

LR(1) grammar (' ' is ϵ):

```
(0) S' -> begin STMT_LIST end
(1) STMT_LIST -> STMT
(2) STMT -> DEC
(3) STMT -> IF_STMT
(4) STMT -> WHILE_LOOP
(5) STMT -> ASSIGN
(6) DEC -> data id
(7) ASSIGN -> id = EXPR
(8) WHILE_LOOP -> while (BOOL_EXPR) STMT
(9) IF_STMT -> if (BOOL_EXPR) STMT else STMT
(10) EXPR -> TERM + TERM
(11) EXPR -> TERM * TERM
(12) TERM -> FACTOR - FACTOR
(13) TERM -> FACTOR / FACTOR
(14) TERM -> FACTOR % FACTOR
(15) FACTOR -> id
(16) FACTOR -> int_lit
(17) FACTOR -> (EXPR)
(18) BOOL_EXPR -> BREL ! BREL
(19) BOOL_EXPR -> BREL ~ BREL
(20) BREL -> BEXPR # BEXPR
(21) BREL -> BEXPR $ BEXPR
(22) BREL -> BEXPR < BEXPR
(23) BREL -> BEXPR > BEXPR
(24) BEXPR -> BTERM + BTERM
    BEXPR -> BTERM * BTERM
    BTERM -> BFACTOR - BFACTOR
    BTERM -> BFACTOR / BFACTOR
    BTERM -> BFACTOR % BFACTOR
    BFACTOR -> id
    BFACTOR -> int_lit
    BFACTOR -> (BEXPR)
    BFACTOR -> bool_lit
```

>>

FIRST table	
Nonterminal	FIRST
S'	{begin}
STMT_LIST	{data,if,while,id}
STMT	{data,if,while,id}
DEC	{data}
ASSIGN	{id}
WHILE_LOOP	{while}
IF_STMT	{if}
EXPR	{id,int_lit,(EXPR)}
TERM	{id,int_lit,(EXPR)}
FACTOR	{id,int_lit,(EXPR)}
BOOL_EXPR	{id,int_lit,(BEXPR),bool_lit}
BREL	{id,int_lit,(BEXPR),bool_lit}
BEXPR	{id,int_lit,(BEXPR),bool_lit}
BTERM	{id,int_lit,(BEXPR),bool_lit}
BFACTOR	{id,int_lit,(BEXPR),bool_lit}

