

# Übungsblock 14

## Lösungen Übungsblatt 13:

**Aufgabe 1:** Erweitern der Wissensbasis zur Sprachanalyse, so dass bei Subjekt und Objekt Singular bzw. Plural unterschieden wird und nur entsprechende Artikel u. Prädikate erlaubt sind.

**Lösung 1:** ohne Geschlechter-Unterscheidung:

s_artikel(der) .	s_o_artikel(die) .
s_artikel(die) .	s_o_artikel(das) .
s_artikel(das) .	s_o_artikel(den) .
m_artikel(die) .	m_o_artikel(die) .

name(peter) .	name(elke) .	name(klaus) .
s_nomen(kuh) .		m_nomen(kuehe) .
s_nomen(auto) .		m_nomen(autos) .
s_nomen(oma) .		
s_nomen(mensch) .		m_nomen(menschen) .
s_nomen(katze) .		m_nomen(katzen) .
s_nomen(buch) .		m_nomen(buecher) .
s_nomen(kind) .		m_nomen(kinder) .
s_nomen(karte) .		m_nomen(karten) .

# Übungsblock 14

```
s_praedikat(besucht) .  
s_praedikat(malt) .  
s_praedikat(mag) .  
s_praedikat(faehrt) .  
s_praedikat(kocht) .  
s_praedikat(liest) .  
s_praedikat(spielt) .
```

```
m_praedikat(besuchen) .  
m_praedikat(malen) .  
m_praedikat(moegen) .  
m_praedikat(fahren) .  
m_praedikat(kochen) .  
m_praedikat(lesen) .  
m_praedikat(spielen) .
```

```
satz([K|[S|R]]) :- s_artikel(K), s_subjekt_mit_Art([K|S]),  
erstes(X,R,O), s_praedikat(X), objekt(O) .  
satz([K|[S|R]]) :- m_artikel(K), m_subjekt_mit_Art([K|S]),  
erstes(X,R,O), m_praedikat(X), objekt(O) .  
satz([K|R]) :- s_subjekt_ohne_Art([K]), erstes(X,R,O),  
s_praedikat(X), objekt(O) .  
satz([K|R]) :- m_subjekt_ohne_Art([K]), erstes(X,R,O),  
m_praedikat(X), objekt(O) .
```

```
erstes(X, [X], []) .  
erstes(X, [X|Y], Y) .
```

# Übungsblock 14

s\_subjekt\_mit\_Art([X,Y]) :- s\_artikel(X),s\_nomen(Y).

s\_subjekt\_mit\_Art([X,Y]) :- s\_artikel(X),name(Y).

m\_subjekt\_mit\_Art([X,Y]) :- m\_artikel(X),m\_nomen(Y).

s\_subjekt\_ohne\_Art([X]) :- s\_nomen(X);name(X).

m\_subjekt\_ohne\_Art([X]) :- m\_nomen(X).

objekt([]).

objekt(L) :- s\_objekt\_mit\_Art(L) ; s\_objekt\_ohne\_Art(L);  
                  m\_objekt\_mit\_art(L);m\_objekt\_ohne\_Art(L).

s\_objekt\_mit\_Art([X,Y]) :- s\_artikel(X), ( s\_nomen(Y);name(Y)).

s\_objekt\_ohne\_Art([X]) :- name(X).

m\_objekt\_mit\_Art([X,Y]) :- m\_artikel(X), m\_nomen(Y).

m\_objekt\_ohne\_Art([X]) :- m\_nomen(X).

