

作业说明

- 在每一章结束后的一周内统一提交；
- 邮件标题格式：学号姓名—第X章
 - ◆ 作业提交到课程平台；
- 注意格式、学习画图；
- 严禁手写拍照上传—违者计作业分0分

第二章作业

- 2-2 (1) (2) 2-3 (1) (2)
- 2-6 2-10
- 2-11 (2) (3) 2-14 (1) (2)
- 补充作业——课程平台下载
- 自己检查是否存在重复

第二章 上下文无关文法和语言

§ 2.1 文法和语言的表示

§ 2.2 文法和语言的定义

§ 2.3 句型的分析

§ 2.4 文法的实用限制和其他表示法

§ 2.5 文法和语言的Chomsky分类

§ 2.3 句型的分析

句型分析：识别输入的符号串是否为某一文法的句型（或句子）的过程。

例：G[E]：E → E+E | E * E | (E) | i

句型 $i+i*i$ 的推导序列有哪些？

不同推导序列 → 同一句型（句子）

§ 2.3 句型的分析

§ 2.3.1 规范推导和规范归约

- 1、最左(右)推导: 在任一步推导 $V \Rightarrow W$ 中, 都是对符号串 V 的最左(右)非终结符号进行替换, 称最左(右)推导。
- 2、规范推导: 即最右推导
- 3、规范句型: 由规范推导所得的句型 (**最右句型**)
- 4、规范归约: 规范推导的逆过程, 称规范归约或**最左归约**

例: $G[\langle \text{标识符} \rangle]$

$$\langle \text{标识符} \rangle \rightarrow \langle \text{字母} \rangle \mid \langle \text{标识符} \rangle \langle \text{字母} \rangle \\ \mid \langle \text{标识符} \rangle \langle \text{数字} \rangle$$
$$\langle \text{字母} \rangle \rightarrow a \mid b \mid \dots \mid z \mid A \mid \dots \mid Z$$
$$\langle \text{数字} \rangle \rightarrow 0 \mid 1 \mid 2 \mid \dots \mid 9$$

问题: 给出句子 $a4y$ 的规范推导和规范归约.

给出句子 $a4y$ 的最左推导.

例: $G[\langle \text{标识符} \rangle]$

$\langle \text{标识符} \rangle \rightarrow \langle \text{字母} \rangle \mid \langle \text{标识符} \rangle \langle \text{字母} \rangle$

$\mid \langle \text{标识符} \rangle \langle \text{数字} \rangle$

$\langle \text{字母} \rangle \rightarrow a \mid b \mid \dots \mid z \mid A \mid \dots \mid Z$

$\langle \text{数字} \rangle \rightarrow 0 \mid 1 \mid 2 \mid \dots \mid 9$

请注意: 规范推导和归范归约互为

推导——自上而下的分析

规范推导

$\langle \text{标识符} \rangle$

$\Rightarrow \langle \text{标识符} \rangle \langle \text{字母} \rangle$

$\Rightarrow \langle \text{标识符} \rangle y$

$\Rightarrow \langle \text{标识符} \rangle \langle \text{数字} \rangle y$

$\Rightarrow \langle \text{标识符} \rangle 4y$

$\Rightarrow \langle \text{字母} \rangle 4y$

$\Rightarrow a4y$

问题: 如何正确选择规则?

规范归约

$a4y$

$\neq \langle \text{字母} \rangle 4y$

$\neq \langle \text{标识符} \rangle 4y$

$\neq \langle \text{标识符} \rangle \langle \text{数字} \rangle y$

$\neq \langle \text{标识符} \rangle y$

$\neq \langle \text{标识符} \rangle \langle \text{字母} \rangle$

$\neq \langle \text{标识符} \rangle$

问题: 如何准确选择可归约串?

归约——自下而上的分析

§ 2.3.2 短语、简单短语和句柄

-----语言句型中的几个概念

1、短语: 文法 $G[Z]$, $\omega = xuy$ 是一句型, $x, y \in V^*$

如有 $Z \overset{*}{\Rightarrow} xUy$, 且 $U \overset{+}{\Rightarrow} u$, $U \in V_n$, $u \in V^+$

称 u 是一个相对于非终结符号 U 句型 ω 的**短语**.

2、简单短语: 文法 $G[Z]$, $\omega = xuy$ 是一句型,

如有 $Z \overset{*}{\Rightarrow} xUy$, 且 $U \Rightarrow u$, $U \in V_n$, $u \in V^+$

称 u 是一个相对于非终结符号 U 句型 ω 的**简单短语**.

