

**Hausaufgabe (Abgabe: 22.12.2009): Lösen Sie bitte eine der drei folgenden Aufgaben.**  
**Hinweis: die Aufgabe 3 ist vermutlich die einfachste dieser Aufgaben.**  
**BN: Für einen BN müssen diese Woche keine Hausaufgaben gemacht werden.**

**Aufgabe 1:**

```
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %  
%  
% Datei:      <Name8>.dtr  
% Zweck:     Aufgabe 8: Flexionsmorphologie deutscher Adjektive  
% Autor:     Name, Datum  
%  
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
```

% Schreiben Sie eine DATR-Theorie, die die Flexionsendungen deutscher Adjektive erfasst. Benutzen Sie die gegebenen Show-Deklarationen.

```
% POSS (possessive Adjektive und (k)ein)      MEINE Hunde/Tiere/Katzen  
%           mask      neut      femn      plur  
%   nom   -           -           e           e  
%   akk   en          -           e           e  
%   gen   es          es          er          er  
%   dat   em          em          er          en
```

```
% STARK                                           GUTER Hund  
%           mask      neut      femn      plur  
%   nom   er          es          e           e  
%   akk   en          es          e           e  
%   gen   en          en          er          er  
%   dat   em          em          er          en
```

```
% GEMISCHT                                       ein GUTER Hund  
%           mask      neut      femn      plur  
%   nom   er          es          e           en  
%   akk   en          es          e           en  
%   gen   en          en          en          en  
%   dat   en          en          en          en
```

```
% SCHWACH                                        der GUTE Hund  
%           mask      neut      femn      plur  
%   nom   e           e           e           en  
%   akk   en          e           e           en  
%   gen   en          en          en          en  
%   dat   en          en          en          en
```

```
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
```

```
# show <sing nom mask> <sing akk mask> <sing gen mask> <sing dat mask>  
      <sing nom neut> <sing akk neut> <sing gen neut> <sing dat neut>  
      <sing nom femn> <sing akk femn> <sing gen femn> <sing dat femn>  
      <plur nom>     <plur akk>     <plur gen>     <plur dat>.
```

SCHWACH:

<sing nom mask> = e  
<sing akk mask> = e n  
<sing gen mask> = e n  
<sing dat mask> = e n  
<sing nom neut> = e  
<sing akk neut> = e  
<sing gen neut> = e n  
<sing dat neut> = e n  
<sing nom femn> = e  
<sing akk femn> = e  
<sing gen femn> = e n  
<sing dat femn> = e n  
<plur nom> = e n  
<plur akk> = e n  
<plur gen> = e n  
<plur dat> = e n.

GEMISCHT:

<sing nom mask> = e r  
<sing akk mask> = e n  
<sing gen mask> = e n  
<sing dat mask> = e n  
<sing nom neut> = e s  
<sing akk neut> = e s  
<sing gen neut> = e n  
<sing dat neut> = e n  
<sing nom femn> = e  
<sing akk femn> = e  
<sing gen femn> = e n  
<sing dat femn> = e n  
<plur nom> = e n  
<plur akk> = e n  
<plur gen> = e n  
<plur dat> = e n.

STARK:

<sing nom mask> = e r  
<sing akk mask> = e n  
<sing gen mask> = e n  
<sing dat mask> = e m  
<sing nom neut> = e s  
<sing akk neut> = e s  
<sing gen neut> = e n  
<sing dat neut> = e m  
<sing nom femn> = e  
<sing akk femn> = e  
<sing gen femn> = e r  
<sing dat femn> = e r  
<plur nom> = e  
<plur akk> = e  
<plur gen> = e r  
<plur dat> = e n.

POSS:

<sing nom mask> =  
<sing akk mask> = e n  
<sing gen mask> = e s  
<sing dat mask> = e m  
<sing nom neut> =  
<sing akk neut> =  
<sing gen neut> = e s  
<sing dat neut> = e m  
<sing nom femn> = e  
<sing akk femn> = e  
<sing gen femn> = e r  
<sing dat femn> = e r  
<plur nom> = e  
<plur akk> = e  
<plur gen> = e r  
<plur dat> = e n.

