上下文无关语言

上下文无关语言

- G_{bra}： $S \to S(S) | \varepsilon$
  $L(G_{bra})$不是RL,是CFL

- 高级程序设计语言的绝大多数语法结构都可以用上下文无关文法(CFG)描述。

上下文无关语言

- 主要内容
  - 关于CFL的分析
    - 派生和归约、派生树
  - CFG的化简
    - 无用符、单一产生式、空产生式
  - CFG的范式
    - CNF
    - GNF
  - CFG的自嵌套特性

上下文无关语言

- 文法G=(V，T，P，S)被称为是上下文无关的。如果除了形如$A\rightarrow\varepsilon$的产生式之外，对于$\forall\alpha\rightarrow\beta\in P$，均有$|\beta|\geqslant|\alpha|$，并且$\alpha\in V$成立。
- 关键：对于$\forall A\in V$，如果$A\rightarrow\beta\in P$，则无论A出现在句型的任何位置，我们都可以将A替换成β，而不考虑A的上下文。

上下文无关文法的派生树

- 算术表达式的文法

G_{exp1}：$E \to E+T|E-T|T$

$T \to T*F|T/F|F$

$F \to F \uparrow P|P$

$P \to (E)|N(L)|id$

$N \to sin|cos|exp|abs|log|int$

$L \to L,E|E$

上下文无关文法的派生树

- 算术表达式$x+x/y\uparrow 2$的不同派生

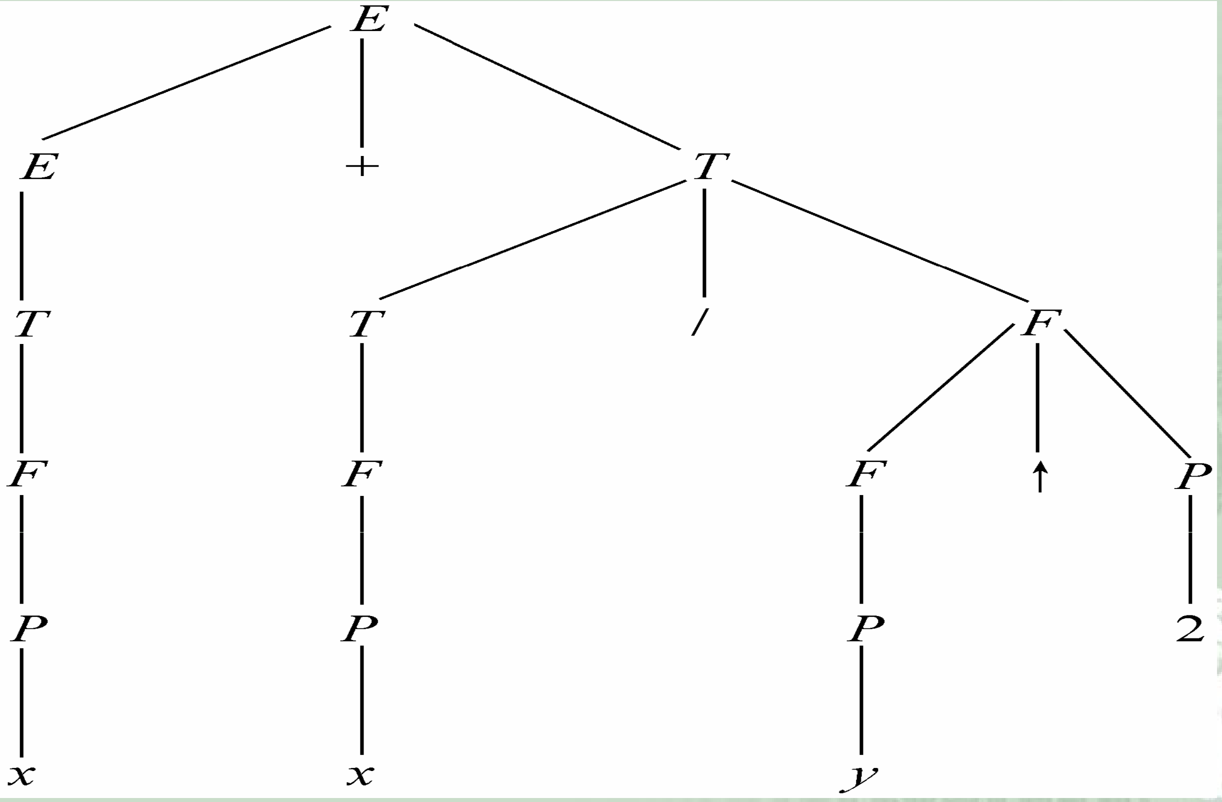
$E \Rightarrow E+T \Rightarrow T+T \Rightarrow F+T \Rightarrow P+T \Rightarrow x+T \Rightarrow x+T/F \Rightarrow x+F/F$
$\Rightarrow x+P/F \Rightarrow x+x/F \Rightarrow x+x/F\uparrow P \Rightarrow x+x/P\uparrow P \Rightarrow x+x/y\uparrow P$
$\Rightarrow x+x/y\uparrow 2$

$E \Rightarrow E+T \Rightarrow E+T/F \Rightarrow E+T/F\uparrow P \Rightarrow E+T/F\uparrow 2 \Rightarrow E+T/P\uparrow 2$
$\Rightarrow E+T/y\uparrow 2 \Rightarrow E+F/y\uparrow 2 \Rightarrow E+P/y\uparrow 2 \Rightarrow E+x/y\uparrow 2$
$\Rightarrow T+x/y\uparrow 2 \Rightarrow F+x/y\uparrow 2 \Rightarrow P+x/y\uparrow 2 \Rightarrow x+x/y\uparrow 2$

$E \Rightarrow E+T \Rightarrow T+T \Rightarrow T+T/F \Rightarrow F+T/F \Rightarrow F+T/F\uparrow P$
$\Rightarrow P+T/F\uparrow P \Rightarrow x+T/F\uparrow P \Rightarrow x+F/F\uparrow P \Rightarrow x+F/F\uparrow 2$
$\Rightarrow x+F/P\uparrow 2 \Rightarrow x+P/P\uparrow 2 \Rightarrow x+P/y\uparrow 2 \Rightarrow x+x/y\uparrow 2$

上下文无关文法的派生树

- 文法G_{exp1}句子x+x/y↑2的结构。



上下文无关文法的派生树

- **派生树(derivation tree)**
  - 一棵(有序)树(ordered tree)；
  - 树的每个顶点有一个标记X，且 $X \in V \cup T \cup \{\varepsilon\}$；
  - 树根的标记为S；
  - 如果非叶子顶点v标记为A，v的儿子从左到右依次为v_1，v_2，…，v_n，并且它们分别标记为X_1，X_2，…，X_n，则$A \rightarrow X_1X_2\ldots X_n \in P$；
  - 如果X是一个非叶子顶点的标记，则$X \in V$；
  - 如果顶点v标记为ε，则v是该树的叶子，并且v是其父顶点的惟一儿子。

上下文无关文法的派生树

- 别称
  - 生成树
  - 分析树(parse tree)
  - 语法树(syntax tree)
- 顺序
  - v_1，v_2是派生树T的两个不同顶点，如果存在顶点v，v至少有两个儿子，使得v_1是v的较左儿子的后代，v_2是v的较右儿子的后代，则称顶点v_1在顶点v_2的左边，顶点v_2在顶点v_1的右边。

上下文无关文法的派生树

- **结果(yield)**
- 派生树T的所有叶子顶点从左到右依次标记为 X_1，X_2，…，X_n，则称符号串$X_1X_2…X_n$是T的结果。
- 一个文法可以有多棵派生树，它们可以有不同的结果。
- **句型α的派生树**：“G的结果为α的派生树”。

上下文无关文法的派生树

- **派生子树(subtree)**
- 满足派生树定义中除了对根结点的标记的要求以外各条的树叫**派生子树(subtree)**。
- 如果这个子树的根标记为**A**，则称之为**A**子树。
- 惟一差别是根结点可以标记非开始符号。

上下文无关文法的派生树

定理 设**CFG G=(V，T，P，S)，S**$\Rightarrow^*$α 的充分必要条件为**G**有一棵结果为α 的派生树。

证明：

- 证一个更为一般的结论：对于任意**A**∈**V**，**A**$\Rightarrow^*$α 的充分必要条件为**G**有一棵结果为α 的**A-**子树。
- 充分性：设**G**有一棵结果为α 的**A-**子树，非叶顶点的个数**n**施归纳，证明**A**$\Rightarrow^*$α 成立。

上下文无关文法的派生树

- 设**A**-子树有**k+1**个非叶子顶点，根顶点**A**的儿子从左到右依次为v_1，v_2，…，v_m，并且它们分别标记为X_1，X_2，…，X_m。
- $A \to X_1X_2 \ldots X_m \in P$。
- 以X_1，X_2，…，X_m为根的子树的结果依次为α_1，α_2，…，α_m。
- X_1，X_2，…，X_m为根的子树的非叶子顶点的个数均不大于**k**。

