Pulkit Shringi
· 596
II

Compiler Design

Q1) Check whether the given grammar is LL(1)

$S \rightarrow iEtS \mid iEtSeS$

$E \rightarrow b$

Q2) Construct predictive parsing table to the given grammar

$S \rightarrow +SS \mid *SS \mid a$ and parse the string $+*aaa$

Q3) Construct LR(1) items for the grammar.

$S \rightarrow AA$

$A \rightarrow Aa \mid b$

Q4) Construct the following grammar

$S \rightarrow L=R$

$S \rightarrow R$

$L \rightarrow *R$

$L \rightarrow id$

$R \rightarrow L$

Q5) Consider the grammar

$A \rightarrow (A) \mid a$

a) Find LR(0) items

b) Construct SLR(1) Parsing table

① $S \to iEtS \mid iEtSeS$

$E \to f$

No left recursion

Left factoring removal : $S \to iEtSS'$

$S' \to eS \mid \varepsilon$

$E \to \varepsilon$

First :

$F(S) = i \quad ; F(S') = e\varepsilon, \varepsilon \mid F(E) = b$

Follow :

$F_o(S) = \{\$, e\} \quad ; F_o(S') = \{\$, e\}$

$E = (E) = \{t\}$

Variable	i	t	e	b	$
S	$S \to iEtSS'$		$S \to eS$		
S'			$S' \to \varepsilon$		$S' \to S$
E				$E \to b$	

② $S \to +SS \mid *SS \mid a$

No left recursion, No left factoring

First $(S) = \{+, *, a\}$

$F_o(S) =$ First (S)

$= \{\$, +, *, a\}$

Recursive Parsing table :-

N.T	+	*	a	$
S	$S \to aSS$	$S \to SS$	$S \to a$	

Parsing String: + * aaa $

String	Input	Action
S *	+ * aaa $	
+ SS $	+ $ aaa $	S → + SS (Match)
SS $	* aaa $	S → * SS
* SS $ $	aaa $	Match
SSS $	aaa $	S → a
a SS $	aa $	Match
SS $	aa $	S → a
a S $	a a $	Match
S $	a $	S → a
a $	$ $	Match
$	$	Match

String → Accepted

(5) A → (A) | a

S ⟹ & A → (A)
A → a

Terminals = {(,), a}
Non terminals = {A}

Argumented Grammar

A' → A
A → (A)
A → a

F(A) = {(, a}
Fo(A) = {$,)}

Closure:-

$$\begin{bmatrix} A' \to .A \\ A \to .(A) \\ A \to .a \end{bmatrix} I_0$$

goto (I_0, A) : $\boxed{A' \to A\cdot}$ I_1

goto $(I_0, ($: $\boxed{\begin{matrix} A \to (\cdot A) \\ A \to \cdot a \end{matrix}}$ I_2

goto (I_0, a) : $\boxed{A \to a\cdot}$ I_3

goto (I_2, A) : $\boxed{A \to (A\cdot)}$ I_4

goto (I_2, A) : $\boxed{A \to a\cdot}$ Same I_3

goto $(I_4,))$: $\boxed{A \to (A)\cdot}$ I_5

State	Action				goto
	(	)	a	$	A
0	s_2		s_3		1
1				accept	
2			s_3		4
3		r_2		r_2	
4		s_5			
5		r_3		r_1	

