形式语言

与自动机及程序设计

陆　玲　周书民◎著

HEUP 哈尔滨工程大学出版社

形式语言与自动机及程序设计

陆　玲　周书民　著

HEUP 哈尔滨工程大学出版社

内容简介

本书介绍了乔姆斯文法体系的四类文法及相应的实例，以及无限自动机、下推机、图灵机及相应实例。利用 VC ++6.0 实现了给定文法所产生的句子、确定的有限自动机、不确定的有限自动机、带空转移的有限自动机、带输出的有限状态自动机、下推自动机、确定的下推自动机、基本的图灵机、具有计算功能的图灵机、状态存储符号的图灵机、多道图灵机、具有子程序的图灵机。

本书的特点是利用计算机实现给定文法所产生的句子、利用计算机模拟一些自动机，书中含有编程思路及过程，并附有相应 VC ++ 源程序代码，本书可作为高等学校计算机科学与技术专业的本科生和研究生的辅助教材，也可作为相关专业工程技术人员的参考书。

图书在版编目(CIP)数据

形式语言与自动机及程序设计/陆玲，周书民著.
—哈尔滨：哈尔滨工程大学出版社，2014.2
ISBN 978-7-5661-0753-4

Ⅰ.①形… Ⅱ.①陆… ②周… Ⅲ.①形式语言-高等学校-教材②自动机理论-高等学校-教材③程序设计-高等学校-教材 Ⅳ.①TP301②TP311

中国版本图书馆 CIP 数据核字(2014)第 029283 号

出版发行 哈尔滨工程大学出版社
社　　址 哈尔滨市南岗区东大直街 124 号
邮政编码 150001
发行电话 0451-82519328
传　　真 0451-82519699
经　　销 新华书店
印　　刷 肇东市一兴印刷有限公司
开　　本 787mm×960mm 1/16
印　　张 10.75
字　　数 220 千字
版　　次 2014 年 2 月第 1 版
印　　次 2014 年 2 月第 1 次印刷
定　　价 21.00 元
http://press.hrbeu.edu.cn
E-mail:heupress@hrbeu.edu.cn

前　言

形式语言与自动机理论属于计算机科学基本理论的内容,在计算机科学的许多领域起着重要的促进作用。近年来,各领域都在应用形式语言与自动机的理论和方法,使得形式语言与自动机越来越受到多方面的重视,并被逐渐推广和应用。

形式语言与自动机的理论性较强,难于理解,本书在参考相关文献的基础上,利用VC ++6.0编程实现了文法产生句子及部分自动机的模拟功能,对于理解形式语言与自动机有较大的帮助。

本书主要介绍了乔姆斯文法体系的短语结构文法、上下文有关文法、上下文无关文法和正则文法四类文法,设计了给定文法产生句子的演示程序。在自动机方面,详细介绍了有限自动机、下推自动机和图灵机的工作过程、具体实例及程序设计。有限自动机主要包括确定的有限自动机、不确定的有限自动机、带空转移的有限自动机、带输出的有限状态自动机;下推自动机包括不确定的下推自动机和确定的下推自动机;图灵机包括基本的图灵机、具有计算功能的图灵机、状态存储符号的图灵机、多道图灵机、具有子程序的图灵机。

本书力求体现以下特点:

(1)通俗易懂　对于每类文法及各类自动机,都有相应实例并详细描述;

(2)重点突出　侧重介绍由文法产生语言及自动机识别语言的基本思路及方法;

(3)程序设计　使用 VC ++6.0MFC 实现文法产生语言及自动机识别语言等模拟程序,并附有程序设计思路及源程序代码。

本书在编写和出版过程中,得到了东华理工大学重点学科的资助,同时得到哈尔滨工程大学出版社编辑部的大力支持和帮助,在此表示衷心的感谢!

同时,本书在撰写过程中,参考或引用了国内外一些专家学者的论著,在此表示感谢!