上下文无关文法的派生树

$X_1 \Rightarrow^* \alpha_1$

$X_2 \Rightarrow^* \alpha_2$

…

$X_m \Rightarrow^* \alpha_m$

而且

$\alpha = \alpha_1 \alpha_2 \ldots \alpha_m$

上下文无关文法的派生树

$A \Rightarrow X_1 X_2 \ldots X_m$

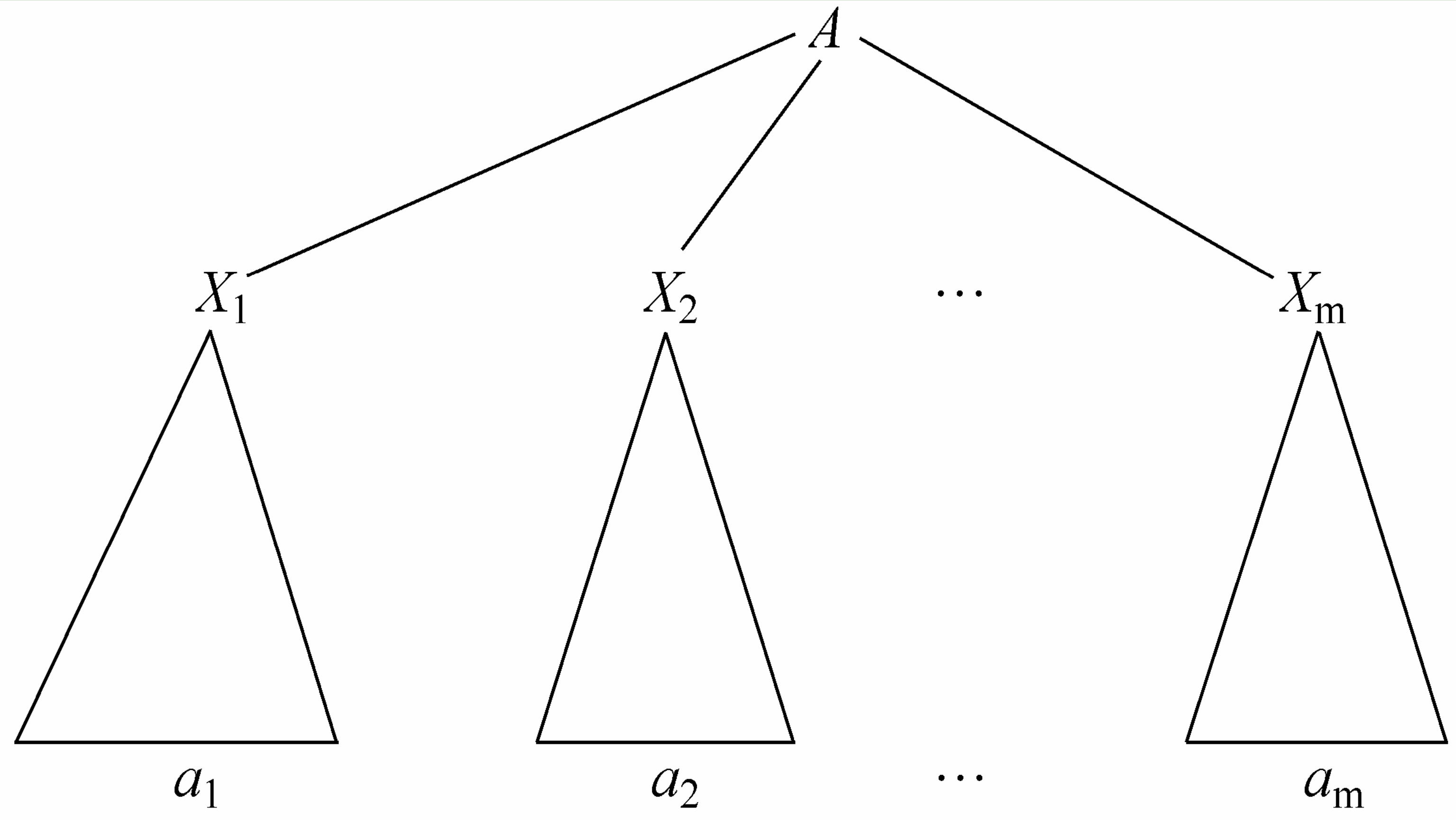
$\Rightarrow^* \alpha_1 X_2 \ldots X_m$

$\Rightarrow^* \alpha_1 \alpha_2 \ldots X_m$

…

$\Rightarrow^* \alpha_1 \alpha_2 \ldots \alpha_m$

上下文无关文法的派生树



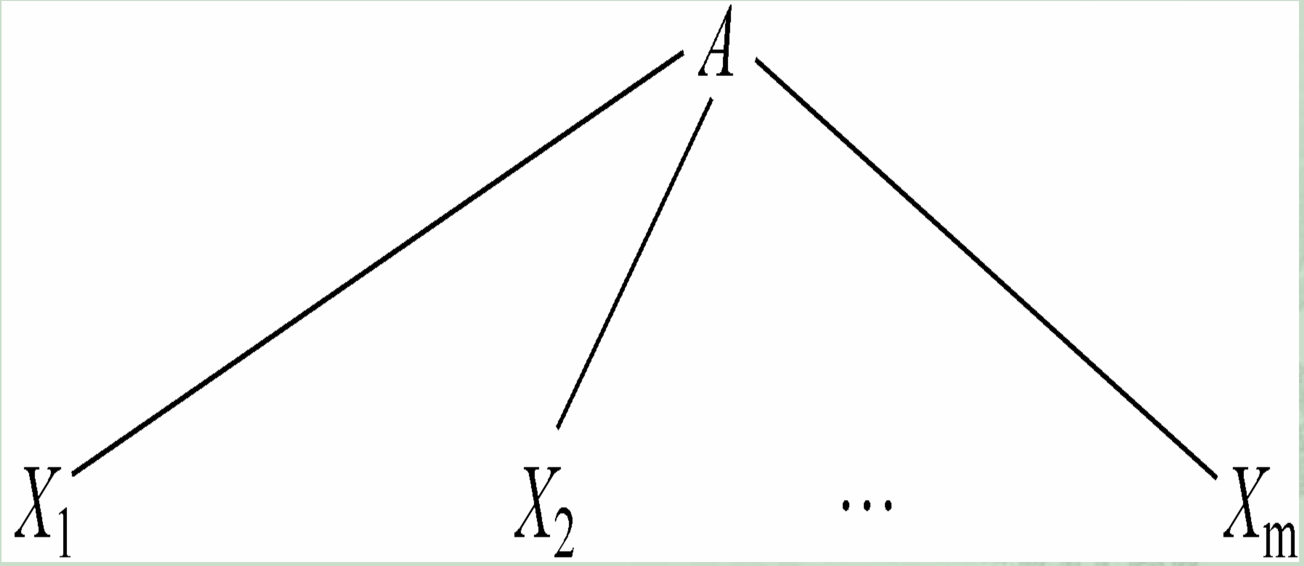
上下文无关文法的派生树

- 必要性
  - 设$A \Rightarrow^n \alpha$，现施归纳于派生步数n，证明存在结果为α的A-子树。

设n≤k(k≥1)时结论成立，往证当n=k+1时结论也成立：令$A \Rightarrow^{k+1} \alpha$，则有：

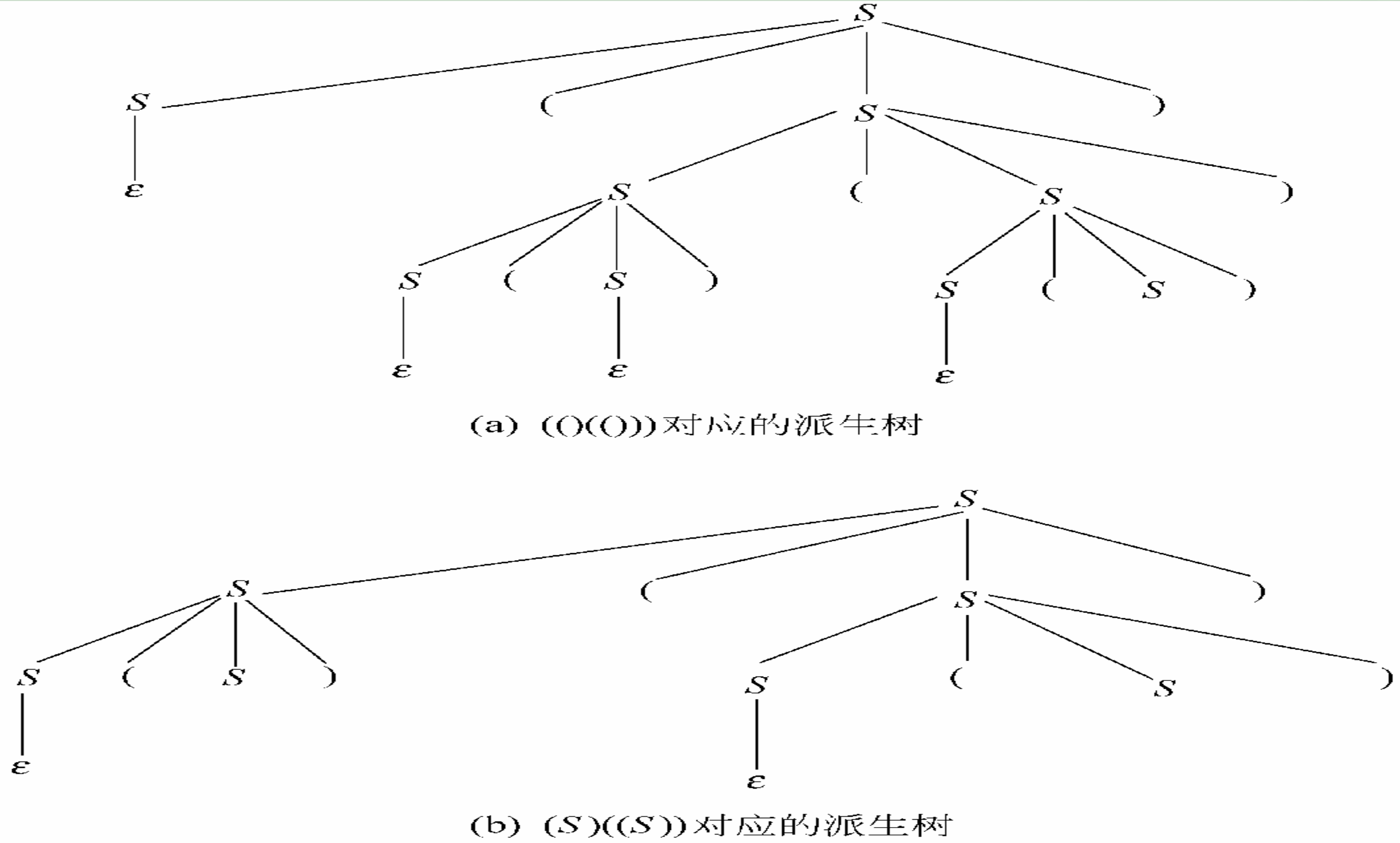
$$
\begin{aligned}
A &\Rightarrow X_1X_2\ldots X_m \\
&\Rightarrow^* \alpha_1X_2\ldots X_m \\
&\Rightarrow^* \alpha_1\alpha_2\ldots X_m \\
&\ldots \\
&\Rightarrow^* \alpha_1\alpha_2\ldots\alpha_m
\end{aligned}
$$

上下文无关文法的派生树



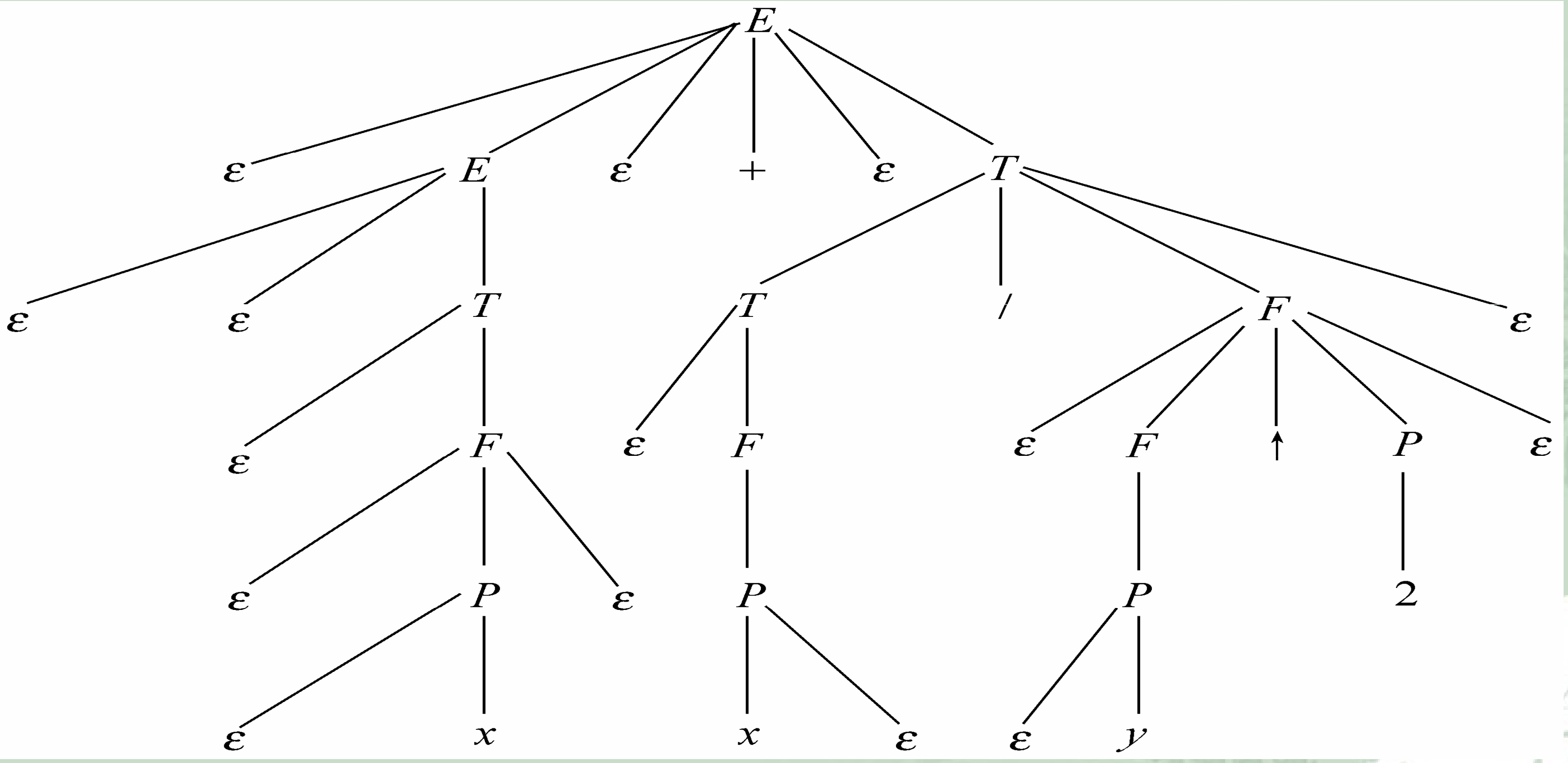
上下文无关文法的派生树

- 例 设G_{bra}：$S \to S(S)|\varepsilon$，(()(()))和(S)((S))的派生树。



上下文无关文法的派生树

- 关于标记 ε 的结点



上下文无关文法的派生树

- **最左派生(leftmost derivation)**
