

Principles of Knowledge Representation and Reasoning

B. Nebel, F. Lindner, T. Engesser
Summer Semester 2018

University of Freiburg
Department of Computer Science

Exercise Sheet 4

Due: May 17th, 2018

Exercise 4.1 (DISJUNCTIVE CONCEPTS IN INHERITANCE NETWORKS, 3)

In this exercise we consider strict inheritance networks that allow for expressing disjunctive concepts of the form C_1 **or** ... **or** C_k . A formula of the form

$$\phi = C_1 \text{ or } \dots \text{ or } C_k \text{ isa } C'_1 \text{ or } \dots \text{ or } C'_l$$

is called a *disjunctive inheritance formula* and its intended interpretation is given by the first-order translation $\pi(\phi) = \forall x((C_1(x) \vee \dots \vee C_k(x)) \rightarrow (C'_1(x) \vee \dots \vee C'_l(x)))$, where the C_i and C'_i are atomic concept terms.

Show that the inheritance problem for disjunctive inheritance networks (given a set of disjunctive inheritance formulae Θ and a disjunctive inheritance formula ϕ , does $\pi(\phi)$ follow from $\{\pi(\vartheta) \mid \vartheta \in \Theta\}$?) can be solved in polynomial time.

Hint: Instead of the translation into first-order logic provide a translation of disjunctive inheritance formulae into propositional logic. You may use the deduction theorem for propositional logic and the fact that the validity of a given set of propositional anti-Horn-formulae can be decided in polynomial time.

Exercise 4.2 (STRICT INHERITANCE WITH NEGATION AND CONJUNCTION, 3)

The inheritance problem for strict inheritance networks with conjunction and negation is coNP-complete. Complete the proof sketch from the lecture.

Exercise 4.3 (INHERITANCE NETWORKS WITH NEGATION, 6)

For this practical assignment, you are asked to implement reasoning in simple inheritance networks that feature negation but no conjunction. Your task is to implement the graph-based approach to decide the inheritance problem for inheritance networks with negation as presented in the lecture.

Input: Your program should be invoked the following way:

```
./inheritance-simple <knowledge base> <query>
```

Here, **<knowledge base>** is the name of a file that contains the knowledge base, and **<query>** is a string¹ that contains an **isa**-statement which is to be evaluated on the knowledge base. Each line in the knowledge base must also be an **isa**-statement. These statements are written as: **concept1 isa concept2**. Both **concept1** and **concept2** are concept terms in either one of the following forms:

- *atomic concepts* are terms consisting of letters, numbers, and hyphens. Not allowed are the keywords **isa** and **not**.
- *negated concepts* are written as terms **not concept**, where **concept** is an atomic concept.

Output: Depending on whether the query follows from the knowledge base the output should be either **Yes.** or **No.** .

Please send your source code to engesser@cs.uni-freiburg.de until **May 17th**.

¹ Usually, when you are working on a shell you have to put the string in quotation marks to protect the spacing.