Second exercise sheet for the lecture

XML and Programming Languages

Dr. Janis Voigtländer

Summer Term 2008

Exercise 5
Give the unabbreviated versions of the following CoreXPath queries, and describe their semantics relative to a context node n:

1. $..[\mathsf{self}{::}\sigma]$
2. $.//\sigma[\mathsf{ancestor\text{-}or\text{-}self}{::}{*}/\delta]$
3. $.//\sigma/\mathsf{ancestor\text{-}or\text{-}self}{::}{*}$ ◇

Exercise 6
Discuss the semantic equivalence or inequivalence of the path expressions

$$/\mathsf{descendant} :: \mathit{test}[q]$$

and

$$//\mathit{test}[q]$$

for arbitrary *test* and q, using the formal semantics. ◇

Exercise 7
In XPath, it is possible to have several filter expressions in a single navigation step, i.e., "$\mathit{axis}{::}\mathit{test}[q_1]\cdots[q_k]$" instead of just "$\mathit{axis}{::}\mathit{test}[q]$". The semantics is given by successive filtering in left-to-right order.

1. Why is this feature superfluous in CoreXPath, and thus has been omitted from our formalization?
2. What is the situation for full XPath? ◇

Exercise 8
Assume we wanted to allow unions "$\cup$" anywhere in path expressions, rather than just at top-level. What changes to the CoreXPath syntax and the semantic functions would be necessary? Would there be an expressivity gain? $\diamond$

Exercise 9
Consider the query $/\mathsf{descendant}::\sigma/\mathsf{descendant}::\delta$. How does the alternative CoreXPath semantics solve the efficiency problems observed in the lecture for "naive evaluation" on tree documents with string representations $\sigma^n\delta^n\bar{\delta}^n\bar{\sigma}^n$? $\diamond$

Exercise 10
Use the alternative CoreXPath semantics to give a "query tree" for efficient evaluation of $/\mathsf{descendant} :: a/b[c/d \lor \neg(\mathsf{following} :: *)]$. $\diamond$