Universität des Saarlandes
Fachrichtung 6.2 – Informatik
J. Siekmann, S. Autexier, K. Fischer, E. Melis, C. Zinn
M. Schiller, E. Schulz, R.Vollmann

4th Practical Assignment in Artificial Intelligence (WS 2006/2007)
Solutions

- **Please submit your solution by sending an email to your tutor. The e-mail address of your tutor can be found on the webpage.**
- **Indicate the name and matriculation number of each student working in your project.**
- **Please comment your code!**

Knowledge representation has been developed from traditional approaches like semantic nets and frames. The KL-ONE model presented in the lecture bases on these approaches. In the fourth practical assignment you will learn an additional terminological knowledge representation tool—the KRIS system. KRIS has been developed in the working group of Prof. Siekmann at DFKI. You can find more information about this system on the tacos homepage `http://www.dfki.uni-sb.de/∼tacos`.

The KRIS system

The KRIS system allows to formulate knowledge and to deduce represented knowledge. In the following, we give a short introduction to this system using examples that base on relationships between persons in a family.

For the formulation of knowledge, the KRIS system distinguishes between conceptional knowledge and object knowledge. Conceptual knowledge is expressed in terminological components like *concept*, *role* and *attribute*. These components are called *TBox*. Knowledge about special facts (knowledge about objects) is expressed as assertional component. These components are called *ABox*.

Using the formulised knowledge, conclusions can be drawn by asking the KRIS system. Typical questions are:

- how is a concept defined?
- does a concept subsume an other concept?
- are two concepts disjoint?

TBox-File

In a TBox, all necessary and relevant concepts of the domain which should be illustrated are defined. Therefore, we define *concepts* and *roles*

In our domain, we define PERSON as concept. As this concept is not specified furthermore in detail, we introduce this concept as primitive.

```
(defprimconcept PERSON)
```

To distinguish between the different sexes, we introduce this concept and define that no PERSON can be male and female.

```
(defprimconcept MALE)
(defprimconcept FEMALE)
(disjoint MALE FEMALE)
```

As we would like to examine whether PERSON is still alive or already dead, we introduce both as disjoint concepts.

```
(defprimconcept DEAD)
(defprimconcept ALIVE)
(disjoint DEAD ALIVE)
```

Using these primitive concepts, we can define complex concepts like MAN who is a PERSON and MALE or WOMAN who is a PERSON and FEMALE.

```
(defconcept MAN (and PERSON MALE))
(defconcept WOMAN (and PERSON FEMALE))
```

As primitive relation we choose HAS-CHILD and IS-WIFE-OF. These are fomulates as roles.

```
(defprimrole Has-Child)
(defprimrole Is-Wife-Of)
```

Using the relations, we can define the concepts FATHER, MOTHER, PARENT, WIFE or WIDOW. For instance, a PERSON is a FATHER if the person is a MALE and has some children. The definition of complex concepts like FATHER-TWO-TO-FOUR is also possible as we can see in the following example.

```
(defconcept FATHER (and MAN (some Has-Child PERSON)))
(defconcept MOTHER (and WOMAN (some Has-Child PERSON)))
(defconcept PARENT (some Has-Child PERSON))
(defconcept WIFE (and WOMAN (some Is-Wife-Of PERSON)))
(defconcept WIDOW (and WOMAN (some Is-Wife-Of DEAD)))
(defconcept FATHER-TWO-TO-FOUR (and FATHER
              (atleast 2 Has-Child)
              (atmost 4 Has-Child)))
```

ABox-File

The knowledge about concrete facts is specified in a ABox. Therefore *assertions* are specified. As example, we define the following scenario:

```
(assertion Alfred MAN)
(assertion Alfred DEAD)
(assertion Berta WOMAN)
(assertion Berta Alfred Is-Wife-Of)
(assertion Christa WIDOW)
(assertion Berta Christa Has-Child)
(assertion Alfred Christa Has-Child)
(assertion Donald PERSON)
(assertion Donald FATHER-TWO-TO-FOUR)
(assertion Christa Donald Has-Child)
(assertion Christa Emil Is-Wife-Of)
```

```
(assertion Frida Woman)
(assertion Christa Frida Has-Child)
(assertion Grete PERSON)
(assertion Hans MAN)
(assertion Frida Hans Is-Wife-Of)
```

