Homework 5

Wednesday, October 30, 2019 7:30 PM

Q1: Consider the basic regular expressions

$r ::= 0 \mid 1 \mid c \mid r_1 + r_2 \mid r_1 \bullet r_2 \mid r^*$

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 analyze. State clearly the induction hypotheses if applicable in a case.

A1: We first need to prove that P(r) holds for the three base cases:

P(0) is true
P(1) is true
P(c) is true.

Now we can make the Induction hypothesis that P(r) holds and we can start working on the inductive cases:

$P(r_1 + r_2)$ is true, assuming $P(r_1)$ is true and $P(r_2)$ is true.
$P(r_1 \bullet r_2)$ is true, assuming $P(r_1)$ is true and $P(r_2)$ is true.
$P(r^*)$ is true, assuming P® is true.

Q2: Define a regular expression, written ALL, that can match every string.
This definition should be in terms of the following extended regular expressions:

$r ::= 0 \mid 1 \mid c \mid r_1 + r_2 \mid r_1 \bullet r_2 \mid r^* \mid {\sim}r$

A2: $r_{ALL} = 1 + ({\sim}1)^*$

Q3: 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 <= n)

in terms of the usual basic regular expressions

$r ::= 0 \mid 1 \mid c \mid r_1 + r_2 \mid r_1 \bullet r_2 \mid r^*$

A4:

$r^+ \stackrel{\text{def}}{=} r \bullet r^*$
$r^? \stackrel{\text{def}}{=} r + 1$
$r^{\{n\}} \stackrel{\text{def}}{=} r \bullet r \bullet r \bullet \ldots \bullet r$ (n times r)
$r\{m, n\} \stackrel{\text{def}}{=} r \bullet r \bullet r \bullet \ldots \bullet r \bullet (r + 1) \bullet (r + 1) \bullet (r + 1) \bullet \ldots \bullet (r + 1)$ (m times r, n-m times r + 1)

Q4: 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.)

A5:

r_{id} = [a-zA-Z_] • [a-zA-z0-9_]*
$r_{l\text{-}par}$ = (
$r_{r\text{-}par}$ =)
r_{num} = $-^{?}$ • [1-9] • [0-9]*
r_{op} = \+ + - + *

r_{lang} = r_{id} + $r_{l\text{-}par}$ + $r_{r\text{-}par}$ + r_{num} + r_{op})*

(a) "(a3+3)*b" lexes to:
(l-par, (),
(id, a3),
(op, +),
(num, 3),
(r-par,)),
(op, *),
(id, b)
(b) ")()++-33" lexes to
(r-par,)),
(l-par, (),
(r-par,)),
(op, +),
(op, +),
(num, -33)

(c) "(b42/3)*3" doesn't lex, as there's no definition for '/'

Q5: Suppose the following context-free grammar G

S ::= A • S • B | B • S • A | ϵ
A ::= a | ϵ
B ::= b

where the starting symbol is S. Which of the following strings are in the language of G?

A5:

- a --> not in G
- b --> in G
- ab --> in G
- ba --> in G
- bb --> in G
- baa --> not in G

Q6: Suppose the following context-free grammar

S ::= a • S • a | b • S • b | ϵ

Describe which language is generated by this grammar.

A6: The language of all strings composed of a's and b's where all strings are of even length and the right half of the string is the reversed left half of the string and the empty string. (Another description: all palindromes of even length composed of a's and b's)