由于作者水平有限,难免有不妥和错误之处,恳请读者批评指正。

著　者

2013 年 8 月 27 日

目 录

第 1 章　基础知识简介

1.1 集　合

1.1.1　集合及其表示

一定范围内的、确定的并且彼此可以区分的对象汇集在一起形成的整体叫集合，简称为集。集合的成员称为该集合的元素。

集合描述形式：

(1)列举法　元素个数较少的集合，采取列举法。

如 $S=\{a,b,c,d\}$，表示集合 S 由 a,b,c,d 四个元素组成。

(2)命题法　元素个数较多的集合，特别是由无穷多个元素组成的集合，可用集合描述模式 $\{x \mid P(x)\}$。x 表示该集合中的任一元素，$P(x)$ 是一个谓词，它对 x 进行限定，表示由满足 $P(x)$ 的一切 x 构成的集合。

例如，$\{n \mid n$ 是偶数$\}$ 定义了一个由所有偶数组成的集合，$\{x \mid 0 \leqslant x \leqslant 1\}$ 表示在 $[0,1]$ 区间上所有实数的集合。

集合的分类：

(1)有限集(又称有穷集)　集合中的元素是有限个。

例如，全人类组成的集合是一个有限集。

(2)无限集(又称无穷集)　如果集合不是有限集合，则称为无限集合。

例如，全体自然数组成的集合是一个无限集。

可数集：整数、有理数。

不可数集：实数。

(3)空集　不包含任何元素的集合，用 ϕ 表示。

若 a 是集合 A 的一个元素，则记作 $a \in A$(a 属于 A)；否则记作 $a \notin A$。

1.1.2　集合之间的关系

1. 子集

如果集合 A 中的每个元素都是集合 B 的元素，称 A 是 B 的子集，B 是 A 的包集。记作 $A \subseteq B$(集合 A 包含在集合 B 中)或 $B \supseteq A$(集合 B 包含集合 A)。

如果 $A\subseteq B$，且$\exists x\in B$，但 $x\notin A$，称 A 是 B 的真子集，记作 $A\subset B$。

注意：(1)任何集合都包含它本身；

(2)任何集合都包含空集。

2. 集合相等

如果集合 A，B 含有的元素完全相同，则称集合 A 与集合 B 相等，记作 $A=B$。

1.1.3　集合的运算

并：A 与 B 的并是一个集合，该集合中的元素要么是 A 的元素，要么是 B 的元素，记作 $A\cup B$。

$$A\cup B=\{a\mid a\in A \text{ 或 } a\in B\}$$

交：A 与 B 中都有的元素合在一起的集合，记作 $A\cap B$。

$$A\cap B=\{a\mid a\in A \text{ 且 } a\in B\}$$

差：属于 A 与不属于 B 中的元素组成的集合，记作 $A-B$。

$$A-B=\{a\mid a\in A \text{ 且 } a\notin B\}$$

补集：A 是论域 U 上的一个集合，A 补集是由 U 中的、不在 A 中的所有元素组成的集合，记作

$$\overline{A}=U-A$$

$$\overline{\phi}=U\qquad \overline{U}=\phi$$

图 1－1 显示了以上四种集合运算的示意图。

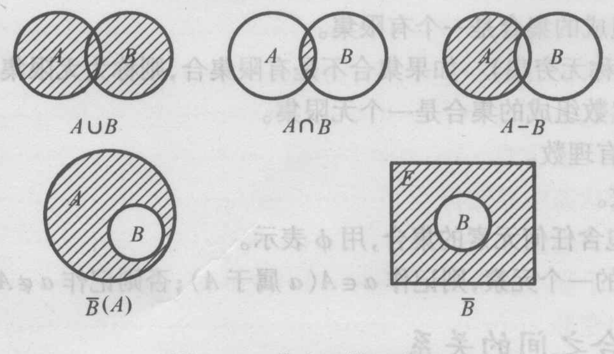

图 1－1　集合运算示意图

1.2 语言及其表示

1.2.1 字母表

字母表是一个非空有穷集合，字母表中的元素称为该字母表的一个字母，又叫做符号、或者字符。字母表一般用Σ表示。

例如：

$\Sigma=\{a,b,c,\cdots,z\}$ 表示26个小写字母组成的字母表；

$\Sigma=\{0,1\}$ 表示二进制数字母表；

$\Sigma=\{==,!=,\geqslant,\leqslant,>,<\}$ 表示逻辑运算符字母表。

1. 字母表的乘积

$$\Sigma_1\Sigma_2=\{ab \mid a\in\Sigma_1,b\in\Sigma_2\}$$

例如：

$\{0,1\}\{0,1\}=\{00,01,10,00\}$

$\{0,1\}\{a,b,c,d\}=\{0a,0b,0c,0d,1a,1b,1c,1d\}$

$\{a,b,c,d\}\{0,1\}=\{a0,a1,b0,b1,c0,c1,d0,d1\}$

2. 字母表Σ的n次幂

$$\Sigma^0=\{\varepsilon\}$$

$$\Sigma^n=\Sigma^{n-1}\Sigma$$

ε是由Σ中的0个字符组成的。

3. Σ的正闭包

$$\Sigma^+=\Sigma\cup\Sigma^2\cup\Sigma^3\cup\Sigma^4\cup\cdots$$

4. Σ的克林闭包

$$\Sigma^*=\Sigma^0\cup\Sigma^+=\Sigma^0\cup\Sigma\cup\Sigma^2\cup\Sigma^3\cup\cdots$$

