Exercises for FLL, Fall 2018, sheet 10

Return Thursday Nov 22, in class

Exercise 1. At the end of this exercise sheet I append an old exercise from 2004 together with its solution. That old exercise outlines in its problem statement how DFAs can be seen as *S*-structures, and in the model solution shows how they can be framed in FOL axioms. Use the old exercise and its solution as a source of inspiration and do the same for context-free grammars in Chomsky normal form! that is, give a similar outline of such grammars as *S*-structures, and provide FOL axioms. Do not try however to code the requirement that every symbol in a CNF grammar must be useful – that would be quite involved. *Note*: it is *much* more difficult to axiomatise the context-free grammars of arbitrary form – a challenge for the very ambitious ones!

************* *attached: the similar problem from a 2004 exercise sheet* **********

Old exercise from 2004: A DFA can be seen as a structure $\mathcal{D} = (A, S^A, Q^A, F^A, \delta^A, {q_0}^A)$, where the carrier *A* consists of the states and symbols, *S* is a unary predicate (intention: *S* denotes the symbols), *Q* is a unary predicate (denoting the states), *F* is a unary predicate (denoting the accepting states), δ is a binary function (denoting the transition function), and q_0 is a constant symbol (denoting the start state). Give a collection Φ of FOL propositions such that every finite *S*-structure $\mathcal{D}$ is a model of Φ iff $\mathcal{D}$ corresponds to a DFA. In other works, axiomatize the DFAs in FOL. Explain each of your propositons in words.

Solution (to the 2004 homework problem, serves as a hint). Here is one possibility:

$\forall x\ ((Sx \vee Qx) \wedge \neg\ (Sx \wedge Qx))$	every thing must be either a state or a symbol
$(\exists x\ Sx \wedge \exists x\ Qx)$	state and symbol sets are not empty
Qq_0	the start state is actually a state
$\forall x\ (Fx \rightarrow Qx)$	the accepting states are actually states
$\forall x \forall y \forall z\ (((Qx \wedge\ Sy)\ \wedge\ \delta xy = z\ ) \rightarrow Qz)$	δ maps state-symbol pairs on states

Note: because in FOL we only know total functions, in any *S*-structure $\mathcal{D}$ the function δ^A is totally defined. For the purposes of interpreting $\mathcal{D}$ as a DFA, it is not relevant which type of values δ^A has on argument pairs that are not of type (state, symbol).