Master of Science in Computer Science - University of Camerino
Compilers A. Y. 2019/2020
Written Test of 19th February 2020 (Session/Appello II)
Teacher: Luca Tesei

NOTE: Regular expressions should be written using the usual rules of precedence: the $*$ operator has precedence on concatenation, which has precedence on the $|$ operator. The notation $(r)^+$ can be used with the usual meaning.

EXERCISE 1 (10 points)

Consider the following regular expression:

$$a^*(bc^* \mid (bc)^+)$$

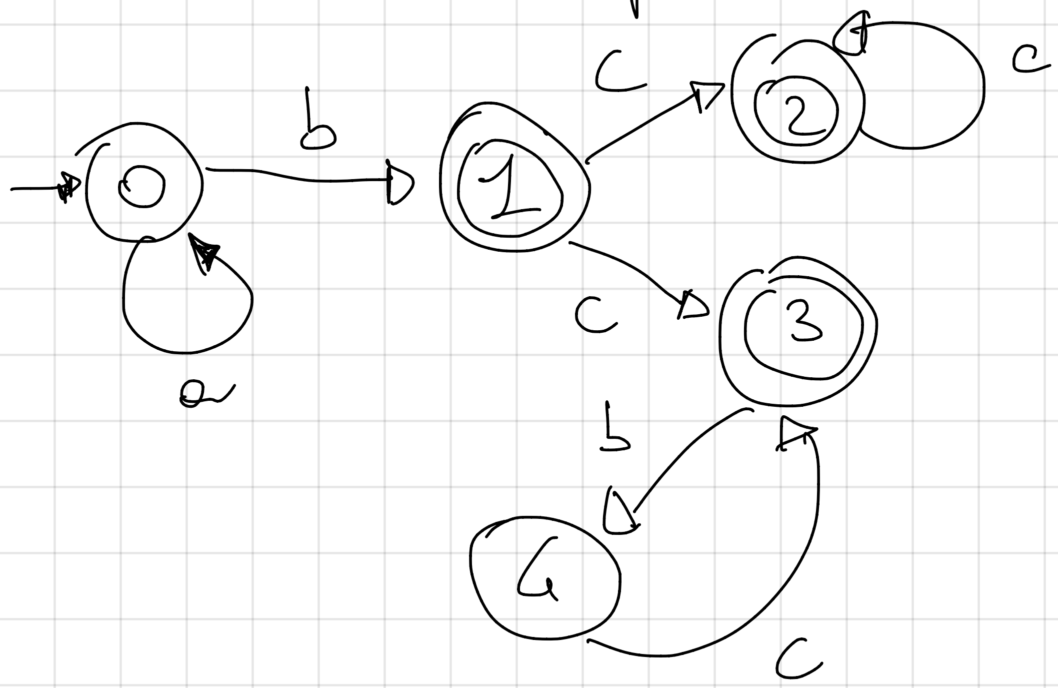
1. Give a minimal automaton accepting the language denoted by the regular expression. Show all the steps leading to your solution.

SOLUTION

The solution is in the following page.

