Compiler Construction (List 4)

Hans de Nivelle

04.11.2015

1. (a) Using the translation scheme on the slides, translate the following regular expression into an NDFA

$$(a|b)^*abbc.$$

   (b) Using the subset construction in the slides, transform the resulting NDFA into a DFA.

   (c) Apply the minimization operation of the slides on the DFA that you obtained in the previous exercise.

2. Do the same (construct NDFA; transform to DFA; minimize) for

$$(\epsilon|aa|bb)^*c.$$

3. The following exercises are based on **flex**: Produce a single flex file that can recognize all of the following tokens. If a token is recognized, the scanner should output something like: `token: (type of token) (attribute of token)`.

   (a) Floating Point numbers (with decimal points and exponents).

   (b) Identifiers.

   (c) A couple of reserved words: **if**, **while**, **do**, **struct**, **class**, **int**, **double**, etc.

   (d) A reasonable set of operators, e.g. `+,-,++,--,*,/,+=,-=`, etc.

   (e) C-style brackets `{,}`.

   (f) Whitespace (tabs, spaces, newlines), comments of form `/* bla bla bla */` and `// bla bla bla`. Whitespace must be ignored. The regular expression for the first type of comment is very tricky.