  - α 的派生过程中，每一步都是对当前句型的最左变量进行替换。
- **左句型(left sentencial form)**
  - 最左派生得到的句型可叫做左句型。
- **最右归约**(rightmost reduction)
  - 与最左派生相应的归约叫做最右归约。

上下文无关文法的派生树

- **最右派生(rightmost derivation)**
  - α 的派生过程中，每一步都是对当前句型的最右变量进行替换。
- **右句型(right sentencial form)**
  - 最右派生得到的句型可叫做右句型。
- **最左归约（leftmost reduction）**
  - 与最右派生相应的归约叫做最左归约。

上下文无关文法的派生树

定理 如果α是**CFG G**的一个句子，则**G**中存在α的最左派生和最右派生。

证明：

基本思路：对派生的步数**n**施归纳，证明对于任意**A**∈**V**，如果$\mathbf{A} \Rightarrow^{\mathbf{n}} \alpha$，在**G**中，存在对应的从**A**到α的最左派生：$\mathbf{A} \Rightarrow^{\mathbf{n}左} \alpha$。

上下文无关文法的派生树

$A \Rightarrow X_1X_2\ldots X_m$

$\Rightarrow^* \alpha_1X_2\ldots X_m$

$\Rightarrow^* \alpha_1\alpha_2\ldots X_m$

$\ldots$

$\Rightarrow^* \alpha_1\alpha_2\ldots\alpha_m$

$A \Rightarrow^{左} X_1X_2\ldots X_m$

$\Rightarrow^{*左} \alpha_1X_2\ldots X_m$

$\Rightarrow^{*左} \alpha_1\alpha_2\ldots X_m$

$\ldots$

$\Rightarrow^{*左} \alpha_1\alpha_2\ldots\alpha_m$

同理可证，句子α有最右派生。

上下文无关文法的派生树

定理 如果α是**CFG G**的一个句子，α的派生树与最左派生和最右派生是一一对应的，但是，这棵派生树可以对应多个不同的派生。

二义性

- 简单算术表达式的二义性文法

G_{exp2}:

