TD 9

09/12/2009

Exercise 1. Design polynomial time algorithms that test if the language of a hedge automaton:

1. is finite
2. contains exactly one element

Exercise 2. A top-down hedge automaton has rules of the form $q(f) \to R_{q,f}$ for a state q, function symbol f and regular language $R_{q,f}$ (we consider here w.l.o.g. that $R_{q,f}$ is uniquely determined by q and f). It is deterministic if $R_{q,f}$ contains at most one word of length n for each $n \in \mathbb{N}$.

1. construct a hedge automaton language which is not accepted by any deterministic top-down hedge automaton
2. construct a deterministic top-down hedge automaton language such that the two encodings into ranked languages that you have seen during the course are not accepted by a deterministic top down ranked automaton.