例如：

$\{0,1\}^+=\{0,1,00,01,11,000,001,010,011,100,\cdots\}$

$\{0,1\}^*=\{\varepsilon,0,1,00,01,11,000,001,010,011,100,\cdots\}$

$\{a,b,c,d\}^+=\{a,b,c,d,aa,ab,ac,ad,ba,bb,bc,bd,\cdots,aaa,aab,aac,aad,aba,abb,$

$abc,\cdots\}$

$\{a,b,c,d\}^* = \{\varepsilon,a,b,c,d,aa,ab,ac,ad,ba,bb,bc,bd,\cdots,aaa,aab,aac,aad,aba,abb,abc,\cdots\}$

1.2.2 句子

Σ是一个字母表，$\forall x\in\Sigma^*$，x 叫作 Σ 上的一个句子。

(1)句子相等

两个句子它们对应位置上的字符都对应相等。

(2)句子的别称

字(word)、(字符、符号)行(line)、(字符、符号)串(string)。

(3)出现

①$x,y\in\Sigma^*$，$a\in\Sigma$，句子 xay 中的 a 叫作 a 在该句子中的一个出现。

②当 $x=\varepsilon$ 时，a 的这个出现为字符串 xay 的首字符。

③如果 a 的这个出现是字符串 xay 的第 n 个字符，则 y 的首字符的这个出现是字符串 xay 的第 $n+1$ 个字符。

④当 $y=\varepsilon$ 时，a 的这个出现是字符串 xay 的尾字符。

(4)句子的长度

①$\forall x\in\Sigma^*$，句子 x 中字符出现的总个数叫作该句子的长度，记作 $|x|$。

②长度为 0 的字符串叫作空句子，记作 ε。

例如：$|abaabb|=6$

$|bbaa|=4$

$|\varepsilon|=0$

(5)连接

$x,y\in\Sigma^*$，x,y 的连接是由串 x 直接相接串 y 所组成的，记作 xy。

(6)串 x 的 n 次幂

$x^0=\varepsilon$

$x^n=x^{n-1}x$

例如：$x=001$，$y=1101$

$x^0=y^0=\varepsilon$

$x^4=001001001001$

$y^4=1101110111011101$

(7)前缀与后缀

设 $x,y,z,w,v\in\Sigma^*$，且 $x=yz$，$w=yv$。

①y 是 x 的前缀,如果 $z \neq \varepsilon$,则 y 是 x 的真前缀;

②z 是 x 的后缀,如果 $y \neq \varepsilon$,则 z 是 x 的真后缀;

③y 是 x 和 w 的公共前缀。

例如,字母表 $\Sigma = \{a, b\}$ 上的句子 $abaabb$ 的前缀、后缀、真前缀和真后缀如下:

前缀　$\varepsilon, a, ab, aba, abaa, abaab, abaabb$

真前缀　$\varepsilon, a, ab, aba, abaa, abaab$

后缀　$\varepsilon, b, bb, abb, aabb, baabb, abaabb$

真后缀　$\varepsilon, b, bb, abb, aabb, baabb$

(8)子串

$w, x, y, z \in \Sigma^*$,且 $w = xyz$,则称 y 是 w 的子串。

(9)公共子串

$t, u, v, w, x, y, z \in \Sigma^*$,且 $t = uyv, w = xyz$,则称 y 是 t 和 w 的公共子串。

1.2.3　语言

1. 语言的概念

对于给定的字母表 Σ,Σ 上字符串(句子)的一个集合称为字母表 Σ 上的语言。

$\forall L \subseteq \Sigma^*$,$L$ 称为字母表 Σ 上的一个语言;$\forall x \in L$,x 叫做 L 的一个句子。