E→E+E|E-E|E/E|E*E

E→ E↑E|(E)|N(L)|id

N→sin|cos|exp|abs|log|int

L→L,E|E

二义性

句子$x+x/y\uparrow 2$在文法中的三个不同的最左派生

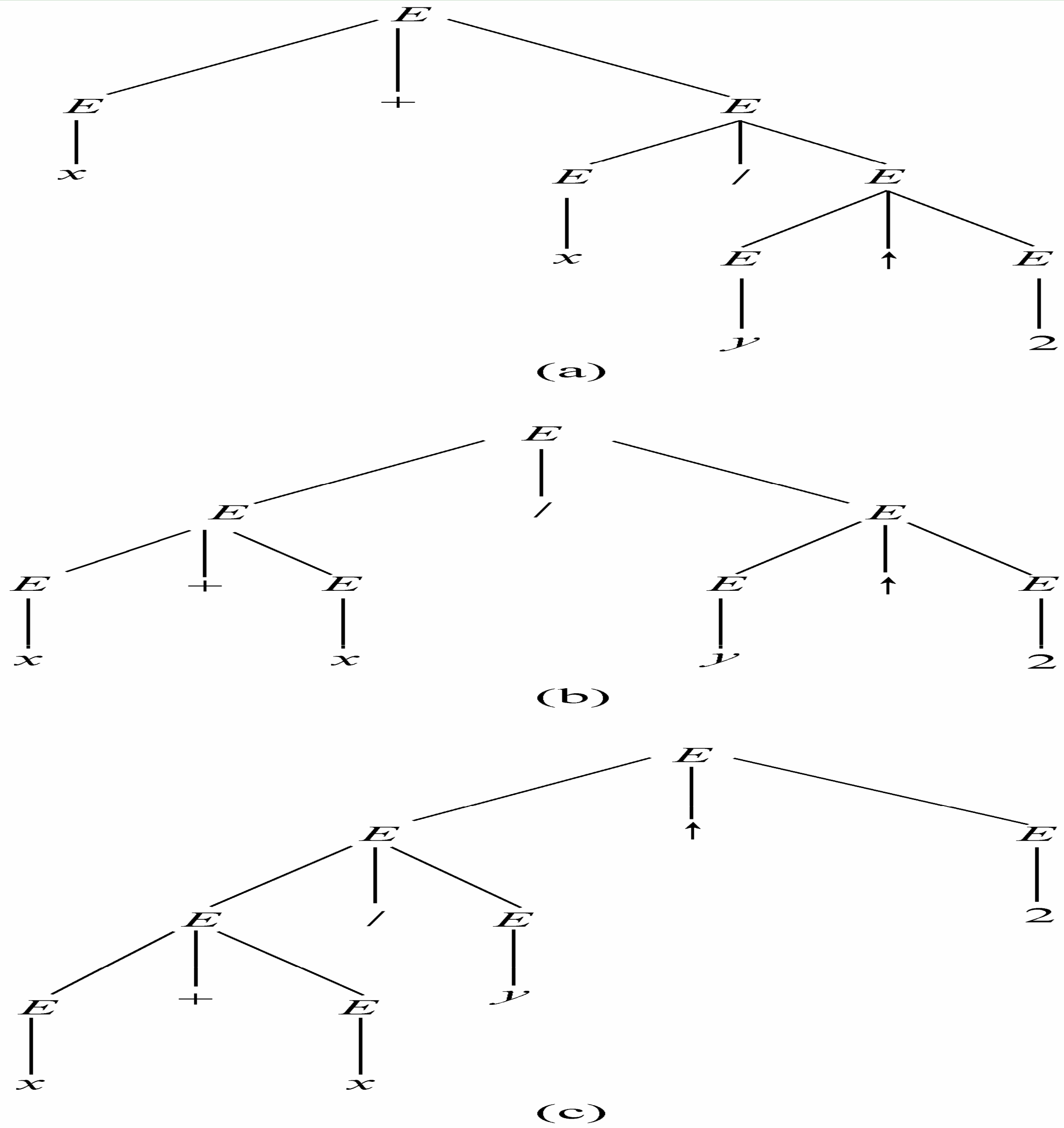
$E \Rightarrow E+E$
$\Rightarrow x+E$
$\Rightarrow x+E/E$
$\Rightarrow x+x/E$
$\Rightarrow x+x/E\uparrow E$
$\Rightarrow x+x/y\uparrow E$
$\Rightarrow x+x/y\uparrow 2$

$E \Rightarrow E/E$
$\Rightarrow E+E/E$
$\Rightarrow x+E/E$
$\Rightarrow x+x/E$
$\Rightarrow x+x/E\uparrow E$
$\Rightarrow x+x/y\uparrow E$
$\Rightarrow x+x/y\uparrow 2$

$E \Rightarrow E\uparrow E$
$\Rightarrow E/E\uparrow E$
$\Rightarrow E+E/E\uparrow E$
$\Rightarrow x+E/E\uparrow E$
$\Rightarrow x+x/E\uparrow E$
$\Rightarrow x+x/y\uparrow E$
$\Rightarrow x+x/y\uparrow 2$

二义性

对应3个不同的语法树



二义性

- **二义性(ambiguity)**
- CFG G=(V，T，P，S)，如果存在w∈L(G)，w至少有两棵不同的派生树，则称G是**二义性的**。否则，G为非二义性的。
- 判定任给CFG G是否为二义性的问题是一个**不可解的**(unsolvable)**问题**。

二义性

- 例 设

$L_{ambiguity}=\{0^n1^n2^m3^m|n,m\geqslant 1\}\cup\{0^n1^m2^m3^n|n,m\geqslant 1\}$

可以用如下文法产生语言$L_{ambiguity}$：

G：$S\rightarrow AB|0C3$

$A\rightarrow 01|0A1$

$B\rightarrow 23|2B3$

$C\rightarrow 0C3|12|1D2$

$D\rightarrow 12|1D2$

语言$L_{ambiguity}$不存在非二义性的文法。

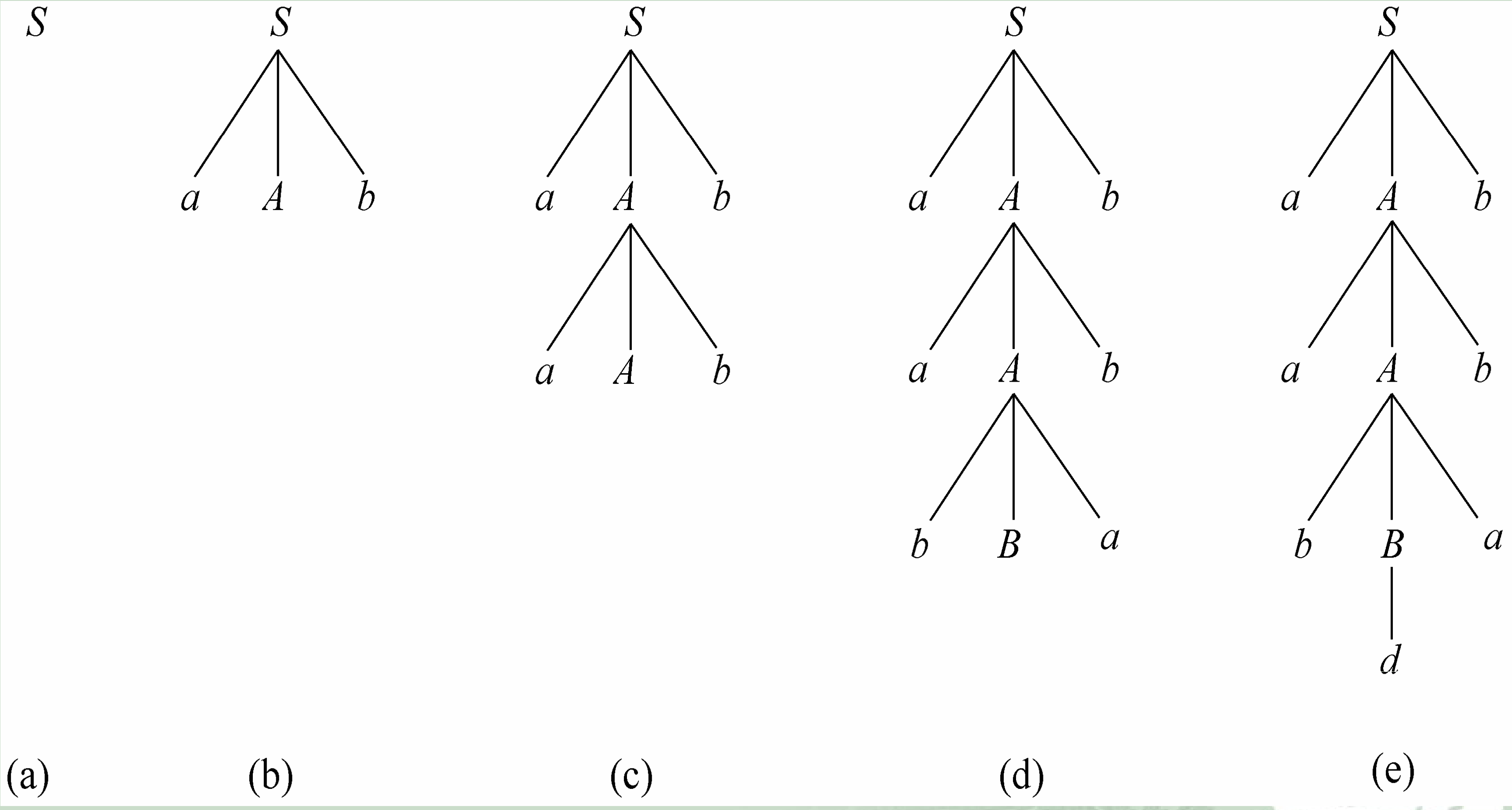
二义性

- **固有二义性的(inherent ambiguity)**
- 如果语言L不存在非二义性文法，则称L是**固有二义性的**，又称L是**先天二义性的**。
- 文法可以是二义性的。
- 语言可以是固有二义性的。

