

## Homework 4

1. Assume you have an alphabet consisting of the letters  $a$ ,  $b$  and  $c$  only. (a) Find a regular expression that recognises the two strings  $ab$  and  $ac$ . (b) Find a regular expression that matches all strings *except* these two strings. Note, you can only use regular expressions of the form

$$r ::= \emptyset \mid \epsilon \mid c \mid r_1 + r_2 \mid r_1 \cdot r_2 \mid r^*$$

2. Define the function *zeroable* which takes a regular expression as argument and returns a boolean.<sup>1</sup> The function should satisfy the following property:

$$\text{zeroable}(r) \text{ if and only if } L(r) = \emptyset$$

3. Define the tokens and regular expressions for a language consisting of numbers, left-parenthesis (, right-parenthesis ), identifiers and the operations  $+$ ,  $-$  and  $*$ . Can the following strings in this language be lexed?
  - " $(a + 3) * b$ "
  - " $() ++ - 33$ "
  - " $(a/3) * 3$ "
4. (Optional) The tokenizer in `regex3.scala` takes as argument a string and a list of rules. The result is a list of tokens. Improve this tokenizer so that it filters out all comments and whitespace from the result.
5. (Optional) Modify the tokenizer in `regex2.scala` so that it implements the `findAll` function. This function takes a regular expressions and a string, and returns all substrings in this string that match the regular expression.

---

<sup>1</sup>In an earlier version there was an error.