Course: Deduction Systems (DS)
Lecturer: Dr Paola Bruscoli
ICCL/Institute for Artificial Intelligence/Computational Logic
Summer Semester 2012

Coursework for Deduction Systems – due 30/06/2012

General Notes:
This coursework is meant to be individual work to be carried out of lecturing and tutorial times.
Its assessment will contribute to the 20% of the final grade for this course ONLY.
Moreover, purpose of this coursework, is to offer students the possibility of a self-assessment, on questions of comparable nature that could be asked during an examination.
Please print this page, sign the declaration below, filling all the remaining fields, including the date of submission.
You may add as many sheets for your solutions as you want, be sure that all your solutions to the single exercises are numbered accordingly.
Before submitting your coursework, please make sure that all sheets are adequately and firmly stapled together.
Deadline for submission is strict; you may submit your work by posting it in the white mailbox out of room 2001.
Good luck!

Declaration:
Hereby, I, (name/family name) ....................................................................
having student registration number .............................................................,
declare that the solution to this coursework is the result of individual work,
and that I am the only author of it.
I have read the general notes and I have submitted a total of (indicate number)............ sheets, including the present one; all sheets are stapled.

Date of submission ....................................... Signature.................................
Exercise 1 [points: 10]
Give the heap representation of the query $p(g(f(X, Y)), g(X), f(Y, Z))$ of $L_0$, explaining how you obtained it and specify the instructions used to load the heap.

Exercise 2 [points: 10]
Compile the program $p(V, g(a), f(b, V))$ of $L_0$, giving all details.

Exercise 3 [points: 10]
Explain why in the compilation of a $L_0$ program we need to take into account a write mode and a read mode. Do we need to take into account this also for the compilation of a $L_0$ query? Why?

Exercise 4 [points: 5]
Describe what is meant by the expressions “flattened form” and what elements of $L_0$ it refers to.

Exercise 5 [points: 5]
Explain what is meant by “dereferencing” and give an example of a situation where it would be necessary to consider it.

Exercise 6 [points: 4]
Is dereferencing related to backtracking at the level $M_0$? Explain your reasons.

Exercise 7 [points: 10]
Terms $g(f(Y, a), Y)$ and $g(X, b)$ are stored at the heap addresses 10 and 50 respectively, and variables $X$ and $Y$ at the addresses 3 and 7 respectively. Trace the effects of executing $\text{unify}(10, 50)$ and verify that it terminates with the eventual dereferenced binding from the addresses 3 and 7 corresponding to the appropriate unifer.

Exercise 8 [points: 6]
Explain the following, referring to $M_1$:

(1) Why do we need to consider the instruction size in $M_1$?
(2) In what sense is the instruction “call” called a control instruction?
(3) Why do we introduce the use of argument registers?

Exercise 9 [points: 4]
For the query and program of exercises 1 and 2, return the set of instructions of $M_1$.

Exercise 10: [points: 4]
Consider the program in $L_2$: $p(X, Y, Z) : -p(a, Y, Z), q(Z, U), r(U, b)$. Indicate the permanent and temporary variables. Do the same for the program $p(X, a, Y) : -q(X, Y)$.

Exercise 11: [points: 6]
Explain why the notion of environment is introduced in $M_2$.

Exercise 12: [points: 6]
Explain the reasons for introducing the instructions "allocate" and "deallocate" and their functioning.

Exercise 13: [points: 4]
Does the compilation of allocate and deallocate instruction modify the Heap? Explain.
Exercise 14: [points: 10]
Consider the program in $L_2$: $p(X, Y, Z) : \neg p(a, Y, Z), q(Z, U), r(U, b)$. Describe the environment associated to it in the stack and give the instructions in $M_2$ to compile it.

Exercise 15: [points: 4]
What is meant by “choice point” and the name of the register storing it, in the machine $M_3$.

Exercise 16: [points: 4]
State the language of clauses and queries in $L_3$.

Exercise 17: [points: 6]
Explain why it is necessary to keep the content of $B$ and of $E$ in the same stack.

Exercise 18: [points: 6]
Describe the trail, what it is needed for, and why we use it in $M_3$.

Exercise 19: [points: 10]
Explain the consequences that adding backtracking in $M_3$ have, on the instruction 'allocate' and 'call'.

Exercise 20: [points: 10]
Write $M_3$ code to compile the following program in $L_3$

\[
\begin{align*}
  p(X, X) & : \neg p(f(Y), X), p(Y, 0). \\
  p(X, 0) & : \neg p(X, f(1)). \\
  p(1, 0) & :
\end{align*}
\]

Exercise 21: [points: 10]
Write $M_3$ code to compile the following program in $L_3$ composed by just one fact: $p(a, [a, X], [f(a), X])$.

Exercise 22 [points: 4]
Give all answer sets of the following programs $P_1$ and $P_2$:

$P_1$ :

\[
\begin{align*}
  p & :\neg q, \not r. \\
  q & :\neg r. \\
  s & .
\end{align*}
\]

$P_2$ :

\[
\begin{align*}
  p & :\not p, \not r. \\
  p & :\neg q, \not r. \\
  q & :\neg r. \\
  s & .
\end{align*}
\]

Exercise 23 [points: 6]
Referring to $P_1$ and $P_2$ from exercise 22, show that $P_1$ is stratified and that $P_2$ is not stratified.

Exercise 24 [points: 10]
Consider the following logic programs:
\( P_1: \)

- football :- not cancel.
- cancel :- rain.
- rain :- not \( \neg \)cloudy.
- cloudy.

\( P_2: \)

- football :- not cancel.
- cancel :- rain.
- rain :- not \( \neg \)cloudy.
- \( \neg \)cloudy.

where ‘\( \neg \)’ denotes classical negation.

