Nonterminal	eol	eof	bcomment	identifier	eq	while	do	end	if	then	else	def	lbracket	rbracket	comma	integer	float	string	nil	\$
PROG	PROG -> eol PROG	PROG -> eof	PROG => bcomment PROG	PROG -> SEC END_ROW PROG		PROG => SEC END_ROW PROG		PROG ->	PROG => SEC END_ROW PROG		PROG ->	PROG -> SEC END_ROW PROG								PROG ->
END_ROW	END_ROW -> eol	END_ROW -> eof																		
SEC				SEC -> identifier eq FCE_EXPR		SEC -> while EXPR do eol PROG end			SEC -> if EXPR then eol PROG else eol PROG end			SEC -> def identifier lbracket PARAMS rbracket eol PROG end								
PARAMS				PARAMS -> identifier PARAM_LIST										PARAMS ->						
PARAM_LIST														PARAM_LIS T -> ''	PARAM_LIST -> comma identifier PARAM_LIST					
CALL_PARAMS	CALL_PARAM S => ''	CALL_PARAM S -> ''		CALL_PARAMS -> ITEM CALL_PARAM_LIST										CALL_PARA MS -> ''		CALL_PARAMS -> ITEM CALL_PARAM_LIST	CALL_PARAMS -> ITEM CALL_PARAM_LIST	CALL_PARAMS -> ITEM CALL_PARAM_LIST	CALL_PARAMS -> ITEM CALL_PARAM_LIST	
CALL_PARAM_LI ST	CALL_PARAM _LIST =>	CALL_PARAM _LIST ->												CALL_PARA M_LIST ->	CALL_PARAM_LIST => COMMMA ITEM CALL_PARAM_LIST					
FCE_EXPR	FCE_EXPR = > EXPR	FCE_EXPR = > EXPR		FCE_EXPR -> FCE																
BR_OR_NOT	BR_OR_NOT -> CALL_PARAM S	BR_OR_NOT -> CALL_PARAM S		BR_OR_NOT -> CALL_PARAMS									BR_OR_NOT -> lbracket CALL_PARAMS rbracket			BR_OR_NOT -> CALL_PARAMS	BR_OR_NOT -> CALL_PARAMS	BR_OR_NOT -> CALL_PARAMS	BR_OR_NOT -> CALL_PARAMS	
FCE				FCE -> identifier BR_OR_NOT																
ITEM				ITEM -> identifier								•				ITEM -> integer	ITEM -> float	ITEM -> string	ITEM -> nil	
EXPR	EXPR -> ''	EXPR -> ''					EXPR ->			EXPR ->										