2017

MADE EASY

WORKBOOK

Detailed Explanations of
Try Yourself *Questions*

Computer Science & IT

Compiler Design

MADE EASY Publications

1 Introduction and Lexical Analysis

Detailed Explanation of Try Yourself Questions

T1 : Solution

(d)

Lexical analyzer produces an error when an illegal character appears in the string pattern that makes invalid token.

T2 : Solution

(b)

G_3 and G_4 are set of left recursive and set of right recursive grammars respectively. $L(G_3) \cong L(G_4)$. Given any left recursive grammar can be converted to right recursive and vice-versa.

■■■■

MADE EASY Publications

2 Parsing and Syntex Directed Translation

Detailed Explanation of Try Yourself Questions

T1 : Solution

(b)

For any particular string:

parse trees = # LMD's = # RMD's

$\therefore \quad l = P = r$

T2 : Solution

(d)

In L-attributed, all attributes follow synthesized attributed notation or restricted inherited attribute notation (parent or left sibling).

In S-attribute, all attributes follow synthesized attribute notation only.

$F \rightarrow (E)G \{F.val = E.val = G.val\}$ This notation is not in L-attributed, and also not S-attributed.

T3 : Solution

(d)

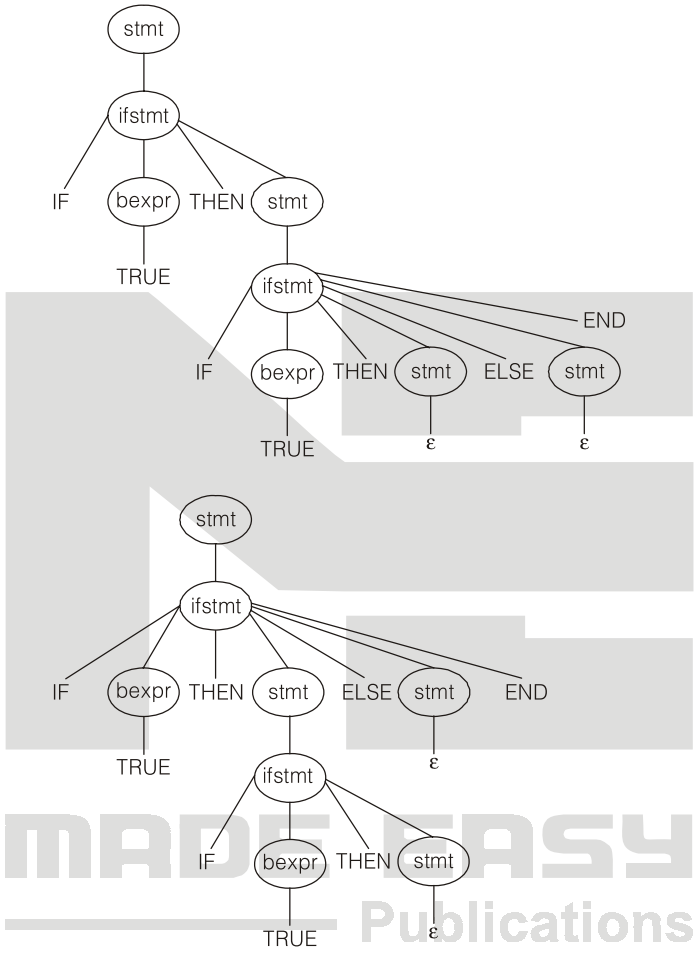
Recursive descent parser can not use left recursive grammar but it can use right recursive grammar.

MADE EASY Publications

T4 : Solution

(a)

For string : "IF TRUE THEN IF TRUE THEN ELSE END" has two parse trees.



∴ The given grammar is ambiguous, hence it is not LL(1) and also not LR(1).

T5 : Solution

(d)

Set 1 has RR conflict.

$A \rightarrow a.$, {b}

$B \rightarrow ba.$, {b, c}

$\{b\} \cap \{b, c\} \neq \phi$

∴ Grammar produces RR conflict for CLR (1).

 © Copyright

T6 : Solution

(d)

Set 3 contain $S \rightarrow \cdot$ as reduced item.

Follow (S) = {), $}

In row 3, entry for column ')' and '$' will be "$r_b$".

$E_1 = r_b$, $E_2 = r_b$ [$\because$ $b:S \rightarrow \varepsilon$]

In set 5, on S it goes to set 6.

In row 5, entry for non-terminals 'S' is state 6.

$E_3 = 6$ [$\because$ $(5) \xrightarrow{S} (6)$]

$E_1 = r_b$, $E_2 = r_b$ and $E_3 = 6$

T7 : Solution

(c)

FOLLOW (A) = LFOLLOW(A) = RFOLLOW(A)

The set of terminals followed by A are same in all sentential forms.

T8 : Solution

(a)

(a) An unambiguous grammar can have different leftmost and rightmost derivation. However, an unambiguous grammar has only one derivation tree. So option (a) is false.
(b) LL(1) is a top-down parser.
(c) LALR is more powerful than SLR.
(d) For any parser, grammar should be unambiguous.

T9 : Solution

(a)

For Input string: aab

LR parser reduces aS to S to parse the string aab when stack has aaS.

■■■■

3 Runtime Environment

Detailed Explanation of Try Yourself Questions

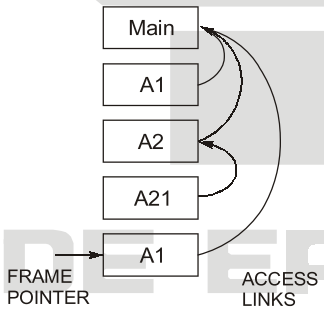
T1 : Solution

(b)

Activation record maintains: Frame pointer, Access link, Local variables, Return address, Return value, etc.

T2 : Solution

(d)



Given calling sequence from the program is: Main → A1 → A2 → A21 → A1

A1 and A2 are defined in Main, so A1 and A2 access links are pointed to Main.

A21 definition is available in A2, hence A21 access link points to A2.

■■■■