(1) Modify \( P_1 \) and \( P_2 \) so that they do not contain classical negations as operator.

(2) Check that the answer set of \( P_1 \) contains cancel.

(3) Check that the answer set of \( P_2 \) contains football.

**Exercise 25** [points: 4]

Show that the following program does not have any answer set:

\[
\text{p} \lor \text{q} :- \text{not q}.
\]

\[
\text{q} :- \text{p}.
\]

**Exercise 26** [points: 10]

The following graph \( G \) can be coloured used using three colours red, green and blue.

(1) Write an answer set program similar to the one presented in slides to colour this graph so that no two adjacent vertices have the same colour.

(2) Use grounding and domain restriction to use search algorithms to find possible colourings of the graph \( G \).
**Exercise 27** [points: 10]
Consider the following program:

\[ p : - q, \text{ not } r. \]
\[ q : - r. \]
\[ s. \]

(1) Use searching and propagation rules to drive the stable model, considering the predicates in the following order \( r, q, p \) and \( s \).
(2) Indicate a heuristic, among those presented in the slides, one could use to guess the best derivation. How would the derivation be, using this heuristic?

**Exercise 28** [points: 10]
Consider the following formulae:

\[ F_1 : ((p \lor q) \land (\neg p \lor q)) \rightarrow q \]
\[ F_2 : (p \land q) \rightarrow (p \lor q) \]

(1) Use tableau method to prove their validity.
(2) Convert them to definitional normal form (DNF), specifying each step of the conversion.

**Exercise 29** [points: 4]
Explain the reasons for using definitional normal form (DNF) instead of conjunctive or disjunctive normal form for transforming a formula prior the use of a tableau method (and other refutation methods).

**Exercise 30** [points: 4]
Convert the following formula to Skolem conjunctive normal form (CNF).

\[ (\forall X)(\exists Z)(\forall Y)(\exists T)((p(X, Y) \rightarrow q(X, Y, Z)) \lor r(X, T)) \]

**Exercise 31** [points: 6]
Use tableau method to decide whether the following formula is valid or not:

\[ (\exists Y)(\forall X)(p(X, Y)) \rightarrow (\forall X)(\exists Y)(p(X, Y)). \]

**Exercise 32** [points: 10]
Consider the following sets of concept descriptions, role names and DL rules:

\[ N_C = \{ \text{Human, Female, Man, Woman, Mother, Father, Parent} \} \]
\[ N_R = \{ \text{HasChild} \} \]

Rules:

\[ \text{Woman} \equiv \text{Human} \land \text{Female} \]
\[ \text{Man} \equiv \text{Human} \land \neg \text{Female} \]
\[ \text{Mother} \equiv \text{Woman} \land \exists \text{HasChild}.\text{Human} \]
\[ \text{Father} \equiv \text{Man} \land \exists \text{HasChild}.\text{Human} \]
\[ \text{Parent} \equiv \text{Mother} \lor \text{Father} \]

(1) Consider the following interpretation \( \mathcal{I} = (\Delta^\mathcal{I}, \mathcal{I}) \):

\[ \Delta^\mathcal{I} = \{ \text{Lucie, Frank, Tim, Jane, Josephine, Bob, Jack, Julia} \} \]
Human = \{ Lucie, Frank, Tim, Jane, Josephine, Bob, Jack, Julia \}
Female = \{ Lucie, Jane, Josephine, Julia \}
Woman = \{ Lucie, Jane, Josephine, Julia \}
Man = \{ Frank, Tim, Bob, Jack \}
Mother = \{ Lucie, Jane, Josephine \}
Father = \{ Frank, Tim, Bob \}
Parent = \{ Lucie, Frank, Tim, Jane, Josephine, Bob \}
HasChild = \{ (Lucie, Josephine), (Frank, Josephine),
             (Tim, Bob), (Jane, Bob), (Josephine, Jack),
             (Bob, Jack), (Josephine, Julia), (Bob, Julia) \}

Verify that this interpretation models the given rules.

2) Write an interpretation which does not satisfy the given rules.

3) Show that the following statements hold for the interpretation given in part 1:
   • Parent subsumes Human
   • Human does not subsumes Parent

4) Express the following concept definitions in terms of constraints:
   Woman = \text{Human} \cap \text{Female}
   Father = \text{Man} \cap \exists \text{HasChild.Human}

Exercise 33: [points: 5]
Consider the term rewriting system \( R \)
\[
x + x \rightarrow x.
\]
\[
x + (a + y) \rightarrow x + y
\]

(1) How many redexes are in \((a + a) + (a + a)\)?
(2) Is \(((a + a) + (a + b), a + b)\) in the transitive closure of \( \rightarrow_R \)?
(3) Is \((b, (a + b))\) in the transitive closure of \( \rightarrow_R \)? In the reflexive closure
   of \( \rightarrow_R \)? In the transitive reflexive closure of \( \rightarrow_R \)?

Exercise 34: [points: 3]
Let \( f \) be commutative.
Is \( f(f(c, b), f(a, f(a, b))) =_{C} f(f(a, f(b, a)), f(b, c)) \)?

Moreover, let \( g \) be commutative and associative.
Is \( f(a, g(f(a, b), g(a, f(a, c)))) =_{AC} f(g(g(f(c, a), a), f(a, b)), a) \)?

Exercise 35: [points: 4]
Let \( R \) be the term rewriting system in exercise 34, and let \( E \) the equational
theory stating that + is associative and commutative. Does the term
\( a + (b + (a + b)) R/E \)-rewrite to \( b + a \)? And \( (R, A) \)-rewrite?

Exercise 36: [points: 10]
How would the \( R \) and the \( E \) of exercise 35 be expressed in TOM?