LR(0) Automaton

Que 1. Consider the CFG fragment with non-terminal symbols {D, T, L}, with start symbol D, terminal symbols { id, :, ;, ,, var, integer, real } and the productions P listed below.

D → var L : T ; $
T → integer
T → real
L → L , id
L → id

Construct the LR(0) DFA for the above grammar. Clearly mention what are the items in each state. Based on the LR(0) DFA, provide the LR(0) parse table for the grammar. Briefly describe the process followed to build the automaton and the parse table. Is the grammar LR(0)?

Start Symbol : D
Non-Terminals : {D, T, L}
Terminals : {id, :, ;, ,, var, integer, real}
Creating an augmented grammar by adding a new production Rule: D' -> D.
where . is the dot marker used.

Augmented grammar:

1. D' -> D.
2. D → var L : T ; $
3. T → integer
4. T → real
5. L → L , id
6. L → id

LR(0) DFA:

State 0: D' -> .D D -> .var L : T ;	State 1 (0, D): D' -> D.	State 2 (0, var): D -> var .L : T ; L -> .L , id L -> .id	State 3 (2, L): D -> var L .: T ; L -> L. , id
State 4 (2, id): L -> id.	State 5 (3, :): D -> var L : .T ; T -> .integer T -> .real	State 6 (3, ,): L -> L , .id	State 7 (5, T): D -> var L : T. ;
State 8 (5, integer): T -> integer.	State 9 (5, real): T -> real.	State 10 (6, id): L -> L , id.	State 11 (7, ;): D -> var L : T ;.

PARSING TABLE:

	ACTION							GOTO		
	id	:	;	,	var	integer	real	D	T	L
0					S2			1		
1	Acc	Acc	Acc	Acc	Acc	Acc	Acc			
2	S4									3
3		S5		S6						
4	R6	R6	R6	R6	R6	R6	R6			
5						S8	S9		7	
6	S10									
7			S11							
8	R3	R3	R3	R3	R3	R3	R3			
9	R4	R4	R4	R4	R4	R4	R4			
10	R5	R5	R5	R5	R5	R5	R5			
11	R2	R2	R2	R2	R2	R2	R2			

Sakshi Shashikant Yeole(20CS01047)