EX1 $a^*(bc^* \mid (bc)^+)$

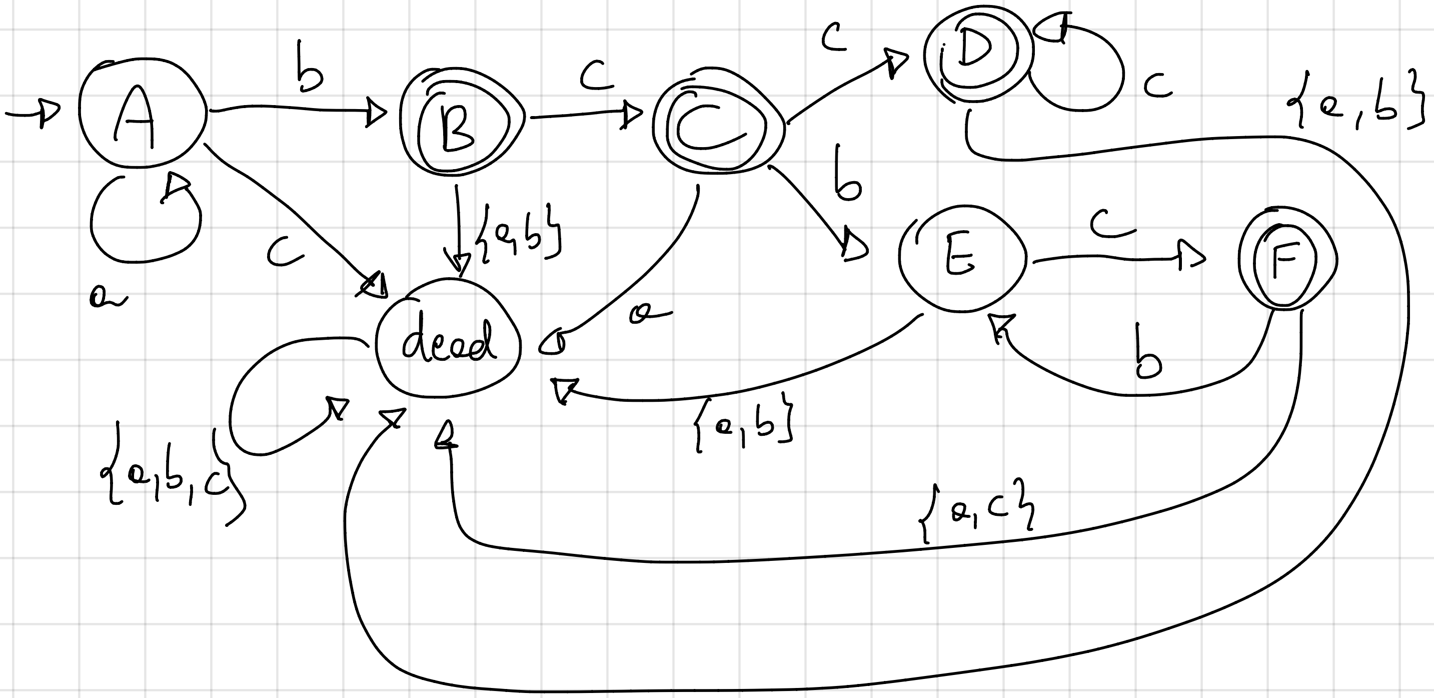
An NFA for the language is



Let's use the subset construction algorithm for obtaining an equivalent DFA

	a	b	c
{0} = A	{0}	{1}	{ }
{1} = B	{ }	{ }	{2, 3}
{2, 3} = C	{ }	{4}	{2}
{2} = D	{ }	{ }	{2}
{4} = E	{ }	{ }	{3}
{3} = F	{ }	{4}	{ }

The obtained DFA, completed with a dead state, is:



Let's try to minimise this DFA.

Partition 1 (A E dead) (B C D F)

$A \xrightarrow{c} dead$
$E \xrightarrow{c} F$
$dead \xrightarrow{c} dead$
} ⇒ A can be differentiated

Partition 2 (E) (A dead) (B C D F)

$A \xrightarrow{b} B$
$dead \xrightarrow{b} dead$
} ⇒ A and dead are different

Partition 3 (E) (A) (dead) (B C D F)

$B \xrightarrow{c} C$
$C \xrightarrow{c} D$
$D \xrightarrow{c} D$
$F \xrightarrow{c} dead$
} F can be differentiated

Partition 4 (E) (A) (dead) (BCD) (F)

$$\left.\begin{array}{l} B \xrightarrow{b} \text{dead} \\ C \xrightarrow{b} E \\ D \xrightarrow{b} \text{dead} \end{array}\right\} \Rightarrow C \text{ can be differentiated}$$

Partition 5 (E) (A) (dead) (BD) (C) (F)

$$\left.\begin{array}{l} B \xrightarrow{c} C \\ D \xrightarrow{c} D \end{array}\right\} \text{ B and D are different}$$

Partition 6 (E) (A) (dead) (B) (D) (C) (F)

The DFA is minimal because no states can be considered equivalent.