3、句柄: 句型最左边的简单短语为该句型的**句柄**.

短语: 文法 $G[Z]$, $\omega = xuy$ 是一句型,

如有 $Z \Rightarrow^* xUy$, 且 $U \Rightarrow^+ u$, $U \in V_n$, $u \in V^+$

称 u 是一个相对于非终结符号 U 句型 ω 的短语.

$\omega = xuy$ 是一句型

结论: u 是一个相对于非终结符号 U 句型 ω 的短语

$u? \quad U?$

$U \in V_n, u \in V^+, x, y \in V^*$

$Z \Rightarrow^* xUy$

$U \Rightarrow^+ u$

$Z \Rightarrow^* xUy \Rightarrow^+ xuy$

简单短语: 文法 $G[Z]$, $\omega = xuy$ 是一句型,

如有 $Z \Rightarrow^* xUy$, 且 $U \Rightarrow u$, $U \in V_n$, $u \in V^+$

称 u 是一个相对于非终结符号 U 句型 ω 的简单短语

$\omega = xuy$ 是一句型

结论: u 是一个相对于非终结符号 U 句型 ω 的简单短语

$u?$ $U?$

$U \in V_n$, $u \in V^+$, $x, y \in V^*$

$Z \Rightarrow^* xUy$

$U \Rightarrow u$

u 是某产生
式的右部

$Z \Rightarrow^* xUy \Rightarrow xuy$

句柄: 句型最左边的简单短语为该句型的句柄.

说明：

- 短语或简单短语必须是对某一句型来说的，并且是该句型的一个子串。
- 短语或简单短语必须是相对某一非终结符号的。
- 两个条件缺一不可。
- 一句型可以有几个短语和简单短语。
- 一句型只有一个句柄（无二义性的文法）。
- 最左归约归约的是当前句型的句柄

例G[S]:

$S \rightarrow AB$

$A \rightarrow Aa | bB$

$B \rightarrow a | Sb$

问题: 给出句型baSb的短语、简单短语和句柄.

Sb是相对于B句型baSb的短语且为简单短语

a是相对于B句型baSb的短语且为简单短语.

ba是相对A句型baSb的短语.

句柄为a.

baSb是相对S句型baSb的短语.

(1) $S \Rightarrow bBB \Rightarrow ba\underline{B} \Rightarrow ba\underline{Sb}$ 且 $B \Rightarrow \underline{Sb}$
 $S \Rightarrow AB$ U u u

(2) $S \Rightarrow ASb \Rightarrow b\underline{B}Sb \Rightarrow ba\underline{Sb}$ 且 $B \Rightarrow \underline{a}$

$S \Rightarrow AB$

(3) $S \Rightarrow \underline{A}Sb \Rightarrow ^+\underline{ba}Sb$ 且 $A \Rightarrow ^+\underline{ba}$

$S \Rightarrow AB$

(4) $S \Rightarrow ^*\underline{S} \Rightarrow ^+\underline{ba}Sb$ 且 $S \Rightarrow ^+\underline{ba}Sb$

短语: 文法G[Z], $\omega = xu y$ 是一句型,

如有 $Z \Rightarrow^* xUy$, 且 $U \Rightarrow^+ u$, $U \in V_n$, $u \in V^+$

称u是一个相对于非终结符号U句型 ω 的短语.

§ 2.3.3 语法树

语法树 (Syntax Tree) —— 描述一个句型或句子的语法结构

树：若干个结点组成的有限集



m —— 结点n的直接前驱或父结点

n —— 结点m的直接后继或子结点

根 —— 仅有的一个没有任何前驱的结点

树 —— 无回路性和连通性

§ 2.3.3 语法树

1、语法树：一个句型或句子推导过程的图示法表示，形成一棵语法树。

例G[S]:

