

# Principles of Knowledge Representation and Reasoning

B. Nebel, S. Wöflf, J. Hué  
M. Westphal  
Winter Semester 2012/2013

University of Freiburg  
Department of Computer Science

## Exercise Sheet 8

**Due: December 19th, 2012**

### Exercise 8.1 (CUMULATIVE LOGICS, 2)

Show that in system **C** the rule (MPC) can be derived.

$$\frac{\alpha \vdash \beta \rightarrow \gamma, \quad \alpha \vdash \beta}{\alpha \vdash \gamma}$$

In your proof you may use the five basic rules of system **C**, the derived rules *Supraclassicality*, *Equivalence*, and *And*, as well as propositional logic tautologies.

### Exercise 8.2 (CUMULATIVE MODELS, 1+1+1+1)

We consider the rules *MPC*, *EHD*, *Cumulativity*, and *Monotonicity*. Which of these rules are valid in all cumulative models? Provide proofs or counterexamples.

### Exercise 8.3 (WELL-FOUNDED SEMANTICS FOR LOGIC PROGRAMS, 6)

The aim of this exercise is to implement a program that computes the well-founded models of a logic program. The well-founded semantics can be seen as a three-valued version of the answer set semantics that makes safe inferences only (see <http://cs.ucsc.edu/~avg/Papers/wf.ps><sup>1</sup> for a more comprehensive introduction).

We confine to simple normal logic program, i.e., no special statements, no cardinality rules, no *classical* negation. There is always one and exactly one well-founded model  $X$  for such a logic program  $\Pi$ . This model consists of two parts:

- $X^+$  which is the set of atoms considered *true* and  $X^-$  which is the set of atoms considered *false*.

The well-founded model of  $\Pi$  can be computed incrementally in the following way:

- Step 1: Set  $X^+ = \emptyset$  and  $X^- = \emptyset$ .
- Step 2: Repeat until no changes occur.
  - (a) For every rule  $r$ , if  $\forall b \in \text{body}^+(r), b \in X^+$  and  $\forall c \in \text{body}^-(r), c \in X^-$ , then  $\text{head}(r)$  is added to  $X^+$ .
  - (b) For every atom  $a$ , if  $\forall r \in \Pi$  s.t.  $a = \text{head}(r)$ ,  $\exists b \in \text{body}^+(r), b \in X^-$  and  $\exists c \in \text{body}^-(r), c \in X^+$ , then  $\text{head}(r)$  is added to  $X^-$ .

---

<sup>1</sup>A. Van Gelder, K.A. Ross and J.S. Schlipf. The Well-Founded Semantics for General Logic Programs. Journal of the ACM 38(3), pp. 620—650, 1991

In the beginning, the sets  $X^+$  and  $X^-$  are empty. Then two kinds of deductions can be made: (a) if all the atoms of a rule's body are set then the rule can be applied and the head can be considered true and (b) if an atom cannot be deduced by any rules then it is considered false. These two kind of deductions are applied until a fixpoint is reached.

Consider the following logic program:

$$\Pi = \left\{ \begin{array}{lll} r_1 : & a. & r_2 : \quad c \leftarrow b, \text{not } d. \quad r_3 : \quad b \leftarrow a. \\ r_4 : & d \leftarrow \text{not } c. & r_5 : \quad f \leftarrow b, e. \quad r_6 : \quad g \leftarrow a, b, \text{not } f. \\ & & r_7 : \quad \perp \leftarrow c, d. \end{array} \right\}$$

We detail the result of the application of each step in the next table. The well-founded model for  $\Pi$  is thus  $X^+ = \{a, b, g\}$ ,  $X^- = \{e, f\}$  and  $c, d$  remain unknown:

| Step | $X^+$       | $X^-$       |
|------|-------------|-------------|
| 0    | $\emptyset$ | $\emptyset$ |
| 1    | $a$         | $e$         |
| 2    | $a, b$      | $e, f$      |
| 3    | $a, b, g$   | $e, f$      |
| 4    | $a, b, g$   | $e, f$      |

The input of your solver will be the lpars transformation of a logic program which can be obtained by one of the command lines

```
gringo -g myprog.lp
clingo --pre --lpars -g myprog.lp
```

For example, the above program will have as output from gringo (and as input for your solver):

```
1 2 0 0
1 3 2 1 5 4
1 4 1 0 2
1 5 1 1 3
1 6 2 0 4 7
1 8 3 1 6 2 4
1 1 2 0 3 5
0
8 g
6 f
5 d
4 b
3 c
7 e
2 a
0
B+
0
B-
1
0
1
```

The lparse format divides the output in 5 parts separated by lines containing only '0'. The first part describes the program itself, each line representing a rule. The second part gives the names associated to each number in the program. The third and fourth parts describes the set of atoms already set to true or false. You can notice that '1' is by default set to false: it is to be used as head of constraint rules. description of rules are thus a sequence of numbers: the first one '1' states the rule is normal and can be ignored here as all rules are supposed normal. The second number is the head of the rule. The third is the size of the body and the fourth is the size of the negative body. Then the end of the line contains the atoms in the body starting by the negative body. You can still find a more complete description of this format within the lparse manual.