

**Exercise sheet 6**  
(Due: Monday, 3. March)

As preparation for the exercise, try out the variants of Earley parsers we discussed in class with an example grammar and some sentences of your choice. Carefully look through some tracing output and make sure it corresponds to your understanding of the underlying algorithm.

1. Left-recursive rules are a serious problem for simple top-down parsers. The Earley algorithm has a strong top-down component. Explore with a small example grammar how the Earley algorithm (in the variant introducing preterminals as passive edges, i.e., `parser/earley/preterminals/earley.pl` and `earley_trace.pl`) deals with left-recursion. To do so, provide
  - a small grammar including direct and indirect left recursion and
  - one example run (in the form we saw in class) including use of a directly recursive rule and
  - one example run including use of an indirectly recursive set of rules, and
  - a short discussion (five sentences) of the issue of recursion in these example runs.
  - Will the Earley algorithm always terminate? Give a five sentence argumentation for or against.
2. Read chapter 6 of Gazdar and Mellish (1989)