$S \rightarrow AB$

$A \rightarrow Aa | bB$

$B \rightarrow a | Sb$

句型baSb

最左推导

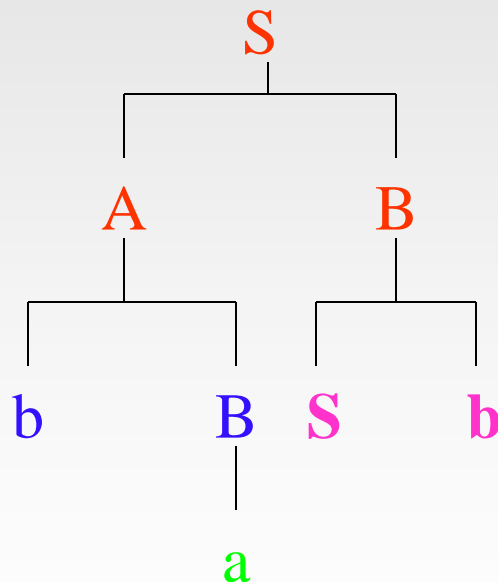
$S \Rightarrow AB$

$\Rightarrow bBB$

$\Rightarrow baB$

$\Rightarrow baSb$

语法树



例 $G[S]$:
 $S \rightarrow AB$
 $A \rightarrow Aa | bB$
 $B \rightarrow a | Sb$

句型 $baSb$

另一种推导的语法树

推导

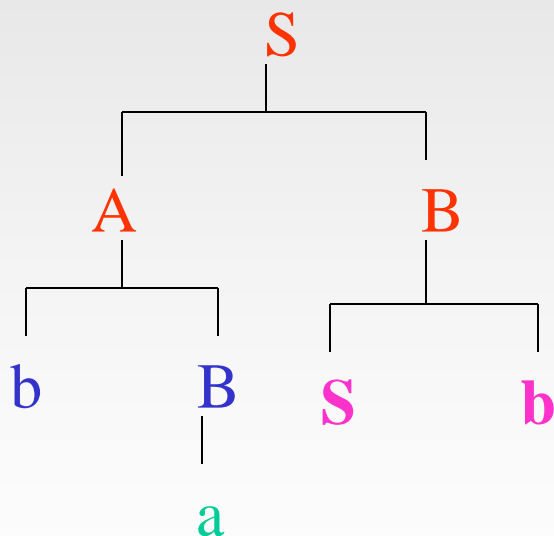
$S \Rightarrow AB$

$\Rightarrow ASb$

$\Rightarrow bBSb$

$\Rightarrow baSb$

语法树



几个结论:

- 1、对每个语法树, 至少存在一个推导过程
- 2、对于每个推导, 都有一个相应的语法树(但不同的推导可能有相同的语法树)。
- 3、树的末端结点形成所要推导的句型。

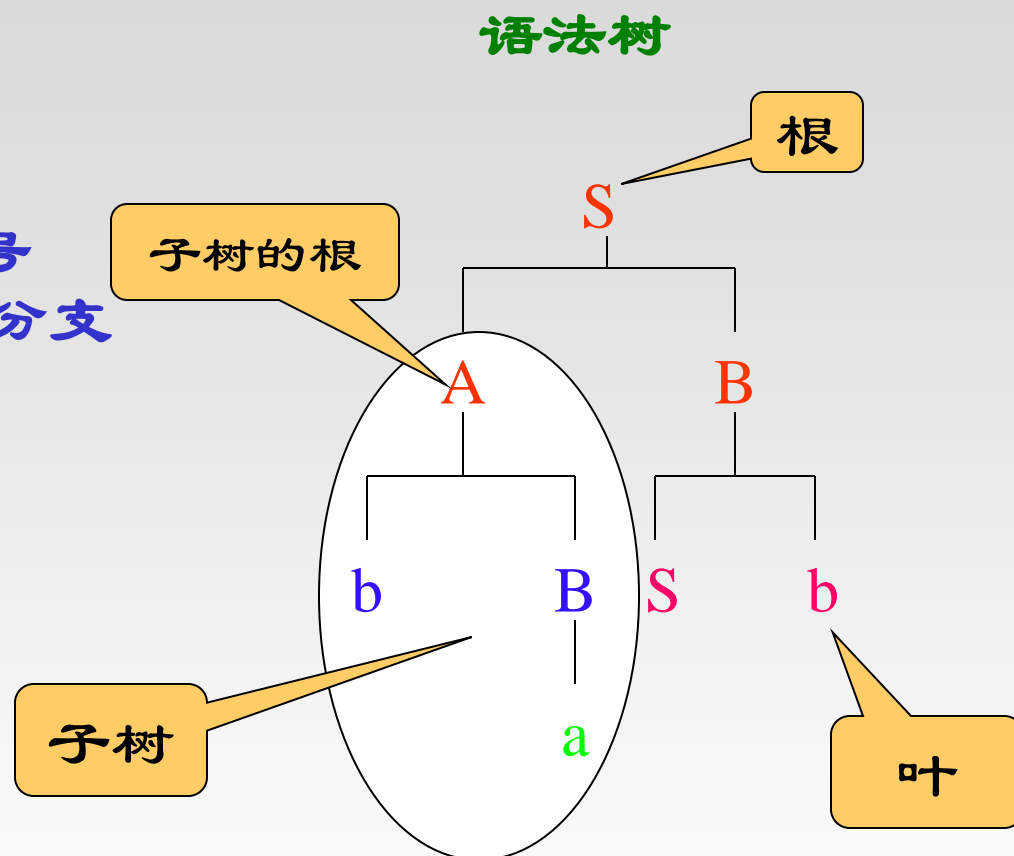
但这个句型也可能对应两棵不同的语法树, 这就是文法的二义性问题.

