

1. 给定 CFG 文法 $G: S \rightarrow S + S \mid SS \mid (S) \mid S * \mid a$ 和输入串 $(a + a) * a$

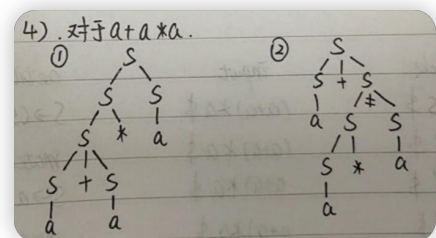
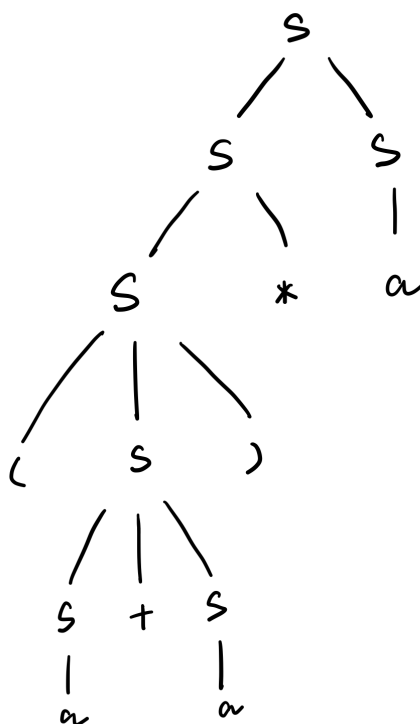
(1) 给出这个输入串的一个最左推导(Leftmost Derivation);

$S \Rightarrow SS \Rightarrow S * S \Rightarrow (S) * S \Rightarrow (S + S) * S \Rightarrow (a + S) * S \Rightarrow (a + a) * S \Rightarrow (a + a) * a$

(2) 给出这个输入串的一个最右推导(Rightmost Derivation);

$S \Rightarrow SS \Rightarrow Sa \Rightarrow S * a \Rightarrow (S) * a \Rightarrow (S + S) * a \Rightarrow (S + a) * a \Rightarrow (a + a) * a$

(3) 给出这个输入串的一颗语法分析树(Parse Tree);



(4) 判断该文法是否为二义性(Ambiguity)文法并给出适当说明或解释;

该文法是二义性文法, 说明: 例如, 对+没有指定结合性, 对于串 $a+a+a$, 运用最左推导, 有两棵对应的语法分析树, 故该文法是二义性文法

(5) 该文法是否为 LL 文法? 若是, 说明原因; 若否, 请将其调整为 LL。

该文法不是 LL 文法, 因为 $FIRST(S + S) \cap FIRST(SS) \cap FIRST(S*) \neq \emptyset$

调整为 LL 文法:

$S \rightarrow aS' \mid (S)$

$S' \rightarrow +SS' \mid *S'S' \mid \epsilon$

期末如果有类似的题目希望大家可以像这样简洁一些
推导过程+最终一个简洁结果会比较好

2. 给定 CFG 文法 G:

$$E \rightarrow TB$$

$$T \rightarrow FC$$

$$B \rightarrow ATB|\epsilon$$

$$C \rightarrow MFC|\epsilon$$

$$F \rightarrow (E)|a$$

$$A \rightarrow +|-$$

$$M \rightarrow *|/$$

(1) 求文法 G 的 FIRST 和 FOLLOW 集

FIRST集:

$$FIRST(E) = FIRST(T) = FIRST(F) = \{ (, a \}$$

$$FIRST(B) = \{ +, -, \epsilon \}$$

$$FIRST(C) = \{ *, /, \epsilon \}$$

$$FIRST(A) = \{ +, - \}$$

$$FIRST(M) = \{ *, / \}$$

FOLLOW集:

$$FOLLOW(E) = FOLLOW(B) = \{), \$ \}$$

$$FOLLOW(T) = FOLLOW(C) = \{), +, -, \$ \}$$

$$FOLLOW(F) = \{), +, -, *, /, \$ \}$$

$$FOLLOW(A) = FOLLOW(M) = \{ (, a \}$$

(2) 判断 G 是否为 LL(1)文法, 并详细说明理由

G为LL(1)文法, 原因是对于 $B \rightarrow ATB|\epsilon$,

$FIRST(ATB) = \{ +, - \} \cap FOLLOW(B) = \{), \$ \} \neq \emptyset$; 对于 $C \rightarrow MFC|\epsilon$,

$FIRST(MFC) = \{ *, / \} \cap FOLLOW(C) = \{), +, -, \$ \} \neq \emptyset$;

3. 给定以下 CFG 文法 G1 和 G2:

G1: $S \rightarrow A$ $A \rightarrow \epsilon$ $A \rightarrow bbA$	G2: $S \rightarrow A$ $A \rightarrow \epsilon$ $A \rightarrow Abb$
---	---

(1) 对文法 G1, 求解其 FIRST 和 FOLLOW 集, 并构建 LL(1)分析表;

FIRST集:

$$FIRST(S) = FIRST(A) = \{ b, \epsilon \}$$

FOLLOW集:

$$FOLLOW(S) = FOLLOW(A) = \{ \$ \}$$

LL(1)分析表:

table	b	\$
S	$S \rightarrow A$	$S \rightarrow A$
A	$A \rightarrow bbA$	$A \rightarrow \epsilon$

(2) 判断 G1 和 G2 是否是 LL(1)文法, 并简述理由。若不是, 怎样使其成为 LL(1)文法?

G1是LL(1)文法, 因为G1的LL(1)分析表中每个表项至多有一条规则。G2不是LL(1)文法, 因为存在左递归, 并且:

$$FIRST(\epsilon) = \{ \epsilon \}, FOLLOW(A) \cap FIRST(Abb) = \{ b \} \neq \emptyset$$

修改文法消除左递归:

$S \rightarrow A$
 $A \rightarrow \epsilon$
 $A \rightarrow A'$
 $A' \rightarrow bbA' \mid \epsilon$

(3) 请借助(1)中得到的 G1 文法 LL(1)分析表, 列出解析输入串 bbbb 的过程, 包括 每一步输入串、栈的变化情况和应用的产生式。注: 如有需要, 请自行添加更多行

Stack(left is top)	Input	Action
S\$	bbbb\$	$S \rightarrow A$
A\$	bbbb\$	$A \rightarrow bbA$
bbA\$	bbbb\$	match
bA\$	bbb\$	match
A\$	bb\$	$A \rightarrow bbA$
bbA\$	bb\$	match
bA\$	b\$	match
A\$	\$	$A \rightarrow \epsilon$
\$	\$	accept