SYNTAX DIRECTED TRANSLATION
SEMANTIC ANALYSIS

- Syntax directed translation is translating a string of correct syntax into a sequence of actions by attaching one such action to each rule of a grammar.
- SDT provides a simple way to attach semantics or meanings to any syntax.
- Adding actions in the productions in a CFG results in Syntax-Directed Definitions - SDDs.
- Each symbol in the grammar can have an attribute, which is a value, that will be associated with that symbol. Attributes e.g. type, value, ..
- An SDT can be implemented by first building a parse tree by performing left to right depth-first order, a preordal traversal.
- The translation can be done without building a parse tree in some cases.
- There is a class of SDTs. called "L-attributed" and "S-attributed"
- L-attributed, L - left to right, LL-parsable it encompasses, all translations that can be done during parsing.
- S-attributed, S - synthesized, LR-parsable. easy in connection with bottom-up parse

Syntax Directed Definitions - SDD.

- A CFG with attributes & rules
- attributes are associated with grammar symbols & rules are associated with productions.
- Let X is a symbol (a non-terminal) & ~~X~~val is one of its attributes, then X.val denotes the value of 'val' at particular parse tree node

- There are two kind of attributes for nonterminals
① Synthesized attribute: This attribute is defined in terms of it's children attributes computed result in value of itself.
② Inherited attribute: the value will be defined in its parent production. Its value will be determined by its parents or its siblings or by itself.

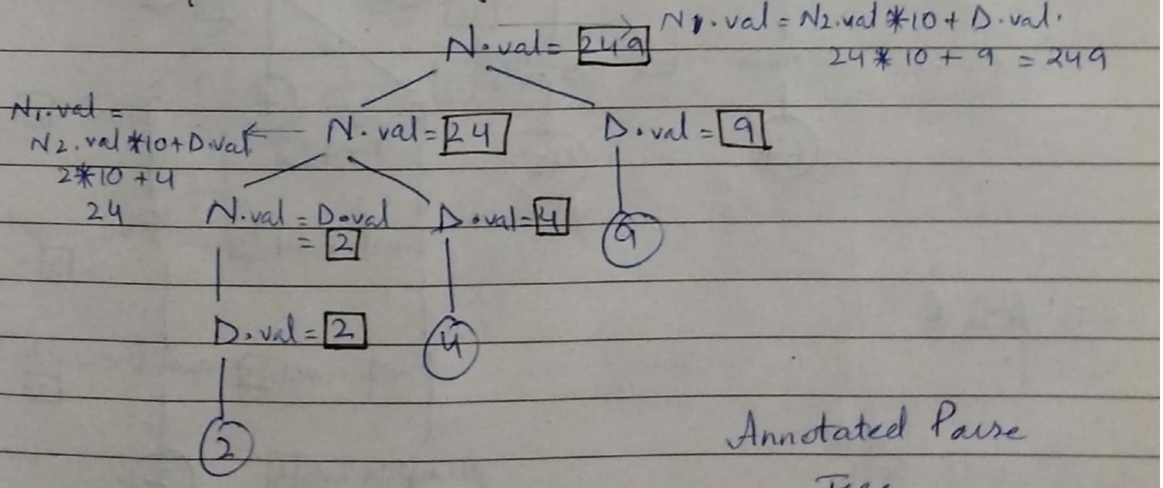
Parse Tree ⟶ Semantic Analyzer ⟶ Annotated Parse Tree.

Example # 1.

Grammar Rule	Semantic rules.
① $number_1 \rightarrow number_2$ digit.	$number_1.val = number_2.val * 10 +$ digit.val
② number ⟶ digit.	number. val = digit.val.
③ digit ⟶ 0|1|2|3...|9.	digit.val = 0, 1, 2, 3, ..., 9.

① First make a parse tree
② Then apply semantic rules.

Make a parse tree for 249.