2、语法树与子树

根: 开始符号

子树: 某一非终结符号
(子树的根) 及其下面的分支

叶: 树的末端结点



语法树的全部末端结点 (自左向右) 形成当前句型

§ 2.3.4 子树与短语、句柄

——通过树来寻找短语、简单短语、句柄

1、短语:子树的末端结点形成的符号串.

这个短语相对的句型:整个树的末端结点.

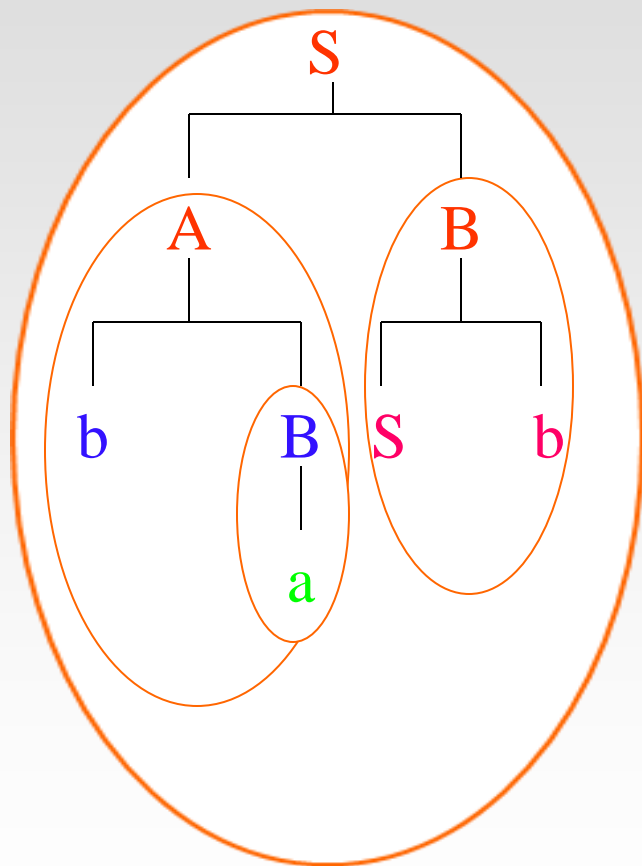
非终结符号:子树的根

2、简单子树:只有一层分支的子树

3、简单短语:简单子树的末端结点形成的符号串

4、句柄:**最左**简单子树的末端结点形成的符号串.

上例G[S]: 句型baSb的语法树



共有四棵子树,

四个短语: baSb, ba, a, Sb

简单短语: a, Sb

句柄: a

这样的结论与短语定义
完全符合, 为什么?