例如:$\{0,1\}$ 上的不同语言

$\{00, 11\}$, $\{0, 1\}$

$\{0, 1, 00, 11\}$, $\{0, 1, 00, 11, 01, 10\}$

$\{00, 11\}^*$, $\{01, 10\}^*$, $\{00, 01, 10, 11\}^*$

2. 语言的连接

$L_1 \subseteq \Sigma_1^*$, $L_2 \subseteq \Sigma_2^*$,语言 L_1 与 L_2 的连接是一个语言,该语言定义为:$L_1L_2 = \{xy \mid x \in L_1, y \in L_2\}$ 是字母表 $\Sigma_1 \cup \Sigma_2$ 上的语言。

例如:$\Sigma_1 = \{a, b\}$, $\Sigma_2 = \{0, 1\}$, $L_1 = \{ab, ba, bb\}$, $L_2 = \{00, 11\}$,则 $L_1L_2 = \{ab00, ab11, ba00, ba11, bb00, bb11\}$。

3. 幂

$\forall L \in \Sigma^*$,L 的 n 次幂是一个语言,该语言定义为:

(1)当 $n = 0$ 时,$L^n = \{\varepsilon\}$;

(2)当 $n \geqslant 1$ 时,$L^n = L^{n-1}L$ 。

4. 正闭包

$L^{+} = L \cup L^{2} \cup L^{3} \cup L^{4} \cup \cdots$

5. 克林闭包

$L^{*} = L^{0} \cup L \cup L^{2} \cup L^{3} \cup L^{4} \cup \cdots$

1.3 VC ++6.0MFC 编程简介

本节介绍在 VC ++6.0 中使用 MFC 方式编写基于对话框的形式语言与自动机的部分程序示例。具体步骤如下：

(1)新建文件

进入 Visual C ++6.0 集成开发环境后，选择“文件 | 新建”菜单，如图 1-2 所示。

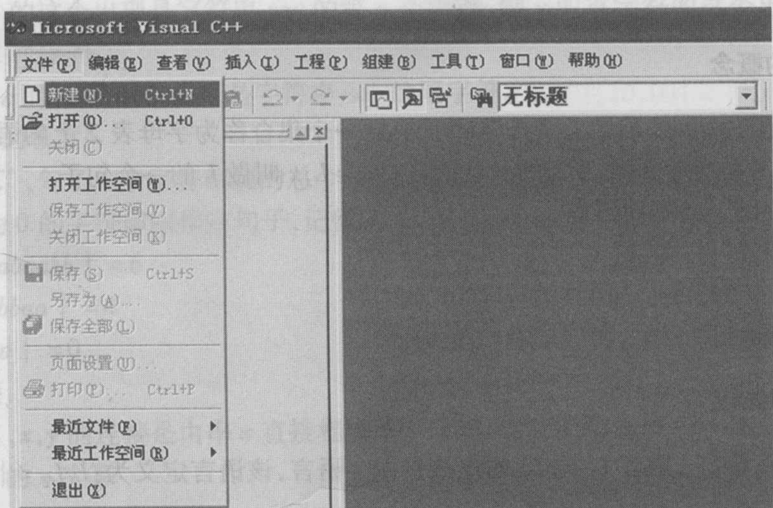

图 1-2 新建文件

(2)新建 MFC AppWizard[exe]工程文件

弹出“新建”对话框后，单击“工程”标签，打开其选项卡，在其左边的列表框中选择 MFC AppWizard(exe)工程类型，在“工程名称”文本框输入工程名(如 grammer)，在“位置”文本框输入工程路径。如果是第一个工程文件，则必须创建一个新的工作区，选择“创建新的工作空间”，如图 1-3 所示。

