1. The *ACTION* function takes as arguments a state *i* and a terminal *a* (**$**, the input endmarker). The value of *ACTION***[***i***,***a***]** can have one of four forms:
   (a) Shift *j* where *j* is a state. The action taken by the parser effectively shifts input *a* to the stack, but uses state *j* to represent *a*.
   (b) Reduce $A \rightarrow \beta$. The action of the parser effectively reduces β on the top of the stack to head A.
   (c) Accept. The parser accepts the input and finishes parsing.
   (d) Error. The parser discovers an error in its input and takes some corrective action.
2. We extend the GOTO function, defined on sets of items, to states: if GOTO[I_i,A]=I_j, then *GOTO* also maps a state *i* and a nonterminal *A* to state *j.*

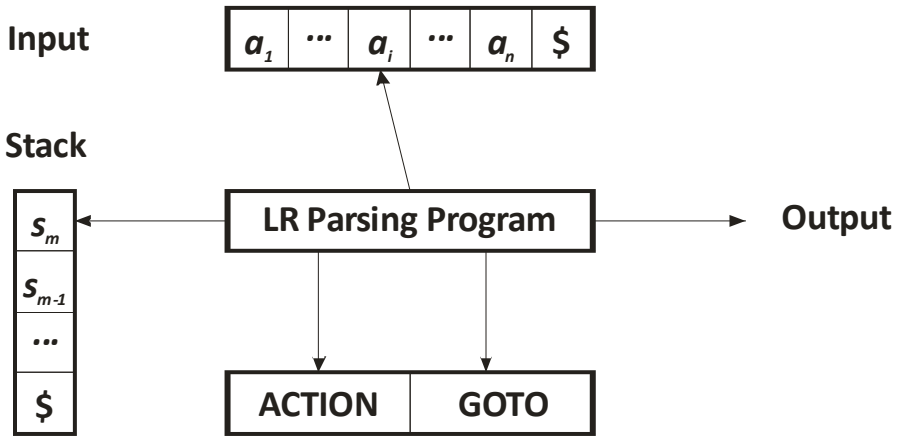

Figure 1. LR Parser Model

	left side		*right side*
1	*E*	→	*E* **+** *T*
2	*E*	→	*T*
3	*T*	→	*T* ***** *F*
4	*T*	→	*F*
5	*F*	→	**(** *E* **)**
6	*F*	→	**id**

Table 1. Set of productions expressions

let *a* be the first symbol of *w*$
while (1) {
    let *s* be the state on top of the stack;
    if (*ACTION***[***s***,***a***]==*shift t*) {**
        push *t* onto the stack
        let *a* be the next input symbol;
    } else if (*ACTION***[***s***,***a***]==*reduce* $A \rightarrow \beta$ **){**
        pop $|\beta|$symbols off the stack;
        let state *t* now be on top of the stack;
        push *GOTO***[***t***,***A***]** onto the stack
        output the production $A \rightarrow \beta$;
    } else if (*ACTION***[***s***,***a***]==*accept*)** break; //parsing is done
    else *error***();**
}

	STACK	SYMBOLS	INPUT	ALGORITHM	ACTION
(1)	0		**id * id + id $**	ACTION[0,id]=s5	shift
(2)	0 5	**id**	*** id + id $**	ACTION[5,*]=r6 GOTO[0,*F*]=3	reduce by $F \rightarrow$ **id**
(3)	0 3	*F*	*** id + id $**	ACTION[3,*]=r4 GOTO[0,*T*]=2	reduce by $T \rightarrow F$
(4)	0 2	*T*	*** id + id $**	ACTION[2,*]=s7	shift
(5)	0 2 7	*T* *****	**id + id $**	ACTION[7,id]=s5	shift
(6)	0 2 7 5	*T* *** id**	**+ id $**	ACTION[5,+]=r6 GOTO[7,F]=10	reduce by $F \rightarrow$ **id**
(7)	0 2 7 10	*T* ***** *F*	**+ id $**	ACTION[10,+]=r3 GOTO[0,T]=2	reduce by $T \rightarrow T$ ***** F
(8)	0 2	*T*	**+ id $**	ACTION[2,+]=r2 GOTO[0,E]=1	reduce by $E \rightarrow T$
(9)	0 1	*E*	**+ id $**	ACTION[1,+]=s6	shift
(10)	0 1 6	*E* **+**	**id $**	ACTION[6,id]=s5	shift
(11)	0 1 6 5	*E* **+ id**	**$**	ACT ION[5,$]=r6 GOTO[6,*F*]=3	reduce by $F \rightarrow$ **id**
(12)	0 1 6 3	*E* **+** *F*	**$**	ACTION[3,$]=r4 GOTO[6,*T*]=9	reduce by $T \rightarrow F$
(13)	0 1 6 9	*E* **+** *T*	**$**	ACTION[9,$]=r1 GOTO[0,*E*]=1	reduce by $E \rightarrow E$ **+** T
(14)	0 1	*E*	**$**	ACTION[1,$]=acc	accept