4、归约

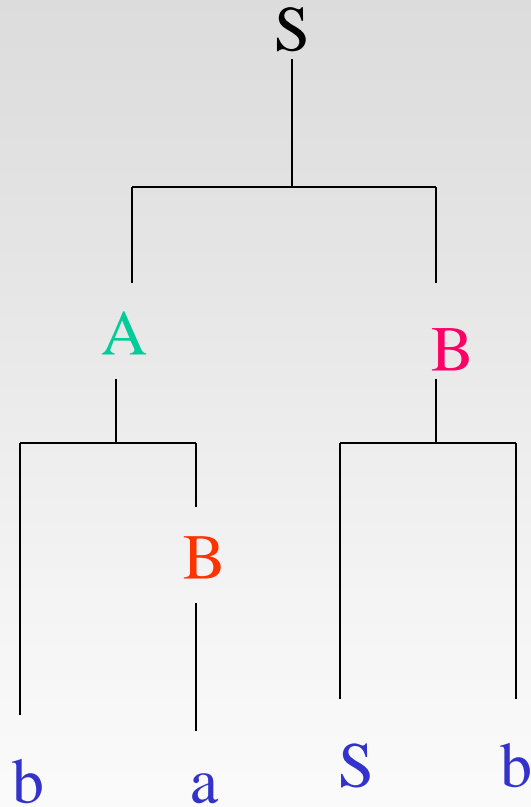
语法树由下向上生长，

通过规则替换到达开始符号的过程。

- 一个句型最左归约所归约的是当前句型的句柄。
- 这个过程非常重要，因为源程序都是符号串形式的，这就需要把它归约为开始符号才算正确。
- 最左归约关键是找当前句型的句柄，这个问题，到语法分析时再着重讲解。
- 二义性文法最左归约的句柄的不唯一性。

例 $G[S]$:
 $S \rightarrow AB$
 $A \rightarrow Aa | bB$
 $B \rightarrow a | Sb$

句型 $baSb$ 的归约过程.



归约过程

$baSb$

$\not\leftarrow bBSb$

$\not\leftarrow ASb$

$\not\leftarrow AB$

$\not\leftarrow S$

§ 2.3.5 文法的二义性

- 如果文法 G 的某一个句子存在两棵或两棵以上不同的语法树，则称句子是二义性的。
- 如果一文法含有二义性的句子，则称该文法是二义性的，否则该文法是无二义性的。

例 $G[E]: E \rightarrow E + E \mid E * E \mid (E) \mid i$

句子 $i+i*i$ 同是最左推导, 对应两棵不同的语法树

最左推导1:

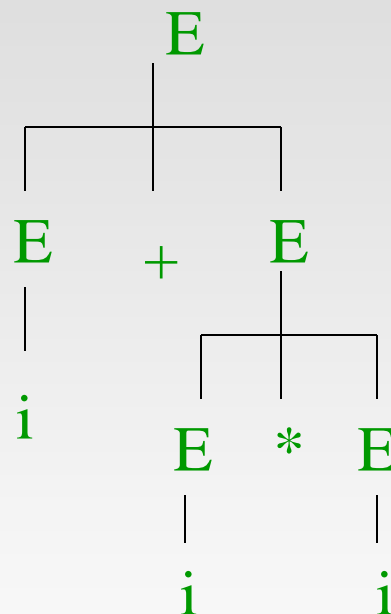
$E \Rightarrow E + E$

$\Rightarrow i + E$

$\Rightarrow i + E * E$

$\Rightarrow i + i * E$

$\Rightarrow i + i * i$

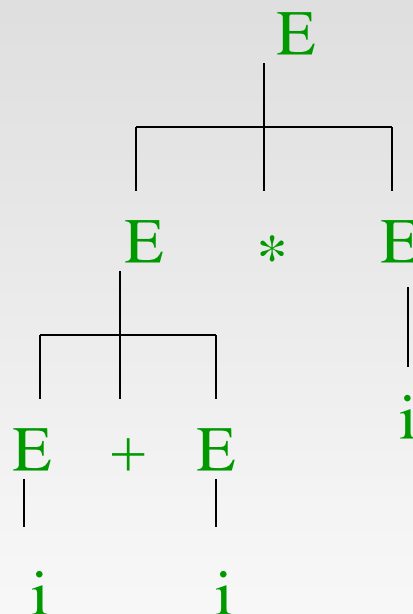


例 $G[E]: E \rightarrow E + E \mid E * E \mid (E) \mid i$

句子 $i+i*i$ 同是最左推导, 对应两棵不同的语法树

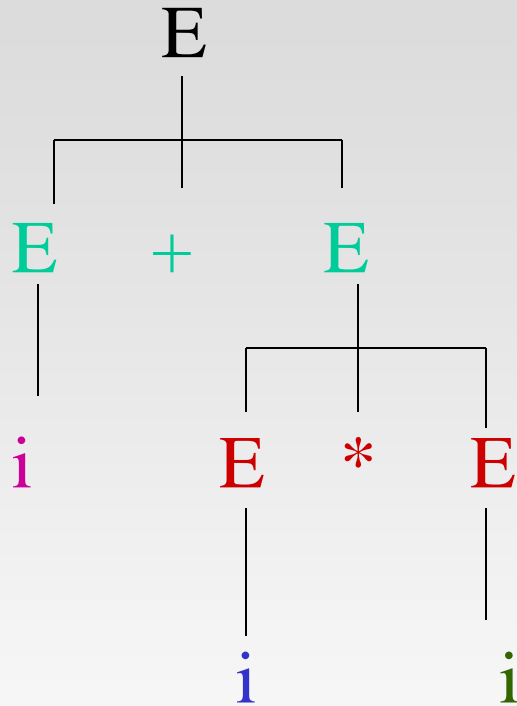
最左推导2:

$E \Rightarrow E * E$
 $\Rightarrow E + E *$
 $\Rightarrow i + E * E$
 $\Rightarrow i + i * E$
 $\Rightarrow i + i * i$



二义性的句子

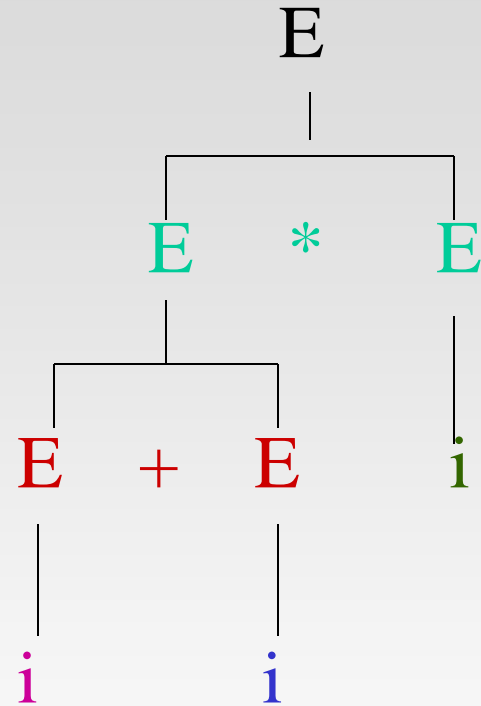
语法树1



句柄: $i, i, i, E * E, E + E$

表示: $i + (i * i)$

语法树2



句柄: $i, i, E + E, i, E * E$

表示: $(i + i) * i$

说明：

①文法的二义性：

某一句子有二个不同的最左（右）推导
或二个不同的最左（规范）归约

②文法的二义性是不可判定的：不存在一种算法，只能用一些简单条件来判定

③特例：若一文法 G 既含左递归又含右递归，则 G 必是二义性文法。（是经验）

例： $G[E]: E \rightarrow E + E | E * E | (E) | i$

文法二义性的消除

1、不改变文法中的原有规则，仅加入一些语法的非形式规定

例： $G[E]: E \rightarrow E + E \mid E * E \mid (E) \mid i$

二义性文法

$i + i * i$

规定：*运算优先级高于+运算，且服从左结合

例： $G[S]: S \rightarrow \text{if } B \text{ then } S \text{ else } S$
 $\mid \text{if } B \text{ then } S$

二义性文法

If B1 then if B2 then S1 else S2;

规定：else跟与它最近的尚未匹配的then匹配。

文法二义性的消除

2、构造一个等价的无二义性文法，把排除的二义性规则合并到原文法中（增加新的非终结符）

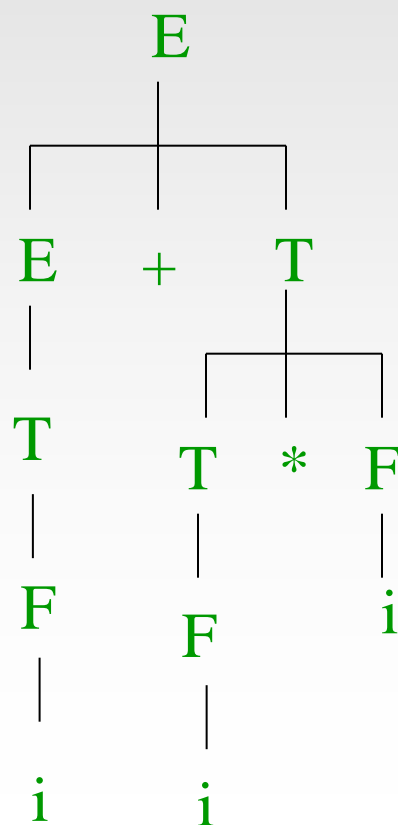
例： $G[E]: E \rightarrow E + E \mid E * E \mid (E) \mid i$

构造 $G_1[E]: E \rightarrow E + T \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow (E) \mid i$