Inference
How conclusions can be drawn using the formulated knowledge is shown in the following example.
Download the KRIS system from `http://www.ags.uni-sb.de/~omega/teach/KI0607/material/kris.tar.gz`. You can also find the link on the lecture homepage. KRIS can be used with clisp. Extract kris.tar.gz and change into the KRIS directory.
Note to Windows users: In order to use clisp you have to add clisp.exe to your PATH variable. To do this go to System Properties → System → Advanced → Environment Variables. Then search for the PATH variable and click edit. Add a semicolon and the path to the directory of your clisp.exe.
In order to use KRIS perform the following steps:

```
> (load "load-kris.lsp")
...
#P"/home/stud/kiss/kris/Kris/load-kris.lsp"
> (load-tkb "sample")
Loading TBox ’sample.tbox’
;;; Warning: File "sample.tbox" does not begin with IN-PACKAGE.
;;; Loading into package "COMMON-LISP-USER"
#<tkb home/stud/kiss/kris/Kris/sample.tbox>
> (classify-tkb)
T
```

Now, you have loaded and classified KRIS and the TBox “sample.tbox”. Now you can ask questions about the conceptual formulated knowledge.

```
; Ask for a description about the definiton of Has-Child and FATHER
> (description ’Has-Child)
(HAS-CHILD)
> (description ’FATHER)
(AND MAN (SOME HAS-CHILD PERSON))
; Ask for all concepts that subsumes FATHER
> (all-supers ’FATHER)
c[*TOP*] c[PERSON] c[MALE] c[MAN] c[PARENT])
; Importantly, PARENT appears altough these concept were independently defined.
; Ask for all concepts that subsumes FATHER
> (all-subs ’FATHER)
(c[*BOTTOM*] c[FATHER-TWO-TO-FOUR])
; Ask for concepts that do not directly subsume FATHER and are directly subsumed respectiv
> (direct-supers ’FATHER)
(c[PARENT] c[MAN])
> (direct-subs ’FATHER)
(c[FATHER-TWO-TO-FOUR])
; Ask whether FATHER subsumes MAN
> (subsumes ’FATHER ’MAN)
NIL
```

```
; Ask whether MAN subsumes FATHER
> (subsumes ’MAN ’FATHER)
T
; Ask whether MAN and WOMAN are disjoint concepts
> (disjoint-concepts ’MAN ’WOMAN)
T
Ask whether two concepts are equivalent
> (equivalent-concepts ’(and MAN (some Has-Child PERSON))
                       ’(and MAN PARENT))
T
; Load the ABox
> (load-akb "sample")
loading abox ’sample.abox’
;;; Warning: File "sample.abox" does not begin with IN-PACKAGE.
;;; Loading into package "COMMON-LISP-USER"
#<akb /home/stud/kiss/kris/Kris/sample.abox>
; check of consistency
> (consistent-akb)
T
; Compute for each object the specific concept.
> (realize-akb :output-mode :name)
Realizing object HANS
...
Realizing object ALFRED
T
((ALFRED (c[DEAD] c[FATHER])) (BERTA (c[WIDOW] c[WIFE] c[MOTHER]))
(CHRISTA (c[WIDOW] c[MOTHER])) (DONALD (c[FATHER-TWO-TO-FOUR]))
(EMIL (c[*TOP*])) (FRIDA (c[WIFE])) (GRETE (c[PERSON])) (HANS (c[MAN])))
; Ask whether an object belongs to a concept
> (is-instance-of ’GRETE ’PERSON)
T
> (is-instance-of ’GRETE ’WOMAN)
NIL
> (is-instance-of ’EMIL ’DEAD)
NIL
; Compute all concepts where Christa belongs
> (individual-concepts ’Christa)
(c[PARENT] c[WOMAN] c[FEMALE] c[PERSON] c[*TOP*] c[MOTHER] c[WIDOW])
; Evaluate all specific concepts that belong to Christa
> (most-specific-concepts ’Christa)
(c[WIDOW] c[MOTHER])
```

Exercise P4.1: **40 P**

1. Extend the TBox "sample.tbox" using definitions of the concepts GRANDFATHER, GRANDMOTHER, GRANDPARENT, BROTHER, SISTER. (12 P)