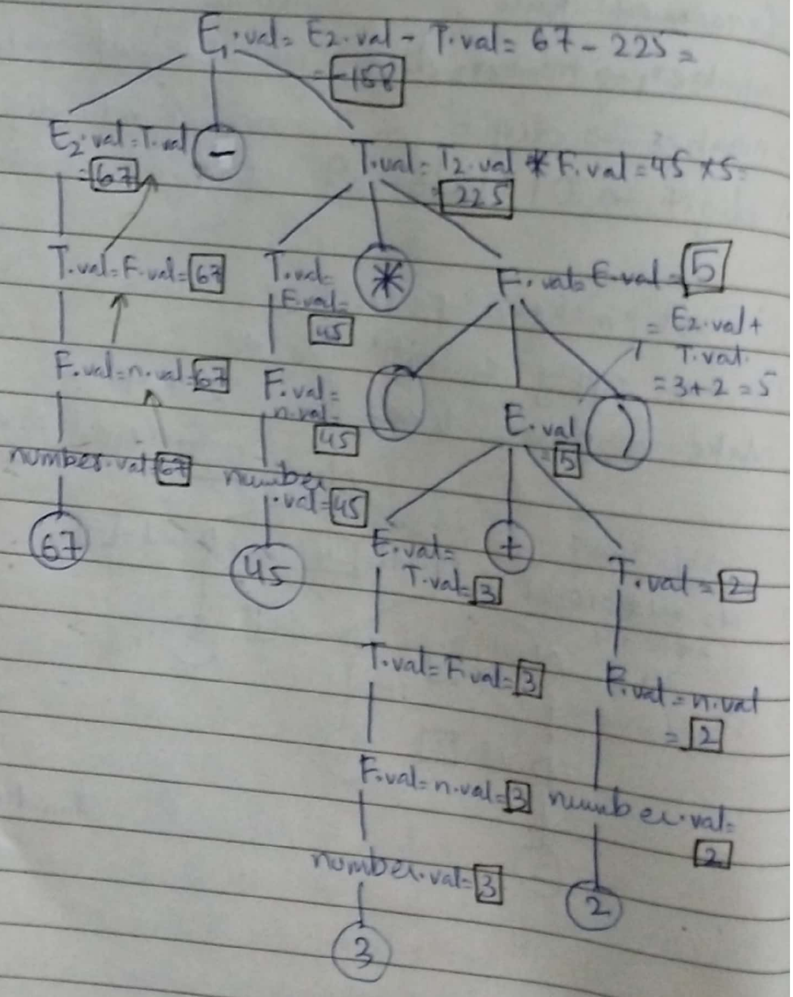


Annotated Parse Tree.

Example # 2.

Grammar rule	Semantic Rules.
① $E_1 \to E_2 + T$	$E_1.val = E_2.val + T.val$
② $E_1 \to E_2 - T$	$E_1.val = E_2.val - T.val$
③ $E \to T$	$E_1.val = T.val$
④ $T_1 \to T_2 * F$	$T_1.val = T_2.val * F.val$
⑤ $T \to F$	$T.val = F.val$
⑥ $F \to (E)$	$F.val = E.val$
⑦ $F \to$ number.	$F.val =$ number.val.

$67 - 45 * (3 + 2)$



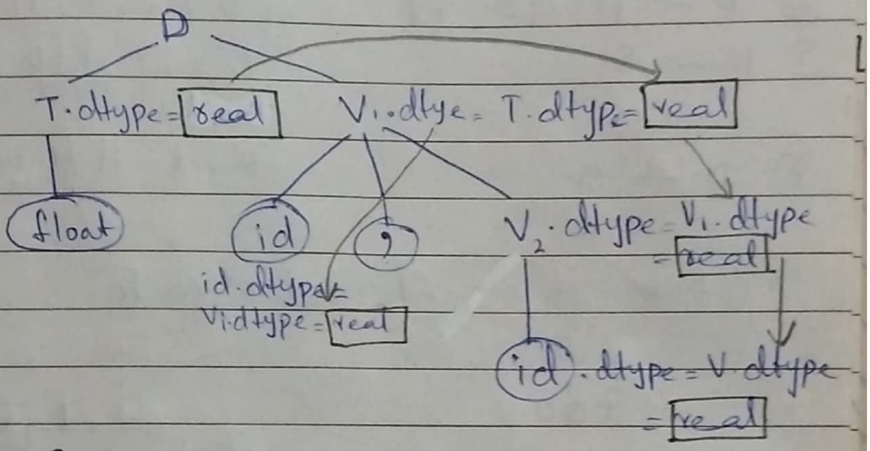
AST

Example # 3.