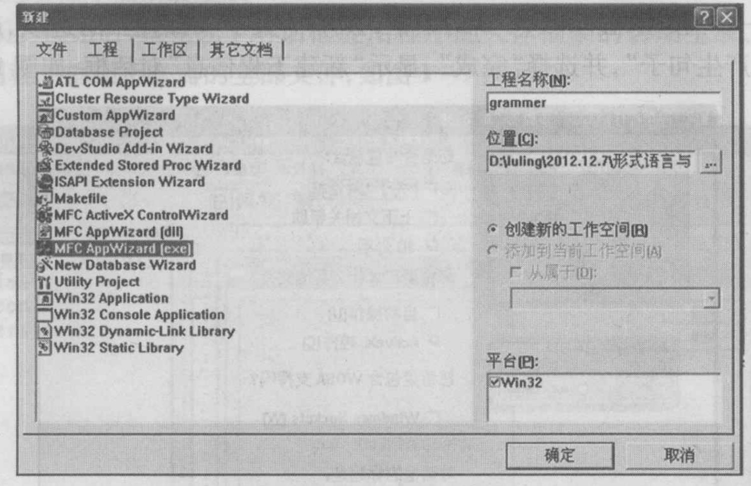

图1-3　新建工程

(3)建立基本对话框应用程序

单击“确定”按钮后,系统显示“MFC 应用程序向导 - 步骤 1”对话框,选择“基本对话框”选项,如图1-4所示。

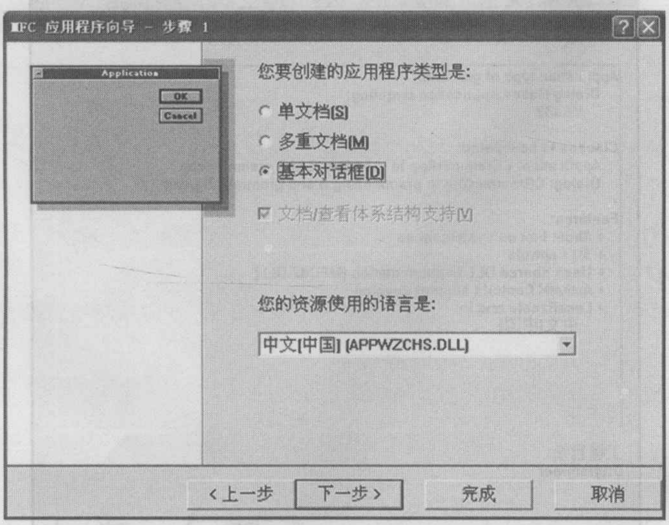

图1-4　MFC 选择“基本对话框”应用程序

(4)选择对话框包含的内容

单击“下一步”按钮,显示图 1-5 对话框,去掉“‘关于’对话框”的选项,改变对话框的标题,如“文法产生句子”,并选择“完成”,显示“新建工程信息”对话框,如图 1-6 所示。

