Formal Languages and Compilers - Exercises II

MSc in Computer Science, University of Camerino
prof. Luca Tesei

Note Regular expressions are written with the usual precedence order: operator * has precedence on concatenation, which has precedence on $|$. Moreover, the usual shorthands $^+$ and ? may be used.

Exercise 1

Write a DFA for the language denoted by the following regexp:

$$a^*b^+c \mid (a|b)^*db^*(c|\epsilon)$$

Justify your answer by showing all the steps that you needed.

Exercise 2

Consider a language of expressions that can be formed using identifiers (**id** tokens) and two binary operators: $\oplus$ and $\otimes$. Give a grammar for this language that embeds the following rules:

- operator $\oplus$ has precedence on $\otimes$;
- operator $\oplus$ is right-associative;
- operator $\otimes$ is left-associative.

Draw a derivation tree for the string $\mathbf{id} \oplus \mathbf{id} \oplus \mathbf{id} \otimes \mathbf{id} \otimes \mathbf{id}$ according to the grammar.

Exercise 3

Consider the following language:

$$\{a^n\, u\, b^{k-1}\, v\, c^m \mid n, k, m > 0 \text{ e } m = n + k\}$$

1. Write a grammar that generates the language
2. Is the language LR? Justify your answer.
3. If the language is LR then give the table of a bottom-up shift-reduce parser for it.

Exercise 4

Consider the following language:

$$\{a^n\, b\, c \mid n > 0\} \cup \{b^n\, c\, b \mid n \geq 0\} \cup \{c\, a^n \mid n > 0\}$$

1. Write a grammar that generates the language.
2. Is the language LL(1)? Justify your answer.
3. If the language is LL(1) then give the table of a top-down predictive parser for it and show the parsing of the strings *cb* and *caa*.