自顶向下的分析和自底向上的分析

- 自顶向下的分析方法
  - 通过考察是否可以从给定文法的开始符号派生出一个符号串，可以判定一个符号串是否为该文法的句子。
- 例
  - **S→aAb|bBa**
  - **A→aAb|bBa**
  - **B→d**

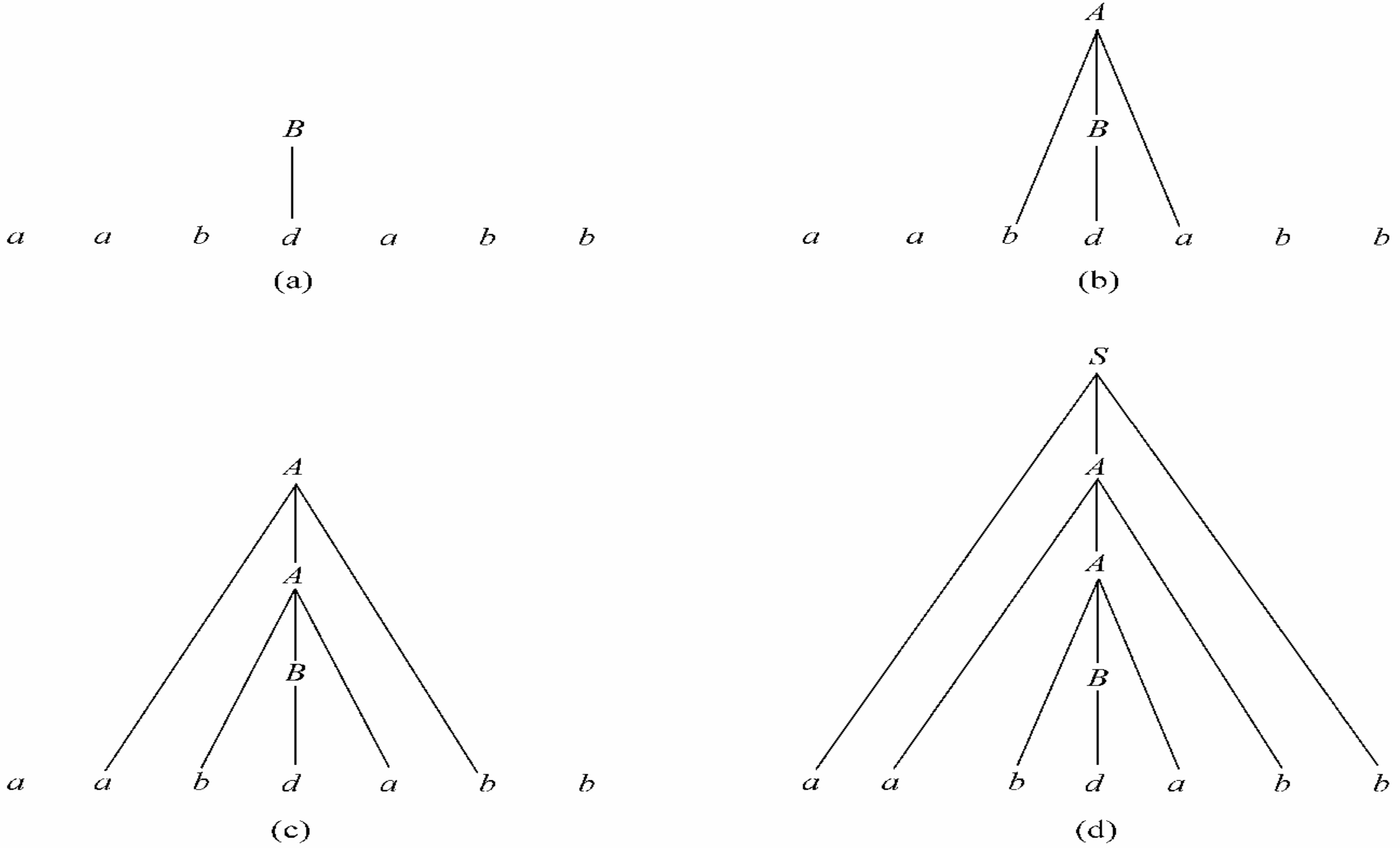
aabdabb的派生树的自顶向下的“生长”过程



自顶向下的分析和自底向上的分析

- **自底向上**的分析方法
  - 通过考察是否可以将一个符号串归约为给定文法的开始符号，完成判定一个符号串是否为该文法的句子的任务。
- 和归约与派生是互逆过程相对应，自顶向下的分析与自底向上的分析是互逆的分析过程。

aabdabb的派生树的自底向上的“生长”过程



(a) (b) (c) (d)

上下文无关文法的化简

- 文法

G_1： $S \rightarrow 0|0A|E$

$A \rightarrow \varepsilon\ |0A|1A|B$

$B \rightarrow \_C$

$C \rightarrow 0|1|0C|1C$

$D \rightarrow 1|1D|2D$

$E \rightarrow 0E2|E02$

- 去掉无用“东西”后的文法

G_2： $S \rightarrow 0|0A$

$A \rightarrow \varepsilon\ |0A|1A|B$

$B \rightarrow \_C$

$C \rightarrow 0|1|0C|1C$

上下文无关文法的化简

- 去掉产生式A→ε 后的文法

G$_3$：S→0|0A

A→0|1|0A|1A|B

B→_C

C→0|1|0C|1C

- 去掉产生式A→B后的文法

G$_4$：S→0|0A

A→0|1|0A|1A| _C

C→0|1|0C|1C

- 可以去掉文法中的无用符号、ε 产生式和单一产生式。

去无用符号

- **无用符号(useless symbol)**
  - 对于任意X∈V∪T，如果存在w∈L(G)，X出现在w的派生过程中，即存在α，β∈(V∪T)*，使得$S \Rightarrow^* \alpha X \beta \Rightarrow^* w$，则称X是有用的，否则，称X是**无用符号**。
- 对**CFG G=(V，T，P，S)**

(1) **G**中的符号X既可能是有用的，也可能是无用的。当X是无用的时候，它既可能是终极符号，也可能是语法变量。

去无用符号

(2) 对于任意$X \in V \cup T$，如果X是有用的，它必须同时满足如下两个条件：

① 存在$w \in T^*$，使得$X \Rightarrow^* w$；

② 存在α，$\beta \in (V \cup T)^*$，使得$S \Rightarrow^* \alpha X \beta$。

(3) 注意到文法是语言的有穷描述，所以，集合V，T，P都是有穷的。从而我们有可能构造出有效的算法，来完成消除文法的无用符号的工作。

去无用符号

算法1 删除派生不出终极符号行的变量。

- 输入：**CFG G=(V，T，P，S)**。
- 输出：**CFG G′=(V′，T，P′，S)，V′**中不含派生不出终极符号行的变量，并且**L(G′)=L(G)**。
- 主要步骤：

去无用符号

(1) OLDV= Φ;

(2) NEWV={A|A→w∈P且w∈T^*};

(3) while OLDV≠NEWV do

begin

(4) OLDV=NEWV;

(5) NEWV=OLDV∪{A|A→α∈P 且α∈$(T\cup OLDV)^*$}

end

(6) V′=NEWV;

(7) P′={ A→α | A→α∈P 且 A∈V′ 且 α∈$(T\cup V')^*$}

去无用符号

- 第(3)条语句控制对NEWV进行迭代更新。第一次循环将那些恰经过两步可以派生出终极符号行的变量放入NEWV；第二次循环将那些恰经过三步和某些至少经过三步可以派生出终极符号行的变量放入NEWV；……，第n次循环将那些恰经过n步和某些至少经过n+1步可以派生出终极符号行的变量放入NEWV。这个循环一直进行下去，直到所给文法G中的所有可以派生出终极符号行的变量都被放入NEWV中。

去无用符号

定理 算法1是正确的。

证明要点：首先证明对于任意$A \in V$，A被放入V'中的充要条件是存在$w \in T$，$A \Rightarrow^n w$。再证所构造出的文法是等价的。

(1)对A被放入NEWV的循环次数n施归纳，证明必存在$w \in T$，满足$A \Rightarrow^+ w$。

去无用符号

(2)施归纳于派生步数n，证明如果$A \Rightarrow^n w$，则A被算法放入到NEWV中。实际上，可以证明，A是在第n次循环前被放入到NEWV中的。

(3)证明L(G′)=L(G)。显然有$L(G') \subseteq L(G)$，所以只需证明$L(G) \subseteq L(G')$。

去无用符号

算法2 删除不出现在任何句型中的语法符号。

- 输入：**CFG G=(V，T，P，S)**。
- 输出：**CFG G′=(V′，T′，P′，S)，V′∪T′** 中的符号必在**G**的某个句型中出现，并且有**L(G′)=L(G)**。
- 主要步骤：

去无用符号

- 主要步骤:

(1) OLDV= Φ;

(2) OLDT= Φ;

(3) NEWV=$\{S\} \cup \{A | S \to \alpha A \beta \in P\}$;

(4) NEWT=$\{a | S \to \alpha a \beta \in P\}$;

