van het tijdschrift, jaar, volume, pagina's.
- als het een artikel in een conferentie is dan moeten de volgende gegevens opgeslagen worden: naam van de conferentie, editors, jaar, pagina's, plaats.
- als het een intern rapport betreft dan hoeven er geen extra gegevens opgenomen te worden.

Een aantal voorbeelden van de gegevens over artikelen is:

- Kurt Konolige, Abduction versus closure in causal theories, tijdschrift: AI, Volume 53, pag. 255–27, jaar 1992. Keywords: abduction, diagnosis.
- Eleni Stroulia en Ashok Goel, Reflective, Self-Adaptive Problem Solvers, conferentie artikel: European Knowledge Acquisition Workshop, 1994, pag.394–413, Hoegaarden. Keywords: knowledge acquisition, problem solvers.
- Marco Schaerf en Marco Cadoli, Tractable Reasoning via Approximation intern rapport. Keywords: approximate reasoning

vraag 1a

Bedenk een representatie in Prolog zodat je de volgende queries (vragen) makkelijk kunt stellen:

- Geef de titels van tijdschrift AI uit het jaartal 1997

- Geef de titels waaraan “Konolige” auteur of mede-auteur is.
- Geef de artikelen die door 2 auteurs geschreven zijn.

Bedenk dat je dus een verzameling moet maken, met voor elk artikel één feit.

```
artikel(Titel,AuteurList,Keywords,TypeArikel)
```

```
AuteurList: lijst van Auteurs
```

```
Auteurs:      auteur(VoorNaam,AchterNaam)
```

```
Keywords: lijst van constantes
```

```
TypeArtikel: tijdschrift(NaamTijdschrift, Jaar, Volume, Paginas)
```

```
              conference(NaamConferentie, Editors,Jaar, Paginas, Plaats)
```

```
              intern-raport
```

```
Paginas: pag(Van,Tot)
```

vraag 1b

Stel bovenstaande vragen ook met je gekozen representatie.

```
?- artikel(Titel,_,tijdschrift(ai,1997,_,_))
```

```
?- artikel(Titel,AuteurList,_, member(auteur(_,konolige),AuteurList)).
```

```
?- artikel(Titel,[_,_],_).
```

Vraag 2: Lijsten

Lijsten lijken op verzamelingen. Echter op een aantal punten zijn lijsten anders dan verzamelingen. Immers: het maakt niet uit of elementen dubbel voorkomen, en ook de volgorde maakt niet uit. Gebruik een lijst voor de representatie van een verzameling. Zo kan de set $\{1,2,3\}$ gerepresenteerd worden als $[1,2,3]$, $[2,1,3]$, en $[1,2,3,3]$.

Vraag 2a: Gelijkheid van twee verzameling

Schrijf een predicaat die slaagt als twee verzamelingen gelijk zijn (bijv. $[1,2,3]$ is gelijk aan $[2,2,3,3,1]$), en faalt als ze verschillend zijn. Bedenk eerst wat het gewenste gedrag is en schrijf dit ook op. Het predicaat ziet er uit als “gelijke-sets (Lijst1, Lijst2)”. Je mag er van uitgaan dat het predicaat gebruikt wordt als: *gelijke-sets(+L1,+L2)*. Verder is het niet toegestaan andere systeempredicaten dan member te gebruiken.

```
gelijke_sets(X,Y):-
    my_subset(X,Y),
    my_subset(Y,X).
```

```
my_subset([],_).
```

```
my_subset([X|Xs],Y):-
    member(X,Y),
    my_subset(Xs,Y).
```

Vraag 2b: Element van een verzameling

Schrijf een predicaat *elem-set(Lijst,Elem)* dat zich gedraagt als een element van een verzameling.

Dus:

elem-set(-E,+L) moet met backtracken alle elementen van de verzameling precies 1x opleveren.
Bijvoorbeeld *elem-set(X,[1,2,3,1])* slaagt 3 keer, met $X = 1$, $X = 2$, $X = 3$.

```
elem-set(E,L):-
    list_to_set(L,L1),
    member(E,L1).

list_to_set([],[]).
list_to_set([L|Ls],L1):-
    member(L,Ls),
    list_to_set(Ls,L1).

list_to_set([L|Ls],[L|Ls1]):-
    \+ member(L,Ls),
    list_to_set(Ls,Ls1).
```

Vraag 3: Termen

Wat geeft de Prolog interpreter voor een antwoord?

```
?: f(X) = f(f(a)).
   X = f(a); No

?: p(f(Y),g(A,B),f(X)) = p(f(a),g(X,b),f(a)).
   Y = a
   A = a
   B = b
   X = a ;

   No

?: p(A,B) = q(A,B).
   No

?: [a,b,c,d] = [X|Xs].
   X = a
   Xs = [b, c, d] ;
   No

?: [a,b|[1,2,3]] = [_|Xs].
   Xs = [b, 1, 2, 3] ;

   No

?: [[a]|[b]] = L.
   L = [[a], b] ;

   No
```

```

?: [a,b,c,b] = [X|[_,_,_|Y]].
    X = a
    Y = [] ;

    No

?: X is 1, Y is X+1, Y < 3.
    X = 1
    Y = 2 ;

    No

?: Y < 3, X is 1, Y is X+1.
    error WARNING: Arguments are not sufficiently instantiated

?: X=3, Y=X+1, Z is Y.
    X = 3
    Y = 3+1
    Z = 4 ;

    No

Gegeven operator definities:
:-op(700,xfy,and).
:-op(800,xfx,imp).
    type fout: had "impl" moeten staan in plaats van "imp"

?: (a and b) = and(a,b).
    Yes.

?: X=and(a,b).

    X = a and b ;

    No

?: and(a,impl(b,c))= a and b impl c.
    WARNING: Syntax error: Operator '=' conflicts with 'impl'

```

vraag 4

Gegeven het volgende programma:

```

q(1,2).
q(1,3).
r(1,2,3).
r(1,2,4).
r(1,3,5).

p(X,Y,Z):- q(X,Y),!,r(X,Y,Z).

```

Wat is het antwoord van Prolog op:

```
?- p(1,Y,Z).
```

Geef zowel het antwoord (ook bij backtracken “;”) en de bijbehorende trace (de goals die aangeroepen worden, en respectievelijk slagen en falen).

$p(1, Y, Z)$
 $q(1, Y), \neg, r(1, Y, Z).$
 $q(1, 2), \neg, r(1, 2, Z).$
 $r(1, 2, 3).$

$Y=2, Z=3;$

$r(1, 2, 4)$
 $Y=2, Z=4;$

No