# Homework 5

1. Consider the basic regular expressions

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

   and suppose you want to show a property $P(r)$ for all regular expressions $r$ by structural induction. Write down which cases do you need to analyse. State clearly the induction hypotheses if applicable in a case.

2. Define a regular expression, written $ALL$, that can match every string. This definition should be in terms of the following extended regular expressions:

$$r ::= \varnothing \mid \epsilon \mid c \mid r_1 + r_2 \mid r_1 \cdot r_2 \mid r^* \mid \;\sim r$$

3. Assume the delimiters for comments are `/*` and `*/`. Give a regular expression that can recognise comments of the form

   `/*` … `*/`

   where the three dots stand for arbitrary characters, but not comment delimiters.

4. Define the following regular expressions

| | |
|---|---|
| $r^+$ | (one or more matches) |
| $r^?$ | (zero or one match) |
| $r^{\{n\}}$ | (exactly $n$ matches) |
| $r^{\{m,n\}}$ | (at least $m$ and maximal $n$ matches, with the assumption $m \leq n$) |

   in terms of the usual basic regular expressions

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

5. Give the regular expressions for lexing a language consisting of identifiers, left-parenthesis `(`, right-parenthesis `)`, numbers that can be either positive or negative, and the operations `+`, `-` and `*`.

   Decide whether the following strings can be lexed in this language?

   (a) `"(a3+3)*b"`

   (b) `")()++-33"`

   (c) `"(b42/3)*3"`

In case they can, give the corresponding token sequences. (Hint: Observe the maximal munch rule and the priorities of your regular expressions that make the process of lexing unambiguous.)

6. (Optional) Recall the definitions for *Der* and *der* from the lectures. Prove by induction on $r$ the property that

$$L(der\, c\, r) = Der\, c\, (L(r))$$

holds.