6) $S \to AA \mid A \to Aa \mid b$

Numbers the productions 1. $S \to AA$

2. $S \to Aa$

3. $A \to b$

Argumented grammar

$S' \to S$

$S \to AA$

$A \to Aa \quad A \to b$

$F(S) = First(A)$

$F(A) = \{b\}$

$= \{a, b\}$

Closure

$S' \to S, \$$

$S \to .AA, \$$

$A \to .Aa, b$

$A \to .b$] I_0

Goto $(I_0, S) = \boxed{S' \to S. \$}$ I_1

Goto $(I_0, A) \Rightarrow S \to A.A, \$$

$A \to .Aa, \$$

$A \to .b, \$$] I_2

Goto $(I_0, b) \to \boxed{A \to b., b}$ I_3

Goto $(I_2, A) \to S \to AA., \$$

$A \to A.a, \$$] I_4

Goto $(I_2, b) \to \boxed{A \to b., \$}$ I_5

Goto $(I_4, a) \Rightarrow \boxed{A \to Aa., \$}$ I_6

LALR Table

State	Action			Goto	
	a	b	$	S	A
0		S_2		1	2
1			accept		
2		S_5			4
35		S_3	r_3		
4	S_6		r_1		
6			r_2		

combining similar parts : $I_{35} : [A \to b \cdot, b/\$]$

③ $S \to AA \mid A \to Aa \mid b$

Number the productions

1 $S \to AA$

2 $S \to Aa$

3 $S \to b$

Argumented Grammar:

$S' \to S$

$S \to AA$

$A \to Aa$

$A \to b$

Introduction of look ahead & closure:

$S' \to \cdot S, \$$

$S \to \cdot AA, \$$

$A \to \cdot Aa, b$

$A \to \cdot b, b$ — I_0

Goto $(I_0, S) \rightarrow [S' \rightarrow S \cdot, \$]$ I_1

Goto (I_0, A) : $\begin{bmatrix} S' \rightarrow A \cdot A, \$ \\ A \rightarrow \cdot Aa, \$ \\ A \rightarrow \cdot b, \$ \end{bmatrix}$ I_2

Goto $(I_0, b) = A \rightarrow b \cdot, b]$ I_3

Goto (I_2, A) : $\begin{bmatrix} S \rightarrow AA \cdot \$ \\ A \rightarrow A \cdot a, \$ \end{bmatrix}$ I_4

Goto (I_3, b) : $[A \rightarrow b \cdot, b]$ I_5

Goto (I_4, a) : $A \rightarrow Aa \cdot, \$]$ I_6

State	Action			Goto	
	a	b	$	S	A
0		S_3		1	2,3
1			accept		
2		S_5			4
3		γ_3			
4	S_6		γ_1		
5			γ_3		
6			γ_2		

(4) $S \rightarrow t = R \mid S \rightarrow R \mid L \rightarrow *R \mid L \rightarrow id \mid R \rightarrow$

$Fo(S) = \{\$\}$:

Number of productions : 1. $S \to L = R$
2. $S \to R$
3. $L \to *R$
4. $L \to id$
5. $R \to L$

Augumented grammar :

$S' \to S$
$S \to L = R$
$S \to R$
$L \to *R$
$L \to id$
$R \to L$

Introduction of dot & closure

$S' \to .S$
$S \to .L.R$
$S \to .R$
$L \to .*R$
$L \to .id$
$R \to .L$
] I_0

goto (I_0, S) : $S' \to S.$] I_1

goto (I_0, L) : $S \to L.=R$
$R \to L.$] I_2

goto (I_0, R) : $S \to R.$] I_3

goto $(I_0, *)$: $L \to *.R$] I_4
goto (I_0, id) $R \to .L$

$L \to id.$ I_5

goto (I_2, =): $S \to L = \cdot R$] I_6
$R \to \cdot * R$ I_7

goto (I_6, R): $S \to L = R \cdot$] I_8

$Fo(S) = \{\$\}$
$FO[L] = \{=, \$\}$
$Fo(R) = \{\$, =\}$

State	Action				Goto		
	=	*	id	$	S	L	R
0		S_3	S_5		1	2	3
1				accept			
2				r_5			
3	S_5, r_3			r_2			
4		S_4	S_5			8	7
5		S_4	S_5				
6	r_3			r_3			
7	r_5			r_5			
8				r_1			

Note: SLR, ~~or~~ there ~~are~~ ~~two~~ (shift, reduce) in a single cell.