Solution:

```
% ;;;=================================================================
;;;TBox=================================================================
(defprimconcept PERSON)
(defprimconcept MALE)
(defprimconcept FEMALE)
(disjoint MALE FEMALE)
(defprimconcept DEAD)
(defprimconcept ALIVE)
(disjoint DEAD ALIVE)

(defconcept MAN (and PERSON MALE))
(defconcept WOMAN (and PERSON FEMALE))

(defprimrole Is-Child-Of)
(defprimrole Is-Husband-Of)
(defconcept FATHER (and MAN (some Is-Child-Of PERSON)))
(defconcept MOTHER (and WOMAN (some Is-Child-Of PERSON)))
(defconcept PARENT (some Is-Child-Of PERSON))
(defconcept WIFE (and WOMAN (some Is-Husband-Of PERSON)))
(defconcept WIDOW (and WOMAN (some Is-Husband-Of DEAD)))
(defconcept FATHER-TWO-TO-FOUR (and FATHER
                                    (atleast 2 Is-Child-Of)
                                    (atmost 4 Is-Child-Of)))
;;;=================================================================
(defconcept GRANDMOTHER (and MOTHER (some Is-Child-Of PARENT)))
(defconcept GRANDFATHER (and FATHER (some Is-Child-Of PARENT)))
(defconcept GRANDPARENT (or GRANDFATHER GRANDMOTHER))

(defprimrole Is-Sibling-Of)
(defconcept BROTHER (and MAN PERSON (some  Is-Sibling-Of PERSON)))
(defconcept SISTER (and FEMALE PERSON (some  Is-Sibling-Of PERSON)))


% ;;;=================================================================
```

---

2. Extend the ABox with the persons IDA and JUERGEN so that for each of these concepts there exists an instance. (12 P)

Solution:

```
% ;;;=================================================================
 ;;;================================================================
 ;;; sample session
 ;;; file "sample.abox"
 (assertion Alfred MAN)
 (assertion Alfred DEAD)
 (assertion Berta WOMAN)
 (assertion Berta Alfred Is-Husband-Of)
 (assertion Christa WIDOW)
 (assertion Berta Christa Is-Child-Of)
 (assertion Alfred Christa Is-Child-Of)
 (assertion Donald PERSON)
 (assertion Donald FATHER-TWO-TO-FOUR)
 (assertion Christa Donald Is-Child-Of)
 (assertion Emil Christa Is-Husband-Of)
 (assertion Frida Woman)
 (assertion Christa Frida Is-Child-Of)
 (assertion Grete PERSON)
 (assertion Hans MAN)
 (assertion Frida Hans Is-Husband-Of)
 ;;; end of file "sample.abox"
 ;;;================================================================
 (assertion Ida WOMEN)
 (assertion Ida Christa Is-Sibling-Of)
 (assertion Juergen Man)
 (assertion Juergen Christa Is-Sibling-Of)


% ;;;=================================================================
```

3. Explain why KRIS answers with `NIL` and not `T` asking the question `is-instance-of 'GRETE 'WOMAN`. (8 P)

Solution:

As (assertion GRETE PERSON) is the only assertion, `NIL` *will be returned.*

4. Explain why KRIS answers with `NIL` asking the question `is-instance-of 'EMIL 'DEAD` although Emil is married with Christa and Christa is a widow. (8 P)

Solution:

If we would have specified in the tbox that every person has exactly one spouse in his life we could conclude that:

- *Christa is widow therefore there is a husband that is dead*
- *Christa has as most on husband*
- *Emil is the husband of Christa, therefore Emil is dead.*

Exercise P4.2: **30 P**

Define for the following facts a TBox.

A human is creature. Scientists are humans. Humans have jobs. Technical jobs are also jobs. Computer science is a technical job. A technician is a human who practises a technical job. A computer scientist is human who practises computer science as job. A technical computer scientist is a technician that practises computer science.

Evaluate the relationship between a technician, technical computer scientist and computer scientist using the KRIS system.

