

CSEN 1003 Compiler, Spring Term 2020
Practice Assignment 5

Discussion: 1.03.20 - 04.03.20

Exercise 5-1

Consider the following CFG:

$$\begin{aligned} S &\rightarrow 0T1S \mid \varepsilon \\ T &\rightarrow 0T1 \mid \varepsilon \end{aligned}$$

- a) Compute the FIRST and FOLLOW sets.

Solution:

$$\text{FIRST}(S) = \{0, \varepsilon\}$$

$$\text{FIRST}(T) = \{0, \varepsilon\}$$

$$\text{FOLLOW}(S) = \{\$ \}$$

$$\text{FOLLOW}(T) = \{1\}$$

- b) Compute the parsing table.

Solution:

Non-terminal	Input Symbol		
	0	1	\$
S	$0T1S$		ε
T	$0T1$	ε	

- c) Prove that this grammar is LL(1).

Solution:

The grammar is LL(1) because every entry in parsing table contains at most one rule.

- d) What language does this grammar recognize?

Solution:

$$\{0^n 1^n \mid n \geq 1\}^*$$

Exercise 5-2

Consider the following CFG:

$$\begin{aligned} S &\rightarrow SAB \mid SBC \mid \varepsilon \\ A &\rightarrow aAa \mid \varepsilon \\ B &\rightarrow bB \mid \varepsilon \\ C &\rightarrow cC \mid \varepsilon \end{aligned}$$

- a) Compute FIRST and FOLLOW sets for each non-terminal.

Solution:

Non-terminal	FIRST	FOLLOW
S	$\{a, b, c, \varepsilon\}$	$\{\$, a, b, c\}$
A	$\{a, \varepsilon\}$	$\{\$, a, b, c\}$
B	$\{b, \varepsilon\}$	$\{\$, a, b, c\}$
C	$\{c, \varepsilon\}$	$\{\$, a, b, c\}$

- b) Build the parsing table.

Solution:

Non-terminal	Input Symbol			
	a	b	c	\$
S	SAB SBC ε	SAB SBC ε	SAB SBC ε	SAB SBC ε
A	aAa ε	ε	ε	ε
B	ε	bB ε	ε	ε
C	ε	ε	cC ε	ε

- c) From the parsing table, show why the grammar is not LL(1).

Solution:

The grammar is not LL(1) because some entries of the parsing table contain more than one rule.

Exercise 5-3

Construct a parsing table for the following grammar. Is the grammar LL(1)?

$S \rightarrow AB$
 $A \rightarrow \text{id } A \mid \text{num}$
 $B \rightarrow CA$
 $C \rightarrow 0C \mid 1$

Solution:

Non-terminal	Input Symbol				
	id	num	0	1	\$
S	AB	AB			
A	$\text{id } A$	num			
B			CA	CA	
C			$0C$	1	

The grammar is LL(1) because every position in the parsing table contains at most one rule.

Exercise 5-4

Consider the following CFG:

$$\begin{aligned} S &\rightarrow (L) \mid \mathbf{a} \\ L &\rightarrow L, S \mid S \end{aligned}$$

- a) Eliminate left recursion.

Solution:

$$\begin{aligned} S &\rightarrow (L) \mid \mathbf{a} \\ L &\rightarrow (L)L' \mid \mathbf{a}L' \\ L' &\rightarrow ,SL' \mid \varepsilon \end{aligned}$$

- b) Compute FIRST and FOLLOW sets for each non-terminal.

Solution:

$$\begin{aligned} \text{FIRST}(S) &= \{ (, \mathbf{a} \} \\ \text{FIRST}(L) &= \{ (, \mathbf{a} \} \\ \text{FIRST}(L') &= \{ , , \varepsilon \} \\ \text{FOLLOW}(S) &= \{), , , \$ \} \\ \text{FOLLOW}(L) &= \{) \} \\ \text{FOLLOW}(L') &= \{) \} \end{aligned}$$

- c) Build the parsing table.

Solution:

Non-terminal	Input Symbol				
	\mathbf{a}	,	()	\$
S	\mathbf{a}		(L)		
L	$\mathbf{a}L'$		$(L)L'$		
L'		$,SL'$		ε	

- d) Trace the operations of a predictive top-down parser on the string $(\mathbf{a}, (\mathbf{a}, \mathbf{a}))$.

Exercise 5-5

Consider the following CFG:

$$S \rightarrow SS+ \mid SS* \mid \mathbf{a}$$

- a) Eliminate left recursion and left factor the grammar.

Solution:

$$\begin{aligned} S &\rightarrow \mathbf{a}S' \\ S' &\rightarrow SS'' \mid \varepsilon \\ S'' &\rightarrow +S' \mid *S' \end{aligned}$$

- 4

e) Trace the operations of a predictive top-down parser on the string **aa+**.