去无用符号

(5) while OLDV≠NEWV 或者OLDT≠NEWT do

begin

(6) OLDV=NEWV;

(7) OLDT=NEWT;

(8) NEWV=OLDV∪{B| A∈OLDV且 A→αBβ∈P 且B∈V};

(9) NEWT=OLDT∪{a| A∈OLDV且 A→αaβ∈P 且a∈T};

end

去无用符号

(10)　　V′ =NEWV;

(11)　　T′ =NEWT;

(12)　　P′ ={ A→α | A→α ∈P且 A∈V′ 且 α ∈(T′ ∪V′)*}。

去无用符号

定理 算法2是正确的。

证明要点：

(1) 施归纳于派生步数**n**，证明如果$S \Rightarrow^n \alpha X \beta$，则当$X \in V$时，X在算法中被语句(3)或者语句(8)放入NEWV；当$X \in T$时，它在算法中被语句(4)或者语句(9)放入NEWT。

(2) 对循环次数**n**施归纳，证明如果X被放入NEWT或者NEWV中，则必定存在α，$\beta \in (NEWV \cup NEWT)^*$，使得$S \Rightarrow^n \alpha X \beta$。

(3) 证明$L(G') = L(G)$。

去无用符号

定理 对于任意**CFL L**，**L**$\neq$ Φ，则存在不含无用符号的**CFG G**，使得**L(G)=L**。

- 证明要点：
- 依次用算法**1**和算法**2**对文法进行处理，可以得到等价的不含无用符号的文法。
- 不可先用算法**2**后用算法**1**。

去无用符号

例 设有如下文法

$S \to AB|a|BB$，$A \to a$，$C \to b|ABa$

先用算法2，文法被化简成：

$S \to AB|a|BB$，$A \to a$

再用算法1，可得到文法：

$S \to a$，$A \to a$

显然，该文法中的变量A是新的无用变量。

去ε-产生式

- ε-产生式（ε-production）
  - 形如A→ε的产生式叫做ε-产生式。
  - ε-产生式又称为空产生式（null production。
- 可空(nullable)变量
  - 对于文法G=(V，T，P，S)中的任意变量A，如果$A \Rightarrow^{+} \varepsilon$，则称A为可空变量。

去ε-产生式

- 对形如$A \rightarrow X_1X_2\ldots X_m$的产生式进行考察，找出文法的可空变量集U，然后对于$\forall H \subseteq U$，从产生式$A \rightarrow X_1X_2\ldots X_m$中删除H中的变量。对于不同的H，得到不同的A产生式，用这组A产生式替代产生式$A \rightarrow X_1X_2\ldots X_m$。
- 必须避免在这个过程中产生新的ε-产生式：当$\{X_1，X_2，\ldots，X_m\} \subseteq U$时，不可将$X_1$，$X_2$，…，$X_m$同时从产生式$A \rightarrow X_1X_2\ldots X_m$中删除。

去ε-产生式

算法3 求CFG G的可空变量集U。

- 输入：CFG G=(V，T，P，S)。
- 输出：G的可空变量集U。
- 主要步骤：

(1) OLDU= Φ**;**

(2) NEWU={A| A→ε ∈P};

去ε-产生式

(3) while NEWU$\neq$OLDU do

begin

(4) OLDU = NEWU;

(5) NEWU= OLDU $\cup$ {A|A$\rightarrow\alpha\in$P并且 $\alpha\in$OLDU*}

end

(6) U=NEWU

去ε-产生式

定理 对于任意CFG G，存在不含ε-产生式的CFG G′使得L(G′)=L(G)-{ε}。

证明：

(1) 构造

- 设CFG G=(V，T，P，S)，
- 用算法3求出G的可空变量集U,
- 构造P′。

去ε-产生式

- 对于 $\forall A \rightarrow X_1X_2\ldots X_m \in P$
- 将$A \rightarrow \alpha_1\alpha_2\ldots\alpha_m$放入P′，其中
- if $X_i \in U$ then $\alpha_i = X_i$或者$\alpha_i = \varepsilon$；
- if $X_i \notin U$ then $\alpha_i = X_i$
- 要求：在同一产生式中，α_1，α_2，…，α_m不能同时为ε。

去ε-产生式

- 证明对于任意$w \in T^+$，$A \Rightarrow^n_G w$的充分必要条件是$A \Rightarrow^m_{G'} w$。
- 必要性：设$A \Rightarrow^n_G w$，施归纳于n，证明$A \Rightarrow^m_{G'} w$成立。
- 当n=1时，由$A \Rightarrow_G w$知，$A \rightarrow w \in P$，按照定理所给的构造G′的方法，必定有$A \rightarrow w \in P'$。所以，$A \Rightarrow_{G'} w$成立。

去ε-产生式

- 设$n \leqslant k$时结论成立($k \geqslant 1$)，当$n=k+1$时

$A \Rightarrow X_1X_2\ldots X_m$

$\Rightarrow^*_G w_1X_2\ldots X_m$

$\Rightarrow^*_G w_1w_2\ldots X_m$

…

$\Rightarrow^*_G w_1w_2\ldots w_m$

其中$w_1w_2\ldots w_m=w$，且w_1，w_2，……，$w_m \in T^*$。

去ε-产生式

注意到$w \neq \varepsilon$，必存在$1 \leqslant i \leqslant m$，$w_i \neq \varepsilon$，设i，j，…，k是$w_1$，$w_2$，…，$w_m$中所有非空串的下标，并且$1 \leqslant i \leqslant j \leqslant \ldots \leqslant k \leqslant m$，即：

$$w = w_i w_j \ldots w_k$$

按照G′的构造方法，$A \rightarrow X_i X_j \ldots X_k \in P'$

再由归纳假设，

$X_i \Rightarrow^*_{G'} w_i$，$X_j \Rightarrow^*_{G'} w_j$，…，$X_k \Rightarrow^*_{G'} w_k$。

去ε-产生式

$$A \Rightarrow^*_{G'} X_i X_j \ldots X_k$$
$$\Rightarrow^*_{G'} w_i X_j \ldots X_k$$
$$\Rightarrow^*_{G'} w_i w_j \ldots X_k$$
$$\ldots$$
$$\Rightarrow^*_{G'} w_i w_j \ldots w_k$$

所以，结论对n=k+1成立。由归纳法原理，结论对所有的n成立。

去ε-产生式

充分性：设$A \Rightarrow^m_{G'} w$，施归纳于m，证明$A \Rightarrow^n_G w$成立。

当m=1时，由$A \Rightarrow_{G'} w$知，$A \rightarrow w \in P'$，按照定理所给的构造G′的方法，必定有$A \rightarrow \alpha \in P$。$A \rightarrow w$是通过删除产生式$A \rightarrow \alpha$右部中的可空变量而构造出来的，所以，

$A \Rightarrow_G \alpha \Rightarrow^*_G w$ 成立。

去ε-产生式

设n≤k时结论成立(k≥1)，当m=k+1时

$A \Rightarrow^*_{G'} X_i X_j \ldots X_k$

$\Rightarrow^*_{G'} w_i X_j \ldots X_k$

$\Rightarrow^*_{G'} w_i w_j \ldots X_k$

…

$\Rightarrow^*_{G'} w_i w_j \ldots w_k = w$

其中$X_i \Rightarrow^*_{G'} w_i$，$X_j \Rightarrow^*_{G'} w_j$，…，$X_k \Rightarrow^*_{G'} w_k$。

去 ε -产生式

表明$A\rightarrow X_iX_j\ldots X_k \in P'$。按照$G'$的构造方法，必定存在$A\rightarrow X_1X_2\ldots X_m \in P$，而且

$\{X_i, X_j, \ldots, X_k\} \subseteq \{X_1, X_2, \ldots, X_m\}$

$\{X_1, X_2, \ldots, X_m\} - \{X_i, X_j, \ldots, X_k\} \subseteq U$

从而，

$$A \Rightarrow_G X_1X_2\ldots X_m$$

$$\Rightarrow^*_G X_iX_j\ldots X_k$$

去ε-产生式

再根据$X_i \Rightarrow^*_{G'}\ w_i$，$X_j \Rightarrow^*_{G'}\ w_j$，…，$X_k \Rightarrow^*_{G'}\ w_k$和归纳假设，有

$X_i \Rightarrow^*_G w_i$，$X_j \Rightarrow^*_G w_j$，…，$X_k \Rightarrow^*_G w_k$。

这表明，如下派生成立：

$$
\begin{aligned}
A &\Rightarrow_G X_1X_2\ldots X_m \\
&\Rightarrow^*_G X_iX_j\ldots X_k \\
&\Rightarrow^*_G w_iX_j\ldots X_k \\
&\Rightarrow^*_G w_iw_j\ldots X_k \\
&\ldots \\
&\Rightarrow^*_G w_iw_j\ldots w_k = w
\end{aligned}
$$