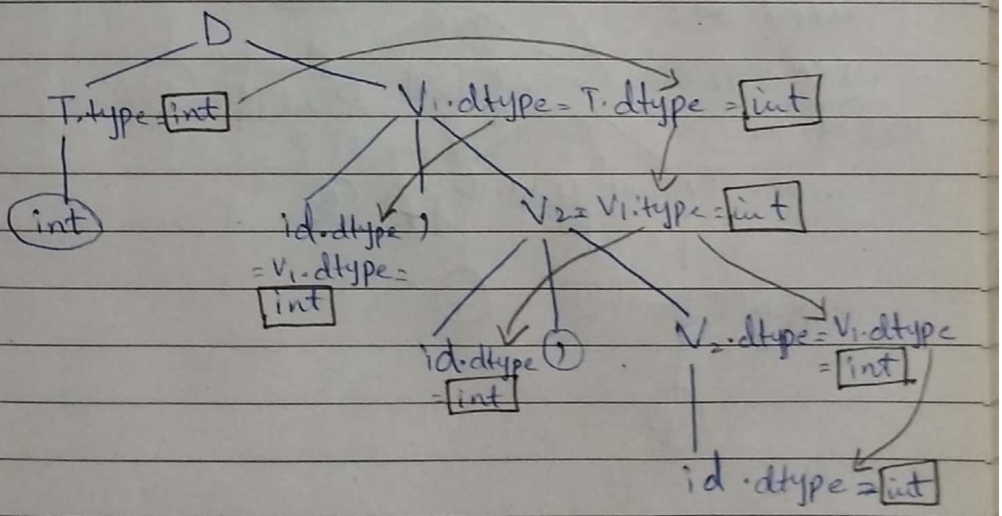
	Semantic rules.
(1) $D \rightarrow T\ V$	$V.dtype = T.dtype.$
(2) $T \rightarrow int$	$T.dtype = integer.$
(3) $T \rightarrow float$	$T.dtype = real.$
(4) $V_1 \rightarrow id, V_2$	$id.dtype = V_1.dtype.$
~~(5)~~	$V_2.dtype = V_1.dtype.$
(5) $V \rightarrow id.$	$id.dtype = V.dtype.$

dtype is an inherited attribute.

float x, y.


D
T.dtype = real
V1.dtye = T.dtype = real
float
id
,
id.dtype = V1.dtype = real
V2.dtype = V1.dtype = real
id.dtype = V.dtype = real

int a, b, c.