EXERCISE 2 (12 points)

Consider the following grammar:

$$\begin{array}{lcl} S & \rightarrow & B \mid Caa \\ B & \rightarrow & bC \\ C & \rightarrow & bbCa \mid \epsilon \end{array}$$

1. Write formally the language generated by the grammar as a set of strings.
2. Is the grammar LR(1)? If so, give the table of a bottom-up shift-reduce parser and show the parsing of the string $bbba$.

SOLUTION

The language can be expressed as follows:

$$L(S) = \{b^{2n+1}\, a^n \mid n \geq 0\} \cup \{b^{2n}\, a^{n+2} \mid n \geq 0\}$$

Let us start checking if the grammar is SLR(1). If so, it is also LR(1). Let us compute the set of LR(0) items:

$I_0 = \begin{array}{lcl} S' & \rightarrow & \cdot S \\ S & \rightarrow & \cdot B \\ S & \rightarrow & \cdot Caa \\ B & \rightarrow & \cdot bC \\ C & \rightarrow & \cdot bbCa \\ C & \rightarrow & \cdot \end{array}$	$I_1 = \texttt{goto}(I_0, S) = S' \rightarrow S\cdot$
$I_2 = \texttt{goto}(I_0, B) = S \rightarrow B\cdot$ $I_3 = \texttt{goto}(I_0, C) = S \rightarrow C \cdot aa$	$I_4 = \texttt{goto}(I_0, b) = \begin{array}{lcl} B & \rightarrow & b \cdot C \\ C & \rightarrow & b \cdot bCa \\ C & \rightarrow & \cdot bbCa \\ C & \rightarrow & \cdot \end{array}$
$I_5 = \texttt{goto}(I_3, a) = S \rightarrow Ca \cdot a$ $I_6 = \texttt{goto}(I_4, C) = B \rightarrow bC\cdot$	$I_7 = \texttt{goto}(I_4, b) = \begin{array}{lcl} C & \rightarrow & bb \cdot Ca \\ C & \rightarrow & b \cdot bCa \\ C & \rightarrow & \cdot bbCa \\ C & \rightarrow & \cdot \end{array}$
$I_8 = \texttt{goto}(I_5, a) = S \rightarrow Caa\cdot$ $I_9 = \texttt{goto}(I_7, C) = C \rightarrow bbC \cdot a$	$\texttt{goto}(I_7, b) = I_7$ $I_{10} = \texttt{goto}(I_9, a) = C \rightarrow bbCa\cdot$

It holds that FOLLOW(S') = FOLLOW(S) = FOLLOW(B) = $\{\$\}$ and FOLLOW(C) = $\{a, \$\}$. There are no conflicts in the set of LR(0) items, meaning that the grammar is SLR(1) and LR(1). The parsing table for the corresponding bottom-up shift-reduce parser is as follows:

	a	b	$\$$	S	B	C
0	r5	s4	r5	1	2	3
1			acc			
2			r1			
3	s5					
4	r5	s7	r5			6
5	s8					
6			r3			
7	r5	s7	r5			9
8			r2			
9	s10					
10	r4		r4			

The parsing of the string $bbba$ is in the following:

STACK	INPUT	ACTION
$\$0$	$bbba\$$	shift 4
$\$0b4$	$bba\$$	shift 7
$\$0b4b7$	$ba\$$	shift 7
$\$0b4b7b7$	$a\$$	reduce 5
$\$0b4b7b7C9$	$a\$$	shift 10
$\$0b4b7b7C9a10$	$\$$	reduce 4
$\$0b4C6$	$\$$	reduce 3
$\$0B2$	$\$$	reduce 1
$\$0S1$	$\$$	accept

EXERCISE 3 (12 points)

Consider a language of expressions defined recursively as follows:

(i) x is an expression;

(ii) if $e_1, e_2, \ldots, e_n$ (with $n > 0$) are expressions then $f(e_1, \ldots, e_n)$ is an expression.

