1. $E \longrightarrow E_{\text{1 or 8}} +_6 T_9$
2. $E \longrightarrow T_2$
3. $T \longrightarrow T_{\text{2 or 9}} *_7 F_{10}$
4. $T \longrightarrow F_3$
5. $F \longrightarrow (_4 E_8)_{11}$
6. $F \longrightarrow \mathrm{id}_5$

	id	+	*	(	)	eof	E	T	F
0									
1									
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									

ACTION GOTO

Figure 1: Grammar G_1 and an LALR Parser for that Grammar

The action of an LALR parser for grammar G_1 on the string `(x+y*z)*x`

```
0:(x+y*z)*x$
s4
0(4:x+y*z)*x$
s5
0(4x5:+y*z)*x$
r6, goto 3
0(4F3:+y*z)*x$ F->x
r4, goto 2
0(4T2:+y*z)*x$ T->F
r2, goto 8
0(4E8:+y*z)*x$ E->T
s6
0(4E8+6:y*z)*x$
s5
0(4E8+6y5:*z)*x$
r6, goto 3
0(4E8+6F3:*z)*x$ F->y
r4, goto 9
0(4E8+6T9:*z)*x$ T->F
s7
0(4E8+6T9*7:z)*x$
s5
0(4E8+6T9*7z5:)*x$
r6, goto 10
0(4E8+6T9*7F10:)*x$
r3, goto 9
0(4E8+6T9:)*x$ T->T*F
r1, goto 8
0(4E8:)*x$ E->E+T
s11
0(4E8)11:*x$
r5, goto 3
0F3:*x$ F->(E)
r4, goto 2
0T2:*x$ T->F
s7
0T2*7:x$
s5
0T2*7x5:$
r6, goto 10
0T2*7F10:$
r3, goto 2
0T2:$
r2, goto 1
0E1:$
halt
```

Let G_2 be the following ambiguous grammar, which illustrates the "dangling else" problem. The Parser table has a shift-reduce conflict, which we resolve using the "closest if" rule. Informally, the alphabet symbols have the following meanings:

- **S** statement
- **L** list of statements
- **a** statement with no control structure
- **w** while condition
- **i** if condition
- **e** else
- **b** begin
- **n** end

1. $S \longrightarrow a_2$
2. $S \longrightarrow w_3\, S_4$
3. $S \longrightarrow i_5\, S_6$
4. $S \longrightarrow i_5\, S_6\; e_7\; S_8$
5. $S \longrightarrow b_9\, L_{10}\, n_{11}$
6. $L \longrightarrow L_{10}\, S_{12}$
7. $L \longrightarrow$ epsilon

	ACTION							GOTO	
	a	w	i	e	b	n	eof	S	L
0									
1							halt		
2									
3									
4									
5									
6									
7									
8									
9									
10									
11									
12									

Figure 2: G_2 and its LALR Parser Table, to Fill in

We can avoid the conflict by choosing an unambiguous context-free grammar equivalent to G_2, namely G_3 below.

1. $S \longrightarrow M_2$
2. $S \longrightarrow U_3$
3. $M \longrightarrow a_4$
4. $M \longrightarrow w_5\, M_6$
5. $U \longrightarrow w_5\, U_7$
6. $U \longrightarrow i_8\, S_9$
7. $M \longrightarrow i_8\, M_2\, e_{10}\, M_{11}$
8. $U \longrightarrow i_8\, M_2\, e_{10}\, U_{12}$
9. $M \longrightarrow b_{13}\, L_{14}\, n_{15}$
10. $L \longrightarrow L_{14}\, S_{16}$
11. $L \longrightarrow$ epsilon

Informally, the alphabet symbols have the following meanings:

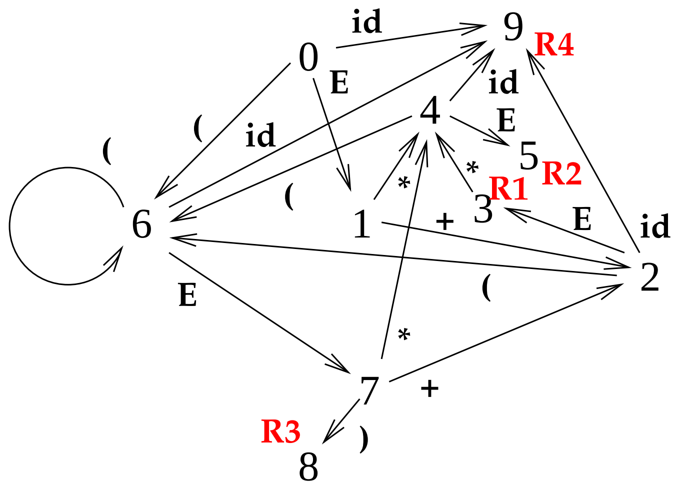
S	statement
M	matched statement
U	unmatched statement
L	list of statements
a	statement with no control structure
w	while condition
i	if condition
e	else
b	begin
n	end

	a	w	i	e	b	n	eof	S	M	U	L
0	s4	s5	s8		s13			1	2	3	
1							halt				
2	r1	r1	r1	s10	r1	r1	r1				
3	r2	r2	r2		r2	r2	r2				
4	r3	r3	r3	r3	r3	r3	r3				
5	s4	s5	s8		s13				6	7	
6	r4	r4	r4	r4	r4	r4	r4				
7	r5	r5	r5		r5	r5	r5				
8	s4	s5	s8		s13			9	2	3	
9	r6	r6	r6		r6	r6	r6				
10		s5	s8		s13				11	12	
11	r7	r7	r7	r7	r7	r7	r7				
12	r8	r8	r8		r8	r8	r8				
13	r11	r11	r11		r11	r11					14
14	s4	s5	s8		s13	s15		16	2	3	
15	r9	r9	r9	r9	r9	r9	r9				
16	r10	r10	r10		r10	r10					

ACTION GOTO

Figure 3: LALR Parser Table for the Unambiguous Grammar G_3

1. $E \longrightarrow E_{1 \text{ or } 7} +_{2} E_{3}$
2. $E \longrightarrow E_{1 \text{ or } 3 \text{ or } 7} *_{4} E_{5}$
3. $E \longrightarrow (_{6} E_{7})_{8}$
4. $E \longrightarrow id_{9}$



	id	+	*	(	)	eof	E
0							
1							
2							
3							
4							
5							
6							
7							
8							
9							

ACTION GOTO

Figure 4: LALR Parser Table for an Ambiguous Grammar Equivalent to G_1