Goto	
	{{[S' -> .begin STMT_LIST end, \$]}
goto(0, begin)	{{[S' -> begin.STMT_LIST end, \$]}
goto(1, STMT_LIST)	{{[S' -> begin STMT_LIST.end, \$]}
goto(1, STMT)	{{[STMT_LIST -> STMT., end]}
goto(1, DEC)	{{[STMT -> DEC., end]}
goto(1, IF_STMT)	{{[STMT -> IF_STMT., end]}
goto(1, WHILE_LOOP)	{{[STMT -> WHILE_LOOP., end]}
goto(1, ASSIGN)	{{[STMT -> ASSIGN., end]}
goto(1, data)	{{[DEC -> data.id, end]}
goto(1, if)	{{[IF_STMT -> if.(BOOL_EXPR) STMT
goto(1, while)	{{[WHILE_LOOP -> while.(BOOL_EXPR
goto(1, id)	{{[ASSIGN -> id.= EXPR, end]}
goto(2, end)	{{[S' -> begin STMT_LIST end., \$]}
goto(8, id)	{{[DEC -> data id., end]}
goto(9, (BOOL_EXPR))	{{[IF_STMT -> if (BOOL_EXPR).STMT
goto(10, (BOOL_EXPR))	{{[WHILE_LOOP -> while (BOOL_EXPR
goto(11, =)	{{[ASSIGN -> id =.EXPR, end]}
goto(14, STMT)	{{[IF_STMT -> if (BOOL_EXPR) STMT
goto(14, DEC)	{{[STMT -> DEC., else]}
goto(14, IF_STMT)	{{[STMT -> IF_STMT., else]}
goto(14, WHILE_LOOP)	{{[STMT -> WHILE_LOOP., else]}
goto(14, ASSIGN)	{{[STMT -> ASSIGN., else]}
goto(14, data)	{{[DEC -> data.id, else]}
goto(14, if)	{{[IF_STMT -> if.(BOOL_EXPR) STMT
goto(14, while)	{{[WHILE_LOOP -> while.(BOOL_EXPR
goto(14, id)	{{[ASSIGN -> id.= EXPR, else]}
goto(15, STMT)	{{[WHILE_LOOP -> while (BOOL_EXPR
goto(15, DEC)	{{[STMT -> DEC., end]}
goto(15, IF_STMT)	{{[STMT -> IF_STMT., end]}
goto(15, WHILE_LOOP)	{{[STMT -> WHILE_LOOP., end]}
goto(15, ASSIGN)	{{[STMT -> ASSIGN., end]}
goto(15, data)	{{[DEC -> data.id, end]}
goto(15, if)	{{[IF_STMT -> if.(BOOL_EXPR) STMT
goto(15, while)	{{[WHILE_LOOP -> while.(BOOL_EXPR
goto(15, id)	{{[ASSIGN -> id.= EXPR, end]}
goto(16, EXPR)	{{[ASSIGN -> id = EXPR., end]}
goto(16, TERM)	{{[EXPR -> TERM.+ TERM, end]; [EXI
goto(16, FACTOR)	{{[TERM -> FACTOR.- FACTOR, +/*];
goto(16, id)	{{[FACTOR -> id., -//%]}
goto(16, int_lit)	{{[FACTOR -> int_lit., -//%]}
goto(16, (EXPR))	{{[FACTOR -> (EXPR)., -//%]}
goto(17, else)	{{[IF_STMT -> if (BOOL_EXPR) STMT
goto(22, id)	{{[DEC -> data id., else]}
goto(23, (BOOL_EXPR))	{{[IF_STMT -> if (BOOL_EXPR).STMT
goto(24, (BOOL_EXPR))	{{[WHILE_LOOP -> while (BOOL_EXPR
goto(25, =)	{{[ASSIGN -> id =.EXPR, else]}
goto(28, +)	{{[EXPR -> TERM +.TERM, end]}
goto(28, *)	{{[EXPR -> TERM *.TERM, end]}
goto(29, -)	{{[TERM -> FACTOR -.FACTOR, +/*]}
goto(29, /)	{{[TERM -> FACTOR /.FACTOR, +/*]}
goto(29, %)	{{[TERM -> FACTOR %.FACTOR, +/*]}
goto(33, STMT)	{{[IF_STMT -> if (BOOL_EXPR) STMT
goto(33, DEC)	{{[STMT -> DEC., end]}
goto(33, IF_STMT)	{{[STMT -> IF_STMT., end]}
goto(33, WHILE_LOOP)	{{[STMT -> WHILE_LOOP., end]}