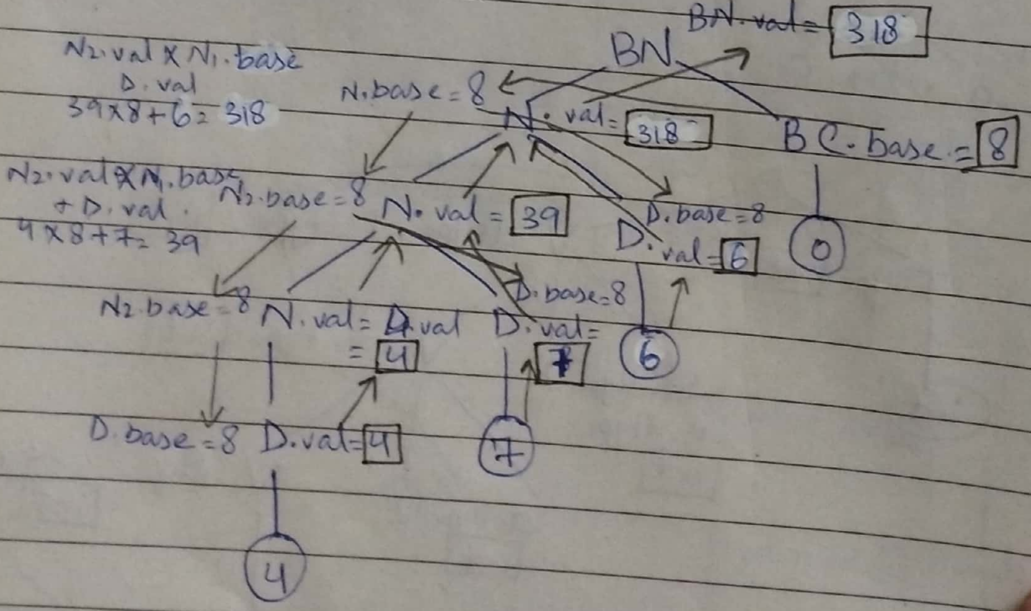
Date______20

Example #4.

	Semantic Rules.
① BN → N BC	BN.val = N.val | N.base = BC.base
② BC → o	BC.base = 8
③ BC → d	BC.base = 10
④ $N_1 \rightarrow N_2$ D	N_1.val = if (D.val = error or N_2.val = error then error else n_2.val * n_1.base + D.val N_2.base = N_1.base. D.base = N_1:base.
⑤ N → D.	N.val = D.val | D.base = N.base.
⑥ D → 0|1|2|3...|7	D.val = 0, 1, 2, 3, 4, 5, 6, 7
⑦ D → 8.	D.val = if (D.base = 8) then error else 8.
⑧ D → 9	D.val = if (D.base = 8) then error else 9.

base is an inherited attribute.

4760



Types of SDTs

S-Attributed SDT

- Synthesized attributes take its value from its children
- Use bottomup parsing

L-Attributed SDT

- Both synthesized & inherited attributes. take its value from parent or sibling (left).
- Use top-down-Parsing

Example.

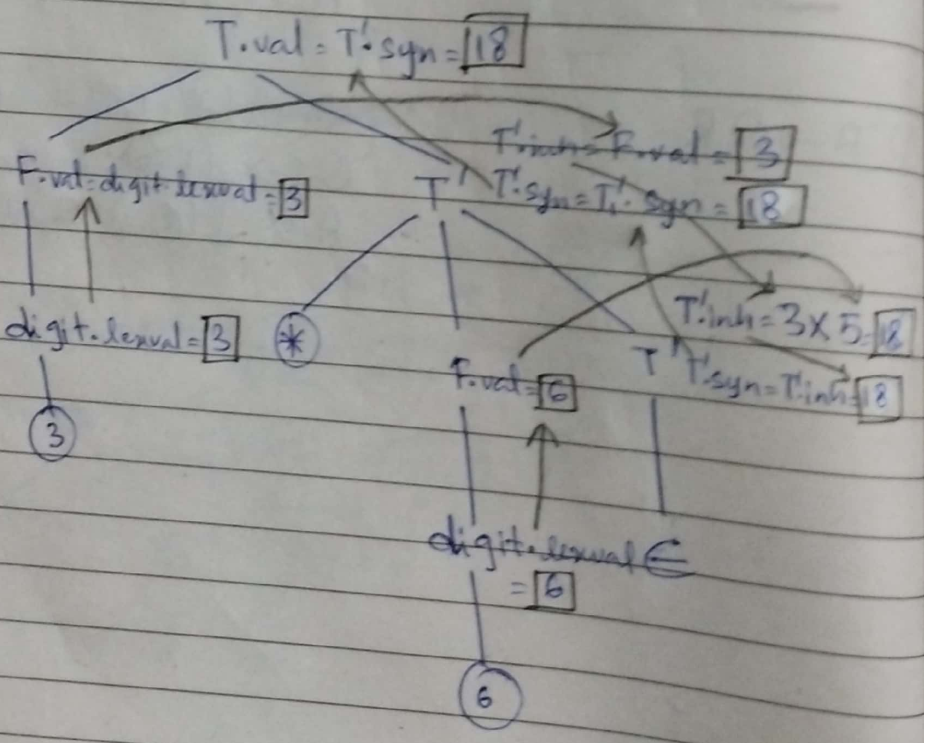
Grammar Rules	Semantic Rules
① $A \to LM$	$\{ L.i = l(A.i)$; Inherited $\left(\begin{smallmatrix}A\\ \downarrow\\ L\end{smallmatrix}\right)$ $M.i = m(L.s)$; Inherited $(L \to M)$ $A.s = f(M.s)$ } Synthesised $\left(\begin{smallmatrix}A\\ \uparrow\\ M\end{smallmatrix}\right)$ $\}$
② $A \to QR$	$\{ R.i = r(A.i)$; Inherited $\left(\begin{smallmatrix}A\\ \downarrow\\ R\end{smallmatrix}\right)$. $Q.i = q(R.s)$; Inherited $(R \leftarrow Q)$ $A.s = f(Q.s)$; Synthesized $\left(\begin{smallmatrix}A\\ \uparrow\\ Q\end{smallmatrix}\right.$ $\}$