Your tasks are:

1. Define a Syntax Directed Translation Scheme suitable to be implemented by a top-down parser and such that it computes, for the starting symbol, an attribute $\mathbf{m}$ of type `int`. For a give expression, $\mathbf{m}$ must give the maximum number of arguments to which the function f is applied to. The maximum must be computed considering any possible subexpression, not only the top level f. Examples:
   - for the expression x it must result $\mathbf{m} = 0$,
   - for the expression $f(x)$ it must result $\mathbf{m} = 1$,
   - for the expression $f(f(x))$ it must result $\mathbf{m} = 1$,
   - for the expression $f(x, f(x, x))$ it must result $\mathbf{m} = 2$,
   - for the expression $f(f(x, x))$ it must result $\mathbf{m} = 2$,
   - for the expression $f(x, f(x, x), f(x, f(x)))$ it must result $\mathbf{m} = 3$,
   - for the expression $f(f(x, x), f(x, f(x), f(f(f(f(x))))))$ it must result $\mathbf{m} = 3$.

SOLUTION

The solution is in the following page.

EX 3 | Let us give first an $LL(1)$ grammar for the Language:

$\Sigma = \{x, f, (,), \text{";"}\}$

$E \to x \mid f(L)$

$L \to EA$

$A \to , L \mid \varepsilon$

$FIRST(E) = \{x, f\} = FIRST(L)$

$FIRST(A) = \{\text{";"}, \varepsilon\}$

$FOLLOW(E) = \{\$, \text{";"},)\}$

$FOLLOW(L) = \{)\}$

$FOLLOW(A) = \{)\}$

The parsing table is

	x	f	(	)	,	\$
E	$E \to x$	$E \to f(L)$				
L	$L \to EA$	$L \to EA$				
A				$A \to \varepsilon$	$A \to , L$	

The table is not multiply defined, so the grammar is $LL(1)$.

The SDT is as follow: (for attributes see next page)