goto(33, WHILE_LOOP)	{[STMT -> WHILE_LOOP., end]}
goto(33, ASSIGN)	{[STMT -> ASSIGN., end]}
goto(33, data)	{[DEC -> data.id, end]}
goto(33, if)	{[IF_STMT -> if.(BOOL_EXPR) STMT]
goto(33, while)	{[WHILE_LOOP -> while.(BOOL_EXPR)
goto(33, id)	{[ASSIGN -> id.= EXPR, end]}
goto(35, STMT)	{[IF_STMT -> if (BOOL_EXPR) STMT]
goto(35, DEC)	{[STMT -> DEC., else]}
goto(35, IF_STMT)	{[STMT -> IF_STMT., else]}
goto(35, WHILE_LOOP)	{[STMT -> WHILE_LOOP., else]}
goto(35, ASSIGN)	{[STMT -> ASSIGN., else]}
goto(35, data)	{[DEC -> data.id, else]}
goto(35, if)	{[IF_STMT -> if.(BOOL_EXPR) STMT]
goto(35, while)	{[WHILE_LOOP -> while.(BOOL_EXPR)
goto(35, id)	{[ASSIGN -> id.= EXPR, else]}
goto(36, STMT)	{[WHILE_LOOP -> while (BOOL_EXPR)
goto(36, DEC)	{[STMT -> DEC., else]}
goto(36, IF_STMT)	{[STMT -> IF_STMT., else]}
goto(36, WHILE_LOOP)	{[STMT -> WHILE_LOOP., else]}
goto(36, ASSIGN)	{[STMT -> ASSIGN., else]}
goto(36, data)	{[DEC -> data.id, else]}
goto(36, if)	{[IF_STMT -> if.(BOOL_EXPR) STMT]
goto(36, while)	{[WHILE_LOOP -> while.(BOOL_EXPR)
goto(36, id)	{[ASSIGN -> id.= EXPR, else]}
goto(37, EXPR)	{[ASSIGN -> id = EXPR., else]}
goto(37, TERM)	{[EXPR -> TERM.+ TERM, else]; [E.
goto(37, FACTOR)	{[TERM -> FACTOR.- FACTOR, +/*];
goto(37, id)	{[FACTOR -> id., -//%]}
goto(37, int_lit)	{[FACTOR -> int_lit., -//%]}
goto(37, (EXPR))	{[FACTOR -> (EXPR)., -//%]}
goto(38, TERM)	{[EXPR -> TERM + TERM., end]}
goto(38, FACTOR)	{[TERM -> FACTOR.- FACTOR, end];
goto(38, id)	{[FACTOR -> id., -//%]}
goto(38, int_lit)	{[FACTOR -> int_lit., -//%]}
goto(38, (EXPR))	{[FACTOR -> (EXPR)., -//%]}
goto(39, TERM)	{[EXPR -> TERM * TERM., end]}
goto(39, FACTOR)	{[TERM -> FACTOR.- FACTOR, end];
goto(39, id)	{[FACTOR -> id., -//%]}
goto(39, int_lit)	{[FACTOR -> int_lit., -//%]}
goto(39, (EXPR))	{[FACTOR -> (EXPR)., -//%]}
goto(40, FACTOR)	{[TERM -> FACTOR - FACTOR., +/*].
goto(40, id)	{[FACTOR -> id., +/*]}
goto(40, int_lit)	{[FACTOR -> int_lit., +/*]}
goto(40, (EXPR))	{[FACTOR -> (EXPR)., +/*]}
goto(41, FACTOR)	{[TERM -> FACTOR / FACTOR., +/*].
goto(41, id)	{[FACTOR -> id., +/*]}
goto(41, int_lit)	{[FACTOR -> int_lit., +/*]}
goto(41, (EXPR))	{[FACTOR -> (EXPR)., +/*]}
goto(42, FACTOR)	{[TERM -> FACTOR % FACTOR., +/*].
goto(42, id)	{[FACTOR -> id., +/*]}
goto(42, int_lit)	{[FACTOR -> int_lit., +/*]}
goto(42, (EXPR))	{[FACTOR -> (EXPR)., +/*]}
goto(44, else)	{[IF_STMT -> if (BOOL_EXPR) STMT]
goto(47, +)	{[EXPR -> TERM +.TERM, else]}
goto(47, *)	{[EXPR -> TERM *.TERM, else]}
goto(49, -)	{[TERM -> FACTOR -.FACTOR, end]}
goto(49, /)	{[TERM -> FACTOR /.FACTOR, end]}
goto(49, %)	{[TERM -> FACTOR %.FACTOR, end]}