Example 5.3:

Grammar / Production	Semantic Rules
1) $T \rightarrow F T'$	$T'.inh = F.val$ (Inherited $T' \leftarrow F$) $T.val = T'.syn$ (Synthesized $T \leftarrow T'$)
2) $T' \rightarrow * F T'_1$	$T'_1.inh = T'.inh \times F.val$ ([illegible] $T'.syn = T'_1.syn$ (Synthesized $T' \leftarrow T'_1$)
3) $T' \rightarrow \epsilon$	$T'.syn = T'.inh$ (Inherited)
4) $F \rightarrow digit$	$F.val = digit.lexval$ (Synthesized)

val and syn are synthesized attributes.
inh is an inherited attribute

3 * 6

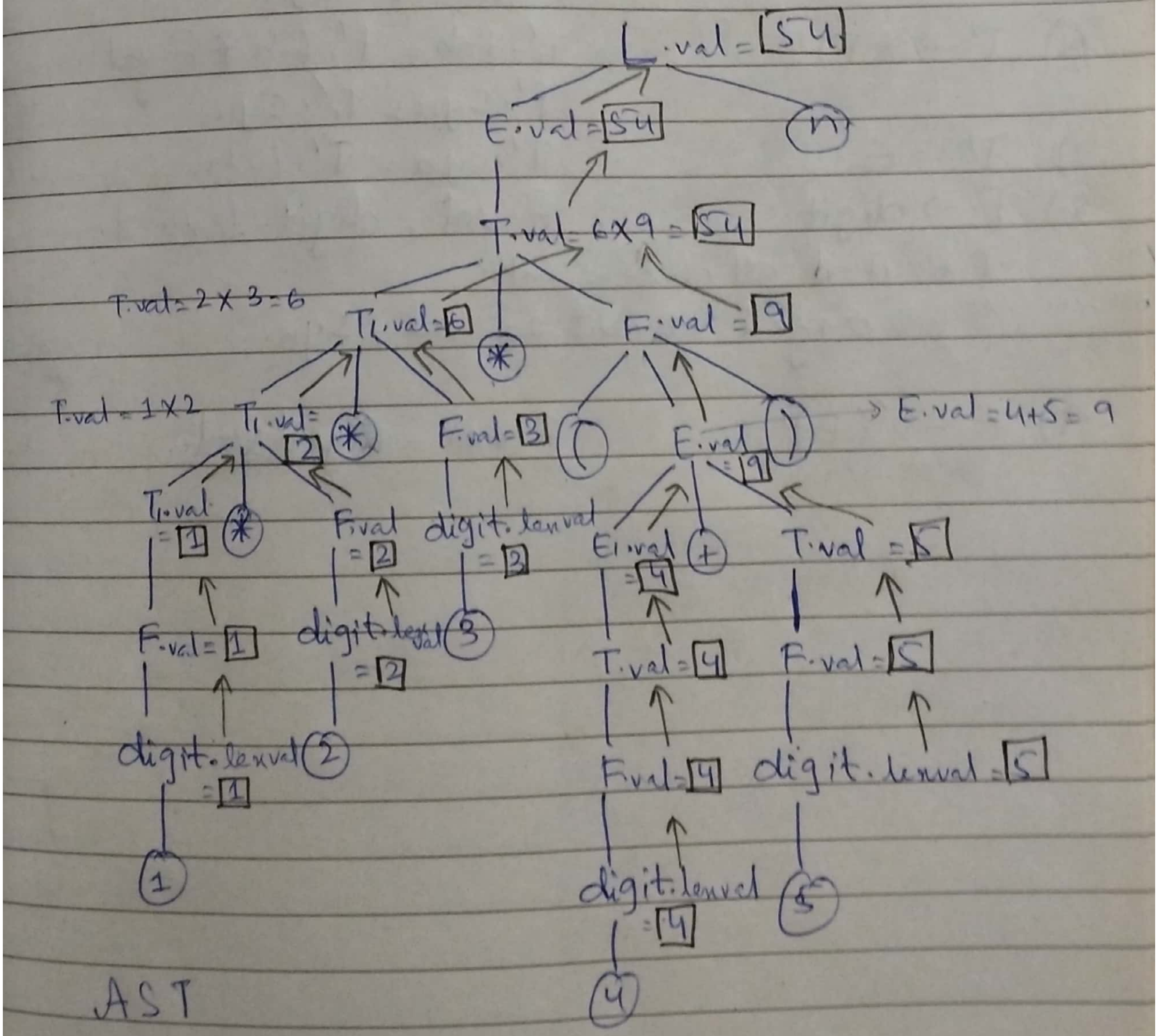


Exercise Question: 5·1t.

Grammar / Productions	Semantic rules.
1) L → E n	L. val = E. val.
2) E → E₁ + T	E. val = E₁. val + T. val.
3) E → T	E. val = T. val.
4) T → T₁ * F	T. val = T₁. val × F. val
5) T → F.	T. val = F. val
6) F → (E)	F. val = E. val
7) F → digit	F. val = digit. lexval.

val is a synthesized attribute.

1 * 2 * 3 * (4 + 5) n.