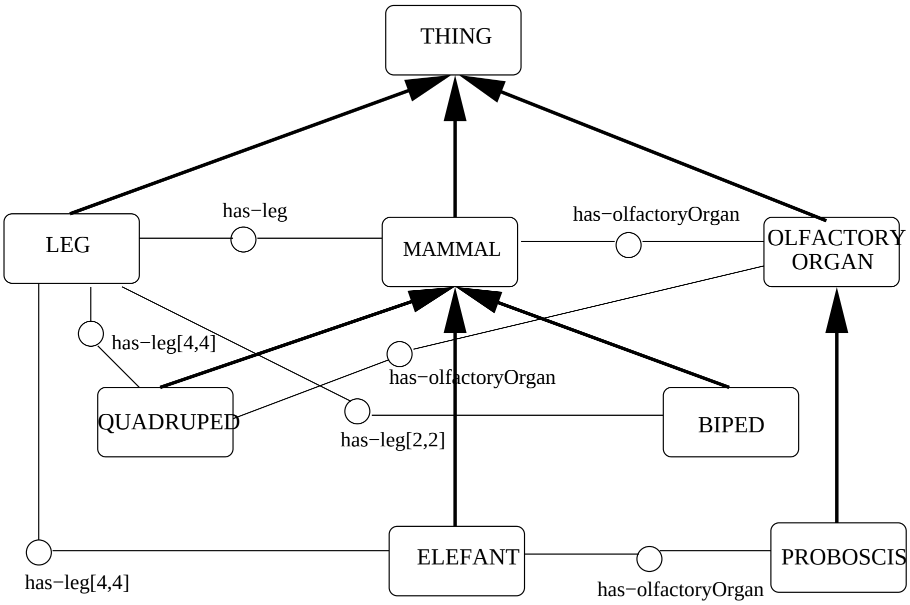
Solution:

```
% ;;;=================================================================
 ;;;=================================================================
 ;;; sample session
 ;;; file "sample.tbox"
(defprimconcept CREATURE)
(defprimconcept SCIENCE)
(defconcept HUMAN (and CREATURE (some practises JOB)))
(defconcept SCIENTIST (and HUMAN (some practises SCIENCE)))

(defprimconcept JOB)
(defprimconcept PROGRAM)
(defprimconcept APPARATUS)
(defconcept TECHNICAL_JOB (AND JOB (some involves APPARATUS)))
(defprimrole executes)
(defconcept COMPUTER (and APPARATUS (some executes PROGRAM)))
(defprimrole involves)
(defconcept COMPUTER_SCIENCE (AND TECHNICAL_JOB (some involves COMPUTER)))

(defprimrole practises)
(defconcept TECHNICIAN (and HUMAN (some practises TECHNICAL_JOB)))
(defconcept COMPUTER_SCIENTIST (and HUMAN (some practises
                                                COMPUTER_SCIENCE)))
(defconcept TECHNICAL_COMPUTER_SCIENTIST (and TECHNICIAN
                                              (some practises COMPUTER_SCIENCE)))

% ;;;=================================================================

;;; Testing:

(subsumes 'job 'technical_job)
; > T
(subsumes 'technical_job 'job)
; > NIL
(subsumes 'job 'computer_science)
; > T
```

```
(subsumes 'human 'computer_scientist)
; > T
(subsumes 'technician 'technical_computer_scientist)
; > T
(subsumes 'computer_scientist 'technical_computer_scientist)
; > T
(direct-supers 'technical_computer_scientist)
; > (c[TECHNICIAN])
(direct-supers 'computer_scientist)
; > (c[TECHNICIAN])
(equivalent-concepts 'technical_computer_scientist 'computer_scientist)
; > T
```

Exercise P4.3: **30 P**

Define a TBox for the semantic net illustrated in the following figure.



Extend the TBox with the concepts `HUMAN`, `NOSE`, and the role `has-nose` and formulate the following questions:

- Have elephants four legs?
- Are humans and elefants disjoint?
- Do elefants exist that have more than one proboscis?
- Do humans exist that have noses?

Solution:

```
% ;;;==============================================================
 ;;;================================================================
;;; abox
```

```
(in-package :kris)
(assertion Clyde ELEFANT)
(assertion John HUMAN)

% ;;;================================================================

% ;;;================================================================
;;;================================================================
;;; tbox
(defprimrole has-leg)
(defprimrole has-olfactoryOrgan)
(defprimconcept LEG)
(defprimconcept OLFACTORY_ORGAN)
(defprimconcept PROBOSCIS OLFACTORY_ORGAN)

(defconcept MAMMAL (and
                         (some has-olfactoryOrgan
                               OLFACTORY_ORGAN)
                         (some has-leg LEG)))

(defconcept THING (or MAMMAL OLFACTORY_ORGAN LEG))

(defconcept QUADRUPED (and MAMMAL
                             (atleast 4 has-leg)
                             (atmost 4 has-leg)))
(defconcept BIPED (and MAMMAL
                             (atleast 2 has-leg)
                             (atmost 2 has-leg)))
(defconcept ELEFANT (and MAMMAL
                         (atleast 4 has-leg)
                         (atmost 4 has-leg)
                         (some has-olfactoryOrgan
                               PROBOSCIS)))

;----------------------------------------------
(defconcept HUMAN (and BIPED
                   (some has-nose NOSE)))
(defconcept NOSE OLFACTORY_ORGAN)
(defprimrole has-nose)

% ;;;================================================================

;;; Questions:

;;;(1) Does an elefant have (exactly) four legs?

(subsumes
          '(and (atleast 4 has-leg)
             (atmost 4 has-leg))
          'elefant)
```

```
; > T

;;;(2) Are elefants and humans disjoint?

(disjoint-concepts 'human 'elefant)

; > T

;;;(3) Do elefants exist that have more than one proboscis?

(disjoint-concepts 'elefant '(atleast 2 has-olfactoryOrgan PROBOSCIS))
; > NIL  ---> so in our model of the world there are elefants with
              more than one proboscis


;;;(4) Do humans exist that have noses?

(disjoint-concepts 'human '(some has-nose NOSE))
; > NIL  ---> so there are humans who have noses
```