## Aufgabe 2:

```

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
%
% File:          GVERB1.DTR
% Purpose:      Simple German verb inflection
%              Simplified and adapted version of german_2.dtr
%              by Gerald Gazdar & Lynne Cahill, 9 September 1989
% Authors:     Wiebke Petersen
% Last modified: 11.12.2009
%
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

```

```

% Nehmen Sie folgende Erweiterungen an dieser Theorie vor:
% 1. Bilden Sie das Partizip Präsens der Verben.
% Fügen Sie dazu den Pfad <pres participle> zur Show-Deklaration hinzu
% 2. Fügen Sie die folgenden Verben hinzu:
% waten, sagen, denken, kegeln, schreiben

```

```

% regulaere Verben:
Verb_R: <infinitive> == "<root bare>" e n
        <pres> == "<root bare>" e
        <pres sing two> == "<root pres>" s t
        <pres sing three> == "<root pres>" t
        <pres plur> == "<infinitive>"
        <pres plur two> == "<root bare>" t
        <past> == "<root past>" t e
        <past sing two> == "<root past>" t e s t
        <past plur> == "<root past>" t e n
        <past plur two> == "<root past>" t e t
        <past participle> == g e "<root past participle>" t.

```

```

% regulaere Verben mit e-Epenthese
Verb_R_E: <> == Verb_R
           <root pres> == "<root bare>" e
           <pres plur two> == "<root bare>" e t
           <root past> == <root pres>.

```

```

% reguläre Verben auf s, z, x
Verb_R_S: <> == Verb_R
           <pres sing two> == "<root bare>" t.

```

```

% starke Verben mit vereinfachter Form in der 1. und 3. Person Präsens
Verb_1: <> == Verb_R
        <pres sing one> == "<root pres>"
        <pres sing three> == "<pres sing one>".

```

```

%-----

```

```

Arbeiten: <> == Verb_R_E
           <root bare> == a r b e i t.

```

```

Bringen: <> == Verb_R
         <root> == b r i n g
         <root past> == b r a c h.

```

```

Duerfen: <> == Verb_1
         <root> == d u e r f
         <root pres> == d a r f
         <root past> == d u r f.

```

```

Kaufen: <> == Verb_R
        <root> == k a u f.

```

```

Lernen: <> == Verb_R
        <root> == l e r n.

```

Mieten: <> == Verb\_R\_E  
<root bare> == m i e t.

Mixen: <> == Verb\_R\_S  
<root> == m i x.

Reisen: <> == Verb\_R\_S  
<root> == r e i s.

Sammeln: <> == Verb\_R % Verb mit e-Elidierung  
<root> == s a m m e l  
<infinitive> == s a m m e l n  
<pres sing one> == s a m m l e.

# show <infinitive>  
<pres sing one>  
<pres sing two>  
<pres sing three>  
<pres plur one>  
<pres plur two>  
<pres plur three>  
<past participle>  
<past sing one>  
<past sing two>  
<past sing three>  
<past plur one>  
<past plur two>  
<past plur three>.

# hide Verb\_R Verb\_R\_E Verb\_R\_S Verb\_1.

### Aufgabe 3:

```

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %
%
% File:             birds_1.dtr
% Purpose:          illustrates simple inheritance and defaults in DATR
% Author:           James Kilbury, 26 October 1992
% Related files:    molluscs.dtr, birds_2.dtr
% Version:          1.01
%   Copyright (c) University of Duesseldorf 1992. All rights reserved.
%
% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

% Erweitern Sie diese Theorie um Enten (eine "normale Ente" und Donald Duck)

BIRD:                 % root of the inheritance network;
    <> == no           % negation of all unspecif. properties
    <has claws> == yes
    <has beak> == yes
    <can fly> == yes.

EAGLE:
    <> == BIRD         % general (default) case
    <is eagle> == yes
    <is carnivorous> == yes.

Eric:
    <> == EAGLE
    <can> == no       % This simple DATR theory doesn't model the
    <is dead> == yes. % connection between being dead and being
                    % unable to do anything; note that a path
                    % <can fly> is unnecessary and undesirable.

Edwina:
    <> == EAGLE.      % a perfectly normal eagle

PENGUIN:
    <> == BIRD
    <is penguin> == yes
    <has claws> == no
    <can fly> == no
    <can swim> == yes.

Penny:
    <> == PENGUIN
    <is pilot> == yes % Again, this theory doesn't model the
    <can fly> == yes. % connection between pilots and flying.

Peter:
    <> == PENGUIN.    % a perfectly normal penguin

% % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % % %

# hide BIRD EAGLE PENGUIN.

# show <is dead>
      <is pilot>
      <is eagle>
      <is penguin>
      <is carnivorous>
      <can fly>
      <can swim>
      <has beak>
      <has claws>.

```