去ε-产生式

表明结论对**m=k+1**成立。由归纳法原理，结论对任意**m**成立。

注意到**A** 的任意性，当**S=A**时结论成立。即：

$S\Rightarrow^{*}_{G} w$ 的充分必要条件是$S\Rightarrow^{*}_{G'} w$

亦即：$L(G')=L(G)-\{\varepsilon\}$。

定理得证。

去单一产生式

文法G_{exp1}： $E \to E+T|E-T|T$

$T \to T*F|T/F|F$

$F \to F \uparrow P|P$

$P \to (E)|N(L)|id$

$N \to sin|cos|exp|abs|log|int$

$L \to L,E|E$

存在派生：

$E \Rightarrow T \Rightarrow F \Rightarrow P \Rightarrow id$

去单一产生式

G_{exp3}： E→E+T|E-T|T*F|T/F|F↑P|(E)|N(L)|id

T→T*F|T/F| F↑P| (E)|N(L)|id

F→F↑P| (E)|N(L)|id

P→(E)|N(L)|id

N→sin|cos|exp|abs|log|int

L→L,E|E+T|E-T|T*F|T/F|F↑P|(E)|N(L)|id

- 该文法中不存在类似的派生。

去单一产生式

- **单一产生式(unit production)**
- 形如**A→B**的产生式称为**单一产生式**。
- **定理** 对于任意**CFG G**，ε $\notin$**L(G)**，存在等价的**CFG** G_1，G_1不含无用符号、ε **-**产生式和单一产生式。
- 满足本定理的**CFG**为化简过的文法。
- 已有去无用符号和去ε **-**产生式的结论，所以只讨论去单一产生式的问题。

去单一产生式

- 证明要点：

（1）构造G_2，满足$L(G_2)=L(G)$，并且G_2中不含单一产生式。

用非单一产生式$A_1 \to \alpha$ 取代$A_1 \Rightarrow^*_G A_n \Rightarrow \alpha$ 用到的产生式系列 $A_1 \to A_2$，$A_2 \to A_3$，…，$A_{n-1} \to A_n$，$A_n \to \alpha$ 。其中，$A_1 \to A_2$，$A_2 \to A_3$，…，$A_{n-1} \to A_n$都是单一产生式。（$n \geqslant 1$）

去单一产生式

（2） 证明$L(G_2)=L(G)$。

用$A_1\rightarrow\alpha$ 所完成的派生$A_1\Rightarrow\alpha$ 与产生式系列 $A_1\rightarrow A_2$， $A_2\rightarrow A_3$， ……， $A_{n-1}\rightarrow A_n$， $A_n\rightarrow\alpha$ 所完成的派生$A_1\Rightarrow^*_G A_n\Rightarrow\alpha$ 相对应。

在原文法中可能会出现一个变量在派生过程中循环出现的情况，在$w\in L(G)$ 证明$w \in L(G_2)$ 的过程中，要取w在G中的一个最短的最左派生。

$$S=\alpha_0\Rightarrow_G\alpha_1\Rightarrow_G\alpha_2\Rightarrow_G\cdots\Rightarrow_G\alpha_n=w$$

去单一产生式

（**3**）删除$\mathbf{G_2}$中的无用符号。

由于在删除单一产生式后，文法中可能出现新的无用符号，因此，我们还需要再次删除新出现的无用符号。

此外，在去ε-产生式后可能会产生新的单一产生式，也可能会引进新的无用符号。这是值得注意的。

乔姆斯基范式

- 如果**CFG G=(V，T，P，S)**中的产生式都具有形式：**A→BC**，**A→a**，其中：**A**、**B**、**C**∈**V**，**a**∈**T**，则称**G**为**乔姆斯基范式文法 (Chomsky normal form，CNF)**，简称为**Chomsky文法**，或**Chomsky范式**。
- 显然，**CNF**中不允许有ε-产生式、单一产生式。

乔姆斯基范式

- **例** 试将文法G_{exp4}转换成等价的 **CNF**。

G_{exp4}：

$E \rightarrow E+T \mid T*F \mid F\uparrow P \mid (E) \mid id$

$T \rightarrow T*F \mid F\uparrow P \mid (E) \mid id$

$F \rightarrow F\uparrow P \mid (E) \mid id$

$P \rightarrow (E) \mid id$

- 可以分两步走
  - 变成$A \rightarrow a$和$A \rightarrow A_1A_2\ldots A_n$的形式。
  - 变成**CNF**。

第一步

$E \rightarrow EA_{+}T|\ T\ A_{*}F|F\ A_{\uparrow}P|\ A_{(}EA_{)}|\ id$

$T \rightarrow TA_{*}F|\ FA_{\uparrow}P|\ A_{(}EA_{)}\ |id$

$F \rightarrow FA_{\uparrow}P|\ A_{(}EA_{)}|id$

$P \rightarrow A_{(}EA_{)}\ |id$

$A_{+} \rightarrow +$

$A_{*} \rightarrow *$

$A_{\uparrow} \rightarrow \uparrow$

$A_{(} \rightarrow ($

$A_{)} \rightarrow)$

第二步

G_{expCNF}：

$E \to EA_1 | \ TA_2$

$E \to \ FA_3 | \ A_(A_4 | \ id$

$T \to TA_2 | \ FA_3 | \ A_(A_4 \ |id$

$F \to FA_3 | \ A_(A_4 |id$

$P \to A_(A_4 \ |id$

$A_+ \to +$

$A_* \to *$

$A_\uparrow \to \uparrow$

$A_(\to ($

$A_) \to)$

$A_1 \to A_+ T$

$A_2 \to A_* F$

$A_3 \to A_\uparrow P$

$A_4 \to EA_)$

乔姆斯基范式

定理 对于任意CFG G， $\varepsilon \notin L(G)$，存在等价的 CNF G_2。

证明要点：

1. 构造CNF

按照上述例子所描述的转换方法，在构造给定**CFG**的**CNF**时，可以分两步走。

假设**G**为化简过的文法

- 构造$G_1=(V_1, T, P_1, S)$，G_1中的产生式都是形如$A \rightarrow B_1B_2\ldots B_m$和$A \rightarrow a$的产生式，其中，$A$，$B_1$，$B_2$，…，$B_m \in V_1$，$a \in T$，$m \geq 2$

乔姆斯基范式

- 构造CNF G_2=(V_2，T，P_2，S)。
  m≥3时，引入新变量：B_1、B_2、…、B_{m-2}，将G_1的形如$A \rightarrow A_1A_2 \ldots A_m$的产生式替换成
  $A \rightarrow A_1B_1$
  $B_1 \rightarrow A_2B_2$
  …
  $B_{m-2} \rightarrow A_{m-1}A_m$

乔姆斯基范式

2. 构造的正确性证明。

- 按照上述构造，证明被替换的产生式是等价的。

  例如：在第二步中$\{A\rightarrow A_1B_1，B_1\rightarrow A_2B_2，\ldots，B_{m-2}\rightarrow A_{m-1}A_m\}$与$A\rightarrow A_1A_2\ldots A_m$等价。

乔姆斯基范式

- **例** 试将下列文法转换成等价的 **CNF。**

  $S \to bA \mid aB$

  $A \to bAA \mid aS \mid a$

  $B \to aBB \mid bS \mid b$

乔姆斯基范式

- 先引入变量B_a，B_b和产生式$B_a \to a$，$B_b \to b$ ，完成第一步变换。

$S \to B_b A \mid B_a B$

$A \to B_b AA \mid B_a S \mid a$

$B \to B_a BB \mid B_b S \mid b$

$B_a \to a$

$B_b \to b$

乔姆斯基范式

- 引入新变量B_1、B_2

  $S \rightarrow B_bA \mid B_aB$

  $A \rightarrow B_bB_1 \mid B_aS \mid a$

  $B \rightarrow B_aB_2 \mid B_bS \mid b$

  $B_a \rightarrow a$

  $B_b \rightarrow b$

  $B_1 \rightarrow AA$

  $B_2 \rightarrow BB$

乔姆斯基范式

- 不能因为原来有产生式$A\rightarrow a$和$B\rightarrow b$而放弃引进变量B_a、B_b和产生式$B_a\rightarrow a$、$B_b\rightarrow b$。
- $L(A)=\{x \mid x\in\{a，b\}^+$ & x中a的个数比b的个数恰多1个$\}$。
- $L(B)=\{x \mid x\in\{a，b\}^+$ & x中b的个数比a的个数恰多1个$\}$。
- $L(B_a)=\{ a \}$。
- $L(B_b)=\{ b \}$。

格雷巴赫范式

- 如果CFG G=(V，T，P，S)中的产生式都具有形式：A→aα，其中，A∈V，a∈T，α∈V*，则称G为**格雷巴赫范式文法(Greibach normal form ,GNF)**，简称为**Greibach文法**，或**Greibach范式**。
- 在GNF中，有如下两种形式的产生式
  - $A \to a$
  - $A \to aA_1A_2\ldots A_m$ (m≥1)

格雷巴赫范式

- 右线性文法是一种特殊的GNF。
- 由于GNF中不存ε-产生式，所以对任意的GNF G，ε $\notin$L(G)。
- 当ε $\notin$L(G′)时，能够找到一个GNF G，使得L(G)=L(G′)。
- 经过化简的CFG，都有一个等价的GNF。