# Übungsblock 14

## Lösung 2: mit Geschlechter-Unterscheidung:

```
s_artikel_m(der) .                s_o_artikel_w(die) .
s_artikel_w(die) .                s_o_artikel_s(das) .
s_artikel_s(das) .                s_o_artikel_m(den) .
m_artikel(die) .                  m_o_artikel(die) .
s_artikel(X) :-s_artikel_m(X);s_artikel_w(X);s_artikel_s(X) .
s_o_artikel(X) :-
s_o_artikel_w(X);s_o_artikel_m(X);s_o_artikel_s(X) .
name_m(peter) .                  name_w(elke) .                name_m(klaus) .
s_nomen_w(kuh) .                  s_nomen_s(buch) .
s_nomen_s(auto) .                  s_nomen_w(karte) .
s_nomen_w(oma) .                  s_nomen_s(kind) .
s_nomen_m(mensch) .                s_nomen_w(katze) .

s_nomen(X) :-s_nomen_m(X);s_nomen_w(X);s_nomen_s(X) .
name(X) :-name_w(X);name_m(X) .
```

# Übungsblock 14

...

```
s_subjekt_mit_Art([X,Y]) :- s_artikel_m(X), s_nomen_m(Y).
```

```
s_subjekt_mit_Art([X,Y]) :- s_artikel_w(X), s_nomen_w(Y).
```

```
s_subjekt_mit_Art([X,Y]) :- s_artikel_s(X), s_nomen_s(Y).
```

```
s_subjekt_mit_Art([X,Y]) :- s_artikel_w(X), name_w(Y).
```

```
s_subjekt_mit_Art([X,Y]) :- s_artikel_m(X), name_m(Y).
```

```
s_objekt_mit_Art([X,Y]) :-
```

```
s_o_artikel_m(X), (s_nomen_m(Y); name_m(Y)).
```

```
s_objekt_mit_Art([X,Y]) :-
```

```
s_o_artikel_w(X), (s_nomen_w(Y); name_w(Y)).
```

```
s_objekt_mit_Art([X,Y]) :- s_o_artikel_s(X), s_nomen_s(Y).
```

# Übungsblock 14

## Aufgabe 2:

Erstellen Sie für die Logelei von Blatt 11 ein Prolog-Programm, welches diese Logelei löst.

## Lösung 2.1:

```
%Lösung: X ist Liste von 3erTupeln [Santa,Name,Entfernung]
%Reihenfolge in X: Insel A,B,C,D
links_von(E11,E12,[E11|[E12|Rest]]).
links_von(E11,E12,[X|Rest]):-X\=E11,links_von(E11,E12,Rest).
uhr_neben(E11,E12,[E12|Rest]):-reverse(rest,[E11|H]).
run:- X =[[_,_,_],[26,_,_],[_,luvland,_],[_,_,7000]], %1,2,4
      member([28,_,4000],X), %3
      member([27,_,_],X), member([25,_,_],X), %damit alle Werte
      member([_,prieland,_],X), %bekannt sind
      member([_,_,6000],X),member([_,_,5000],X), %s.o.
      not(member([25,prieland,_],X)), %5
      member([25,luvland,_],X) ; member([27,luvland,_],X)), %1
      member([_,rauhland,5000],X);member([_,rauhland,7000],X)), %7
      ( uhr_neben([_,bukland,_],[_,rauhland,_],X) ;
        links_von([_,bukland,_],[_,rauhland,_],X)), %7
      write(X).
```

# Übungsblock 14

## Lösung 2.2 (mittels Permutationen):

```
pos(E1, [E1|Rest], 1) .
pos(E1, [X|Rest], Z) :- pos(E1, Rest, Y), Z is Y+1.

run:-
permutation([prieland, rauhland, luvland, bukland], PL),
    PL=[_,_,luvland,_],
    permutation([4000, 5000, 6000, 7000], PE),
    PE=[_,_,_,7000],
    permutation([25, 26, 27, 28], PZ),
    PZ=[_,26,_,_],
    pos(rauhland, PL, Z8), pos(5000, PE, Z9), (Z8==Z9; Z8==4),
    pos(bukland, PL, Z12), (Z8 == Z12+1),
    pos(28, PZ, Z1), pos(4000, PE, Z1),
    pos(25, PZ, Z3), pos(prieland, PL, Z4), Z3=\=Z4,
    pos(27, PZ, Z6), (Z3==3 ; Z6==3),
    write(PZ), nl, write(PL), nl, write(PE) .
```