STATE	ACTION						GOTO		
	id	**+**	*****	**(**	**)**	**$**	*E*	*T*	*F*
0	s5			s4			1	2	3
1		s6				acc			
2		r2	s7		r2	r2			
3		r4	r4		r4	r4			
4	s5			s4			8	2	3
5		r6	r6		r6	r6			
6	s5			s4				9	3
7	s5			s4					10
8		s6			s11				
9		r1	s7		r1	r1			
10		r3	r3		r3	r3			
11		r5	r5		r5	r5			

STATE	ACTION						GOTO		
	id	**+**	*****	**(**	**)**	**$**	*E*	*T*	*F*
0	s5			s4			1	2	3
1		s6				acc			
2		r2	s7		r2	r2			
3		r4	r4		r4	r4			
4	s5			s4			8	2	3
5		r6	r6		r6	r6			
6	s5			s4				9	3
7	s5			s4					10
8		s6			s11				
9		r1	s7		r1	r1			
10		r3	r3		r3	r3			
11		r5	r5		r5	r5			

	STACK	SYMBOLS	INPUT	ALGORITHM	ACTION
(1)	0		**id + id * id $**	ACTION[0,id]=s5	shift
(2)	0 5	**id**	**+ id * id $**	ACTION[5,+]=r6 GOTO[0,*F*]=3	reduce by $F \rightarrow$ **id**
(3)	0 3	*F*	**+ id * id $**	ACTION[3,+]=r4 GOTO[0,*T*]=2	reduce by $T \rightarrow F$
(4)	0 2	*T*	**+ id * id $**	ACTION[2,+]=r2 GOTO[0,*E*]=1	reduce by $E \rightarrow T$
(5)	0 1	*E*	**+ id * id $**	ACTION[1,+]=s6	shift
(6)	0 1 6	*E* **+**	**id * id $**	ACTION[6,id]=s5	shift
(7)	0 1 6 5	*E* **+ id**	*** id $**	ACTION[5,*]=r6 GOTO[6,*F*]=3	reduce by $F \rightarrow$ **id**
(8)	0 1 6 3	*E* **+** *F*	*** id $**	ACTION[3,*]=r4 GOTO[6,*T*]=9	reduce by $T \rightarrow F$
(9)	0 1 6 9	*E* **+** *T*	*** id $**	ACTION[9,*]=s7	shift
(10)	0 1 6 9 7	*E* **+** *T* *****	**id $**	ACTION[7,id]=s5	shift
(11)	0 1 6 9 7 5	*E* **+** *T* *** id**	**$**	ACTION[5,$]=r6 GOTO[7,*F*]=10	reduce by $F \rightarrow$ **id**
(12)	0 1 6 9 7 10	*E* **+** *T* ***** *F*	**$**	ACTION[10,$]=r3 GOTO[6,*T*]=9	reduce by $T \rightarrow T *$ *F*
(13)	0 1 6 9	*E* **+** *T*	**$**	ACTION[9,$]=r1 GOTO[0,*E*]=1	reduce by $E \rightarrow E +$ *T*
(14)	0 1	*E*	**$**	ACTION[1,$]=acc	accept

$(s_0 s_1 \cdots s_m, a_i a_{i+1} \cdots a_n \$)$

$X_1 X_2 \cdots X_m a_i a_{i+1} \cdots a_n$