Frank Richter:

Grammatikformalismen für die Computerlinguistik

Homework Assignment 11 Due: July 13th

Exercise 1. [**Extra Credit: 2 points**]
The implementation of the textbook grammar of Section 3.1.3 is more careful about the specification of the CONTEXT values of the lexical entries than our previous implemented textbook grammars. However, the grammar does not contain anything to enforce reasonable CONTEXT values at phrases.
Assume that the CONTEXT values of phrases are lists containing all and only the elements on the CONTEXT lists of their daughters. Extend the grammar of Section 3.1.3 by code which implements this idea.

The questions in Exercises 2 and 3 are about the *Core Fragment*, which is specified and implemented in TRALE in Section 3.2.1 of *A Web-based Course in Grammar Formalisms and Parsing*. The implementation you need for the new TRALE system we are using in our class is available from `www.sfs.uni-tue-bingen.de/~fr/teaching/ws07-08/lp/grammars.html` with the links under the bullet point 'Chapter 2, Section 3.2.1 Fragment I.'

Exercise 2. [**Extra Credit: 1 point**]
Why does the system answer the query `rec[you,walk]` with `no`?

Exercise 3. [**Extra Credit: 2 points**]
The functional (or non lexical) preposition *to* has a very small lexical entry. In AVM syntax it is simply

$$\begin{bmatrix} \textsc{phon} \left\langle \textit{to} \right\rangle \\ \textsc{syns loc cat head} \begin{bmatrix} \textit{prep} \\ \textsc{pform} \ \textit{to} \end{bmatrix} \end{bmatrix},$$

and its TRALE counterpart is a direct translation of this description into TRALE's notation for lexical entries. If you query TRALE for *to* with `lex to`, however, you get a much more precise description of functional *to* in the denotation of our grammar for an answer (besides a description of lexical *to*, which we ignore here).
Name the parts of the grammar which the compiler used to infer the more precise description of non lexical `to` which it uses at run time for parsing.