无二义性文法



文法二义性的消除

2、构造一个等价的无二义性文法，把排除的二义性规则合并到原文法中（增加新的非终结符）

G[C]: $C \rightarrow \text{if } B \text{ then } C$
 $C \rightarrow \text{if } B \text{ then } C \text{ else } C$
 $C \rightarrow S$

构造G1[C]: $C \rightarrow C1 \mid C2$
 $C1 \rightarrow \text{if } B \text{ then } C1 \text{ else } C1 \mid S$
 $C2 \rightarrow \text{if } B \text{ then } C \mid \text{if } B \text{ then } C1 \text{ else } C2$

无二义性文法

G3[S]: $S \rightarrow A \mid S-A$

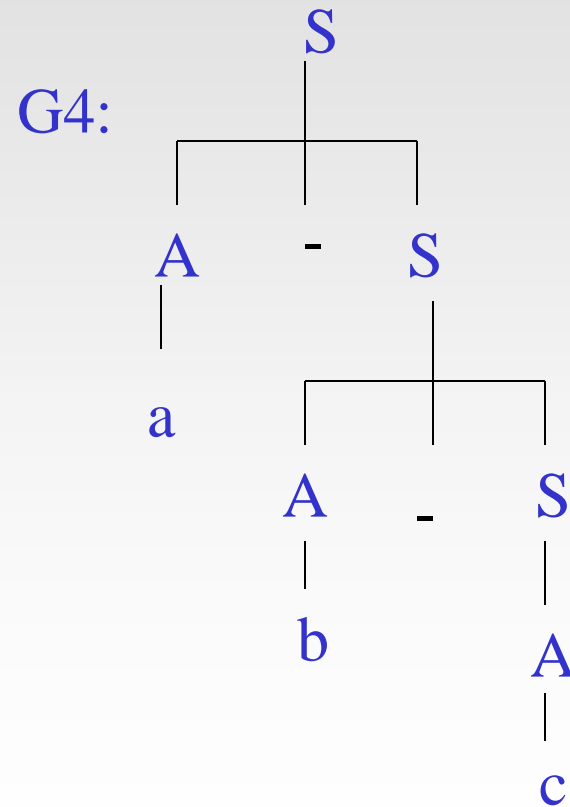
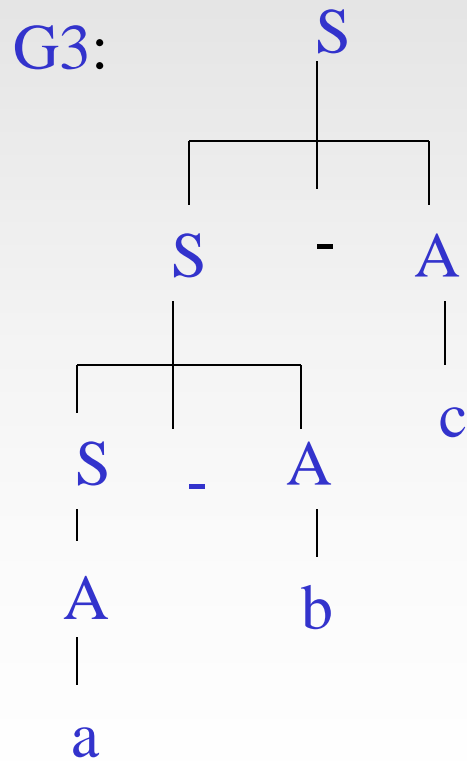
$A \rightarrow a \mid b \mid c$

G4[S]: $S \rightarrow A \mid A-S$

$A \rightarrow a \mid b \mid c$

G3, G4等价文法

对句子 a-b-c语义不同



特别说明：

文法的二义性不等同于语言的二义性

- (1) 通常我们说文法的二义性，而不说语言的二义性，因为有两个文法 G_1 和 G_2 ，其中一个是有二义性，另一个是无二义性，但却有 $L(G_1)=L(G_2)$ ，即两个文法所产生的语言是相同的；
- (2) 语言的二义性指的是它不存在无二义性的文法，通称为先天二义性的语言。

总结:

以上的分析中，忽略了两个问题

- (1) 自顶向下分析时：如有 $V \rightarrow x_1 | x_2 \dots | x_n$ 选哪一产生式可一次推导成功？
- (2) 自底向上分析时，如何尽快找到当前句型的句柄？（进行归约）

这些问题将在语法分析时解决。