AST

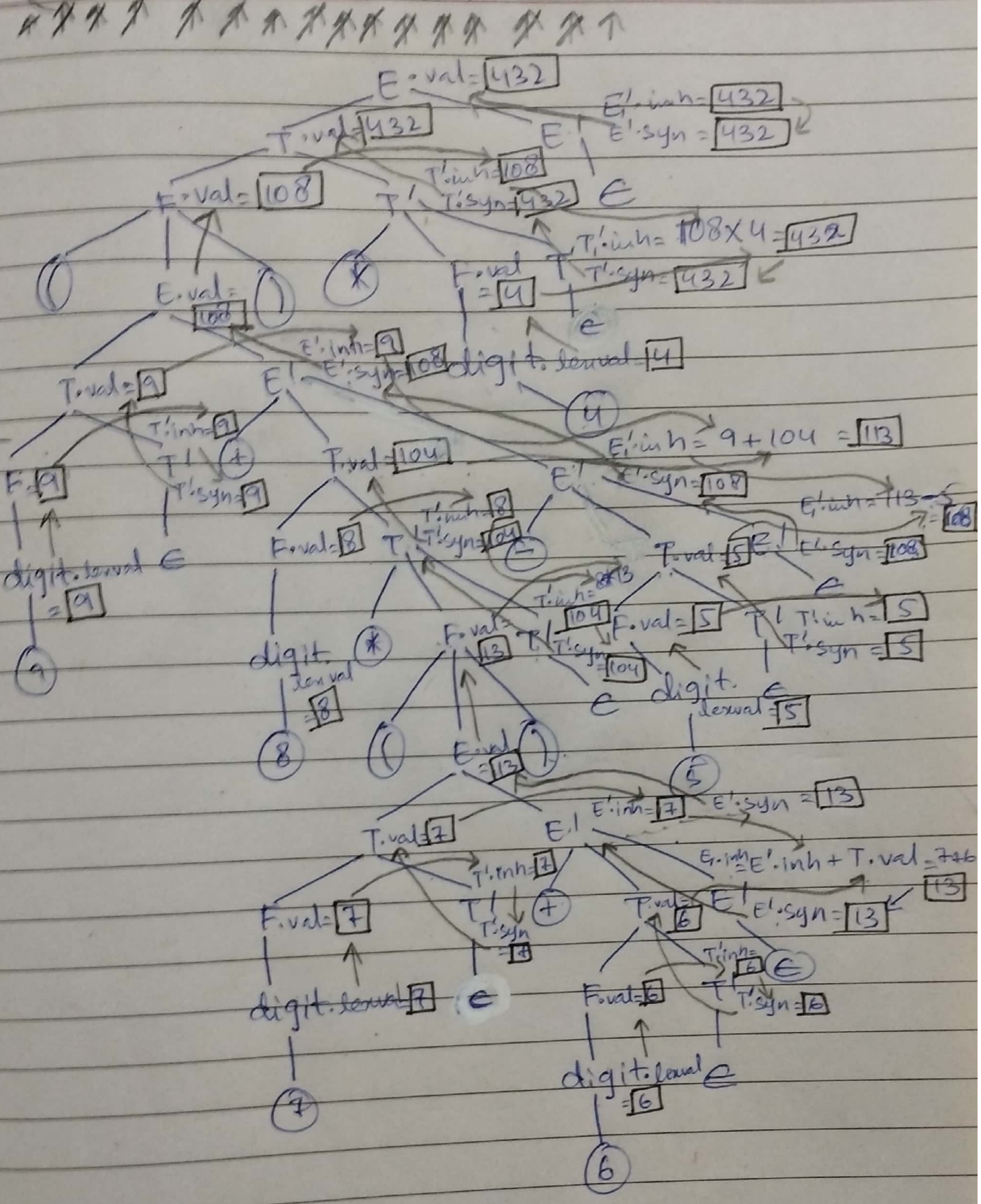
Date

Exercise Question 5.12.

Grammar/Productions	Semantic Rules.
1) $E \rightarrow TE'$	$E'.inh = T.val$ $E.val = E'.syn.$
2) $E' \rightarrow +TE'$	$E'_1.inh = E'.inh + T.val$ $E'.Syn = E'_1.syn.$
3) $E' \rightarrow -TE'$	$E'_1.inh = E'.inh - T.val.$ $E'.syn = E'_1.syn$
4) $E' \rightarrow \in$	$E'.syn = E'.inh.$
5) $T \rightarrow FT'$	$T'.inh = F.val$ $T.val = T'.syn.$
6) $T' \rightarrow *FT'$	$T'_1.inh = T'.inh * F.val$ $T'.syn = T'_1.syn.$
7) $T' \rightarrow \in$	$T'.syn = T'.inh.$
8) $F \rightarrow digit.$	$F.val = digit.lexval.$
9) $F \rightarrow (E)$	$F.val = E.val$

Inherited attribute = inh.
Synthesized attributes = val, syn.

$(9+8*(7+6)-5)*4$


E.val = 432
E'.inh = 432