goto(57, STMT)	{[IF_STMT -> if (BOOL_EXPR) STMT
goto(57, DEC)	{[STMT -> DEC., else]}
goto(57, IF_STMT)	{[STMT -> IF_STMT., else]}
goto(57, WHILE_LOOP)	{[STMT -> WHILE_LOOP., else]}
goto(57, ASSIGN)	{[STMT -> ASSIGN., else]}
goto(57, data)	{[DEC -> data.id, else]}
goto(57, if)	{[IF_STMT -> if.(BOOL_EXPR) STMT
goto(57, while)	{[WHILE_LOOP -> while.(BOOL_EXPR
goto(57, id)	{[ASSIGN -> id.= EXPR, else]}
goto(58, TERM)	{[EXPR -> TERM + TERM., else]}
goto(58, FACTOR)	{[TERM -> FACTOR.- FACTOR, else]}
goto(58, id)	{[FACTOR -> id., -///%]}
goto(58, int_lit)	{[FACTOR -> int_lit., -///%]}
goto(58, (EXPR))	{[FACTOR -> (EXPR)., -///%]}
goto(59, TERM)	{[EXPR -> TERM * TERM., else]}
goto(59, FACTOR)	{[TERM -> FACTOR.- FACTOR, else]}
goto(59, id)	{[FACTOR -> id., -///%]}
goto(59, int_lit)	{[FACTOR -> int_lit., -///%]}
goto(59, (EXPR))	{[FACTOR -> (EXPR)., -///%]}
goto(60, FACTOR)	{[TERM -> FACTOR - FACTOR., end]}
goto(60, id)	{[FACTOR -> id., end]}
goto(60, int_lit)	{[FACTOR -> int_lit., end]}
goto(60, (EXPR))	{[FACTOR -> (EXPR)., end]}
goto(61, FACTOR)	{[TERM -> FACTOR / FACTOR., end]}
goto(61, id)	{[FACTOR -> id., end]}
goto(61, int_lit)	{[FACTOR -> int_lit., end]}
goto(61, (EXPR))	{[FACTOR -> (EXPR)., end]}
goto(62, FACTOR)	{[TERM -> FACTOR % FACTOR., end]}
goto(62, id)	{[FACTOR -> id., end]}
goto(62, int_lit)	{[FACTOR -> int_lit., end]}
goto(62, (EXPR))	{[FACTOR -> (EXPR)., end]}
goto(65, -)	{[TERM -> FACTOR -.FACTOR, else]}
goto(65, /)	{[TERM -> FACTOR /.FACTOR, else]}
goto(65, %)	{[TERM -> FACTOR %.FACTOR, else]}
goto(73, FACTOR)	{[TERM -> FACTOR - FACTOR., else]}
goto(73, id)	{[FACTOR -> id., else]}
goto(73, int_lit)	{[FACTOR -> int_lit., else]}
goto(73, (EXPR))	{[FACTOR -> (EXPR)., else]}
goto(74, FACTOR)	{[TERM -> FACTOR / FACTOR., else]}
goto(74, id)	{[FACTOR -> id., else]}
goto(74, int_lit)	{[FACTOR -> int_lit., else]}
goto(74, (EXPR))	{[FACTOR -> (EXPR)., else]}
goto(75, FACTOR)	{[TERM -> FACTOR % FACTOR., else]}
goto(75, id)	{[FACTOR -> id., else]}
goto(75, int_lit)	{[FACTOR -> int_lit., else]}
goto(75, (EXPR))	{[FACTOR -> (EXPR)., else]}

State	begin	end	data	id	=	while	(BOOL_EXPR)	if	else
0	s1								
1			s8	s11		s10		s9	
2		s12							
3		r1							
4		r2							

5		r ₃							
6		r ₄							
7		r ₅							
8			s13						
9						s14			
10						s15			
11				s16					
12									
13		r ₆							
14			s22	s25	s24			s23	
15			s8	s11	s10			s9	
16				s30					
17								s33	
18								r ₂	
19								r ₃	
20								r ₄	
21								r ₅	
22				s34					
23						s35			
24						s36			
25				s37					
26		r ₈							
27		r ₇							
28									
29									
30									
31									
32									
33			s8	s11	s10			s9	
34								r ₆	
35			s22	s25	s24			s23	
36			s22	s25	s24			s23	
37				s30					
38				s30					
39				s30					
40				s52					
41				s52					
42				s52					
43		r ₉							
44								s57	
45								r ₈	
46								r ₇	
47									
48		r ₁₀							
49									
50		r ₁₁							
51									1
52									1
53									1
54									1
55									1
56									1
57			s22	s25	s24			s23	
58				s30					

59				s ₃₀					
60				s ₆₈					
61				s ₆₈					
62				s ₆₈					
63									r ₉
64									r ₁₀
65									
66									r ₁₁
67		r ₁₂							
68		r ₁₅							
69		r ₁₆							
70		r ₁₇							
71		r ₁₃							
72		r ₁₄							
73				s ₇₇					
74				s ₇₇					
75				s ₇₇					
76									r ₁₂
77									r ₁₅
78									r ₁₆
79									r ₁₇
80									r ₁₃
81									r ₁₄