Lehrstuhl für Automatentheorie
Institut für Theoretische Informatik, TU Dresden
Prof. Dr. F. Baader

Nöthnitzer Str. 46
01187 Dresden
Tel.: 0351/463–39167

6. Exercises for the Course 'Description Logics'

For the following exercises, we will consider *unqualified number restrictions*: concepts of the form $(\geq n\ r)$ and $(\leq n\ r)$, where n is a natural number and r is a role name. The semantics of these constructors is given as follows:

- $(\geq n\ r)^\mathcal{I} = \{x \in \Delta^\mathcal{I} \mid \#\{y \in \Delta^\mathcal{I} \mid (x, y)^\in r^\mathcal{I}\} \geq n\}$,
- $(\leq n\ r)^\mathcal{I} = \{x \in \Delta^\mathcal{I} \mid \#\{y \in \Delta^\mathcal{I} \mid (x, y)^\in r^\mathcal{I}\} \leq n\}$,

where # denotes the cardinality of the set.
The logic extending $\mathcal{ALC}$ with unqualified number restrictions is called $\mathcal{ALCN}$.

Exercise 21:

Express the concept $(\leq 1\ r)$ in first-order logic
Prove that the concept $(\leq 1\ r)$ cannot be expressed in $\mathcal{ALC}$; i. e. prove that there is no $\mathcal{ALC}$ concept C such that the extension of C and the extension of $(\leq 1\ r)$ are the same, for every interpretation $\mathcal{I}$.
Can $(\leq n\ r)$ be expressed in $\mathcal{ALC}$ for any $n \geq 0$?

Exercise 22:

Extend the tableau algorithm given in the lecture such that it can decide ABox consistency in the DL $\mathcal{ALCN}$; that is,

- find new tableau rules and clashes, if necessary,
- extend the proof of local correctness,
- adapt/extend the proofs of soundness and completeness.

Exercise 23:

Prove by induction Lemma 3.8 from the lecture.

Exercise 24:

You have a list of tasks to solve. Each task has an attached priority between 1 (low) and 100 (high). You solve one task after the other following the priority order at every moment. While solving one task, you might find that several subtasks need now to be solved. In this case, you replace the original task by all these new subtasks, each of which has strictly lower priority than the original task.

Use a multiset well-ordering to show that this process terminates eventually.

Exercise 25:

Complete the proof of termination of the tableau algorithm by showing that every rule application leads to a decrease in the well-order (see Lemma 3.10).