$E \to x \quad \{E.m = 0\}$

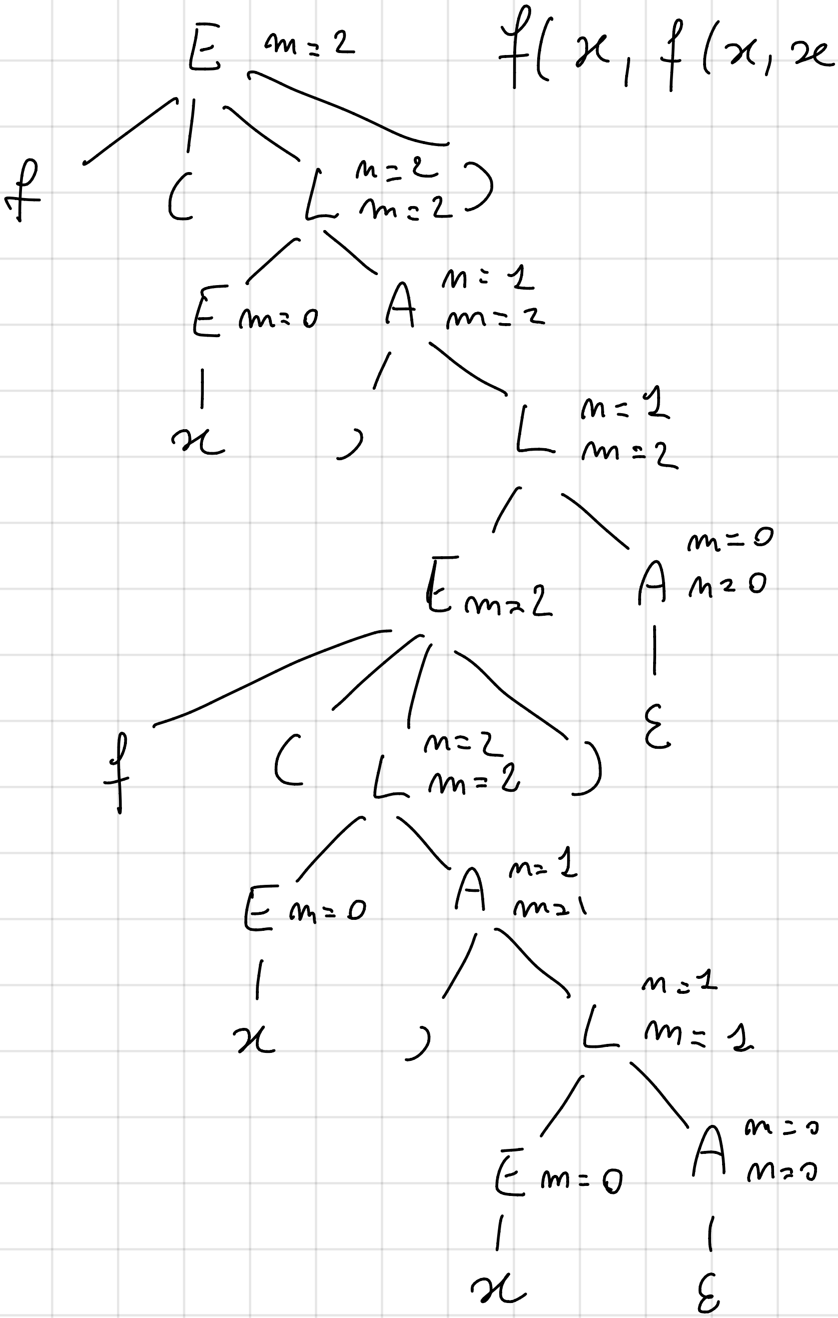
$E \to f(L) \quad \{E.m = L.m\}$

$L \to EA \quad \{L.n = A.n + 1,\ L.m = \max(E.m, L.m)\}$

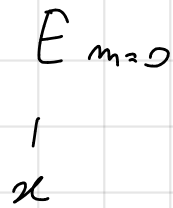
$A \to , L \quad \{A.n = L.n,\ A.m = L.m\}$

$A \to \varepsilon \quad \{A.n = 0,\ A.m = 0\}$

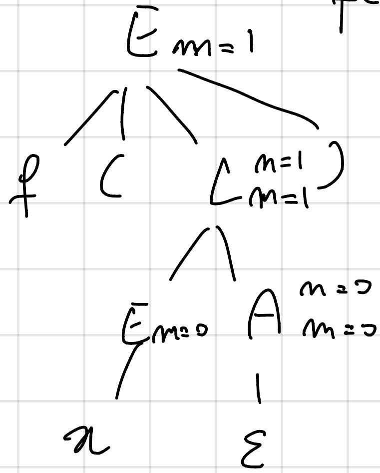
$f(x, f(x, x))$



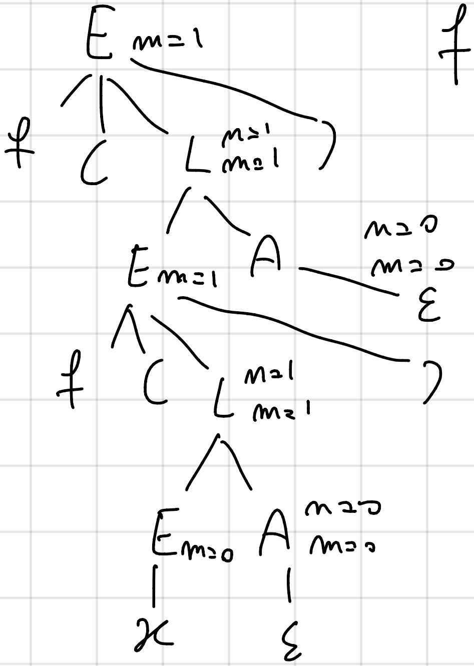
x



$f(x)$



$f(f(x))$



Attributes of symbols

m of type int
synthesized
for symbols E, L, A
m is the maximum number of argument as requested

m of type int
synthesized
for symbols L, A
m is the length of the current list

$f(f(x, x))$