格雷巴赫范式

引理 对于任意的CFG $G=(V, T, P, S)$，$A \to \alpha B \beta \in P$，且G中所有的B产生式为

$$B \to \gamma_1 | \gamma_2 | \ldots | \gamma_n$$

取$G_1=(V, T, P_1, S)$

$$P_1=(P-\{A \to \alpha B \beta\}) \cup \{A \to \alpha \gamma_1 \beta, A \to \alpha \gamma_2 \beta, \ldots, A \to \alpha \gamma_n \beta\}$$

则，$L(G_1)=L(G)$。

格雷巴赫范式

- 证明

  以下两组产生式等价

  - $A \to \alpha B \beta$ ； $B \to \gamma_1 | \gamma_2 | \ldots | \gamma_n$
  - $\{ A \to \alpha \gamma_1 \beta$ ， $A \to \alpha \gamma_2 \beta$ ， …， $A \to \alpha \gamma_n \beta \}$

格雷巴赫范式

- **递归(recursive)**
- 如果G中存在形如$A \Rightarrow^n \alpha A \beta$ 的派生，则称该派生是关于变量A**递归的**，简称为递归派生。
- 当n=1时，称该派生关于变量A**直接递归(directly recursive)**，简称为直接递归派生。
- 形如$A \to \alpha A \beta$ 的产生式是变量A的直接递归的**(directly recursive)**产生式。

格雷巴赫范式

- 当n≥2时，称该派生是关于变量A的**间接递归(indirectly recursive)**派生。简称为间接递归派生。
- 当α＝ε时，称相应的(直接/间接)递归为**(直接/间接)左递归(left-recursive)**；
- 当β＝ε时，称相应的(直接/间接)递归为**(直接/间接)右递归(right-recursive)**。

格雷巴赫范式

- **引理**　对于任意的CFG G=(V，T，P，S)，G中所有A的产生式

$$\begin{cases} A \to A\alpha_1 | A\alpha_2 | \ldots | A\alpha_n \\ A \to \beta_1 | \beta_2 | \ldots | \beta_m \end{cases}$$

可以被等价地替换为产生式组

$$\begin{cases} A \to \beta_1 | \beta_2 | \ldots | \beta_m \\ A \to \beta_1 B | \beta_2 B | \ldots | \beta_m B \\ B \to \alpha_1 | \alpha_2 | \ldots | \alpha_n \\ B \to \alpha_1 B | \alpha_2 B | \ldots | \alpha_n B \end{cases}$$

格雷巴赫范式

- 证明要点：
  用直接右递归取代原来的直接左递归。
  两组产生式产生的符号串都是

  $$\beta_k \alpha_{h_q} \alpha_{h_{q-1}} \ldots\ldots \alpha_{h_2} \alpha_{h_1}$$

  前者是先产生

  $$\alpha_{h_q} \alpha_{h_{q-1}} \ldots\ldots \alpha_{h_2} \alpha_{h_1}$$

  然后产生β$_{\mathbf{k}}$。

格雷巴赫范式

- 后者先产生β $_{\mathbf{k}}$。
- 然后产生

$$\alpha_{h_q}\alpha_{h_{q-1}}\ldots\ldots\alpha_{h_2}\alpha_{h_1}$$

格雷巴赫范式

$$A \Rightarrow A\alpha_{h_1}$$

$$\Rightarrow A\alpha_{h_2}\alpha_{h_1}$$

$$\cdots\cdots$$

$$\Rightarrow A\alpha_{h_{q-1}}..\alpha_{h_2}\alpha_{h_1}$$

$$\Rightarrow A\alpha_{h_q}\alpha_{h_{q-1}}..\alpha_{h_2}\alpha_{h_1}$$

$$\Rightarrow \beta_k\alpha_{h_q}\alpha_{h_{q-1}}..\alpha_{h_2}\alpha_{h_1}$$

$$A \Rightarrow \beta_k\alpha_{hq}B$$

$$\Rightarrow \beta_k\alpha_{h_q}\alpha_{h_{q-1}}B$$

$$\cdots\cdots$$

$$\Rightarrow \beta_k\alpha_{h_q}\alpha_{h_{q-1}}..\alpha_{h_2}B$$

$$\Rightarrow \beta_k\alpha_{h_q}\alpha_{h_{q-1}}..\alpha_{h_2}\alpha_{h_1}$$

格雷巴赫范式

定理 对于任意CFG G， $\varepsilon \notin L(G)$，存在等价的GNF G_1。

- 证明要点：

(1)将产生式都化成如下形式的产生式

$A \to A_1 A_2 \ldots A_m$

$A \to a A_1 A_2 \ldots A_{m-1}$

$A \to a$

其中，A，A_1，A_2，…，$A_m \in V_1$，$a \in T$，$m \geqslant 2$。

格雷巴赫范式

(2)将产生式变成如下形式的产生式

$A_i \to A_j\alpha$　　　　$i<j$

$A_i \to a\alpha$

$B_i \to \alpha$

其中，$V_2=V_1\cup\{B_1, B_2, \ldots, B_n\}$，$V_1\cap\{B_1, B_2, \ldots, B_n\}=\Phi$。

“B类变量”：$\{B_1, B_2, \ldots, B_n\}$是在文法的改造过程中引入的新变量。

V_1中的变量称为“A类变量”。

格雷巴赫范式

（3）根据引理，从编号较大的变量开始，逐步替换，使所有产生式满足GNF的要求：

1) for k=m-1 to 1 do

2) if $A_k \to A_{k+1}\beta \in P_2$ then

3) for 所有的A_{k+1}产生式$A_{k+1} \to \gamma$ do 将产生式$A_k \to \gamma\beta$放入P_3;

4) for k= 1 to n do

5) 用P_3中的产生式将所有的B_k产生式替换成满足GNF要求的形式。

自嵌套文法

- **自嵌套文法(self-embedding grammar)**

CFG G=(V，T，P，S)是化简后的文法，如果G中存在有形如$A\Rightarrow^+\alpha A\beta$的派生，则称G为**自嵌套文法**，其中α、β∈$(V\cup T)^+$。

- 自嵌套的文法描述的语言可以是正则语言
- 例如：

$$S\rightarrow 0S0|1S1|0S1|1S0|0S|1S|0|1$$

自嵌套文法

定理 非自嵌套的文法产生的语言是正则语言。

证明要点：

（1）将G化成GNF。

（2）取RG $G'=(V',T,P',[S])$，其中

$V'=\{[\alpha] \mid \alpha\in V^+ 并且 |\alpha|\leqslant m(n-2)+1\}$

$P'=\{[A\alpha]\rightarrow a[\beta\alpha] \mid A\rightarrow a\beta\in P 并且 \beta\in V^*\}$

当 $\alpha=\varepsilon$ 时，$[\alpha]=\varepsilon$。

小结

本章讨论了CFG的派生树，A子树，最左派生与最右派生，派生与派生树的关系，二义性文法与固有二义性语言，句子的自顶向下分析和自底向上分析；无用符号的消去算法，空产生式的消除，单一产生式的消除。CFG的CNF和GNF；CFG的自嵌套特性。

(1)$S\Rightarrow^{*}\alpha$ 的充分必要条件为G有一棵结果为α的派生树。

(2)如果α是CFG G的一个句子，则G中存在α的最左派生和最右派生。

小结

(3)文法可能是二义性的，但语言只可能是固有二义性的，且这种语言是存在的。

(4)对于任意CFG G，$\varepsilon \notin L(G)$，存在等价的CFG G_1，G_1不含无用符号、ε-产生式和单一产生式。

(5)对于任意CFG G，$\varepsilon \notin L(G)$，存在等价的 CNF G_2。

(6)对于任意CFG G，$\varepsilon \notin L(G)$，存在等价的GNF G_3。

(7)非自嵌套的文法产生的语言是正则语言。

习题

1 构造产生下列语言的CFG。

(1) $\{a^n b^{2m} a^n \mid m, n \geqslant 1\}$。

(2)含有相同个数0和1的所有0、1串。

习题

2 给定文法G_{exp1}：

$$E \rightarrow E+T \mid E-T \mid T$$

$$T \rightarrow T*F \mid T/F \mid F$$

$$F \rightarrow F \uparrow P \mid P$$

$$P \rightarrow (E) \mid N(L) \mid id$$

$$N \rightarrow sin \mid cos \mid exp \mid abs \mid log \mid int$$

$$L \rightarrow L, E \mid E$$

请给出 x+ysin(x↑y)-cos(abs(x-y)/x/y) 的最左派生、最右派生、最右归约，并画出相应的派生树。

习题

3 删除下列文法中的无用符号。

$S \rightarrow AB \mid CA \mid a$

$A \rightarrow a$

$B \rightarrow BC \mid AB \mid DE \mid d$

$C \rightarrow aB \mid b$

习题

4 删除下列文法中的ε-产生式。

$S \to ABCDE \mid aB \mid \varepsilon$

$A \to aBCA \mid BC \mid \varepsilon$

$B \to b \mid bB \mid \varepsilon$

$C \to c \mid cC \mid \varepsilon$

$D \to d \mid dD \mid \varepsilon$

$E \to e \mid eE \mid \varepsilon$

5 消除上题中所给文法在消除ε-产生式后出现的单一产生式。

习题

6 构造与下列文法等价的CNF。

$S \to aBB \mid bAA$

$B \to aBa \mid aa \mid \varepsilon$

$A \to bbA \mid \varepsilon$