E'.syn = 432
T.val = 432
E'
∈
T'.inh = 108
T'.syn = 432
F.val = 108
T'
T'.inh = 108×4 = 432
T'.syn = 432
(
E.val = 108
)
*
F.val = 4
T'
∈
digit.lexval = 4
4
E'.inh = 9
E'.syn = 108
T.val = 9
E'
T'.inh = 9
T'
+
T'.syn = 9
F = 9
digit.lexval = 9
∈
9
T.val = 104
E'.inh = 9+104 = 113
E'
E'.syn = 108
E'.inh = 113−5 = 108
F.val = 8
T'
T'.inh = 8
T'.syn = 104
−
T.val = 5
E'
E'.syn = 108
∈
digit.lexval = 8
8
*
T'.inh = 8×13
104
F.val = 13
T'
T'.syn = 104
∈
F.val = 5
T'
T'.inh = 5
T'.syn = 5
digit.lexval = 5
∈
5
(
E.val = 13
)
E'.inh = 7
E'.syn = 13
T.val = 7
E'
T'.inh = 7
+
F.val = 7
T'
T'.syn = 7
T.val = 6
E'
E'.inh = E'.inh + T.val = 7+6 = 13
E'.syn = 13
digit.lexval = 7
∈
7
T'.inh = 6
∈
F.val = 6
T'
T'.syn = 6
digit.lexval = 6
∈
6