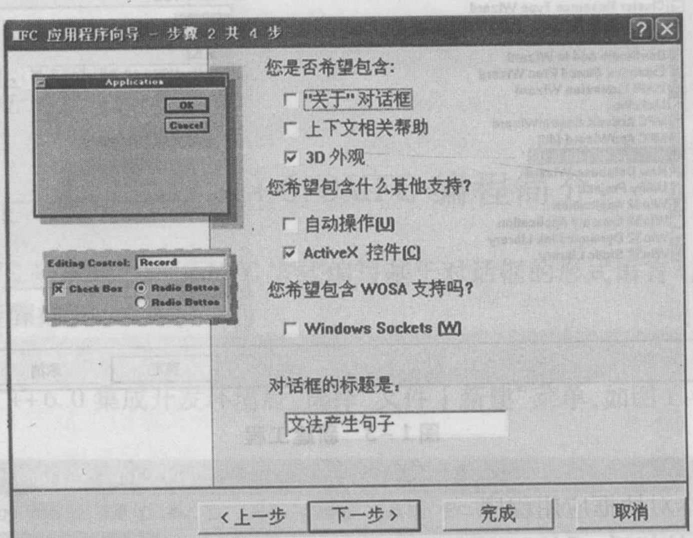

图 1-5　选择对话框包含的内容

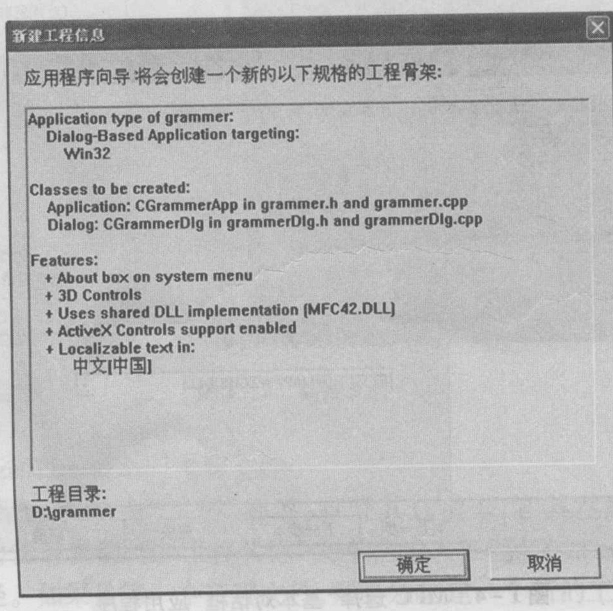

图 1-6　新建工程相关信息

(5)完成应用程序框架的生成

单击“确定”按钮后，就完成了对话框应用程序的大体框架的自动生成，在指定的目录下生成了应用程序框架所必需的全部文件，如图 1-7 所示。

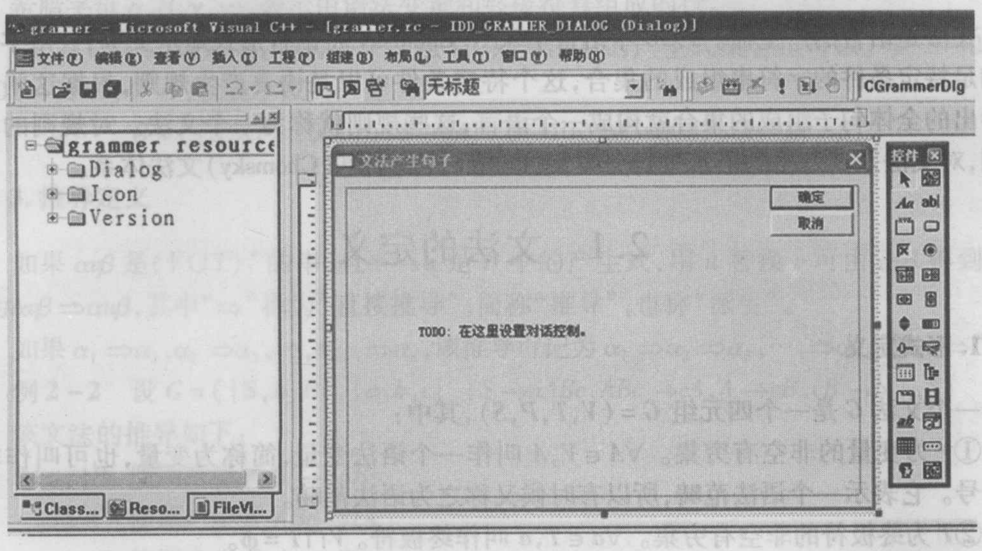

图 1-7　新建工程的对话框

第2章 文 法

文法是语言的产生器。1956 年语言学家 N. Chomsky 将语言形式地定义为,一个字母表上满足特定条件的字符串组成的集合,这个特定条件可以是语言产生规则,根据这些规则推导出的全体句子组成的集合就构成一个语言,这些规则就称为一个文法。对规则的不同限制,对文法及其产生的语言进行分类,就形成了乔姆斯基(Chomsky)文法体系。

2.1 文法的定义

1. 形式定义

一个文法 G 是一个四元组 $G=(V,T,P,S)$,其中:

①V 为变量的非空有穷集。$\forall A\in V$,A 叫作一个语法变量,简称为变量,也可叫作非终极符号。它表示一个语法范畴,所以有时候又称之为语法范畴。

②T 为终极符的非空有穷集。$\forall a\in T$,a 叫作终极符。$V\cap T=\phi$。

③$S\in V$,为文法 G 的开始符号。

④P 为产生式的非空有穷集合。P 中的元素均具有形式 $\alpha\to\beta$,被称为产生式,$\alpha\in(V\cup T)^{+}$,且 α 中至少有一个 V 中元素出现,$\beta\in(V\cup T)^{*}$。α 称为产生式 $\alpha\to\beta$ 的左部,β 称为产生式 $\alpha\to\beta$ 的右部。

例 2-1 以下三个四元组都是文法。

$(\{A\},\{0,1\},\{A\to 01,A\to 0A1\alpha\beta,A\to 1A0\},A)$。

$(\{A,B\},\{0,1\},\{A\to 01,A\to 0A1,A\to 1A0,B\to AB,B\to 0\},A)$。

$(\{S,A,B,C,D\},\{a,b,c,d,\#\},\{S\to ABCD,S\to abc\#,A\to aaA,AB\to aabbB,BC\to bbccC,cC\to cccC,CD\to ccd\#,CD\to d\#,CD\to \#d\},S)$。

2. 一般约定

(1)对一组有相同左部的产生式

$$\alpha\to\beta_1,\alpha\to\beta_2,\cdots,\alpha\to\beta_n$$

可以简单地记为

$$\alpha\to\beta_1\mid\beta_2\mid\cdots\mid\beta_n$$

(2)使用符号

英文字母表较为前面的大写字母,如 $A,B,C,\cdots$表示语法变量;

英文字母表较为前面的小写字母，如 $a,b,c,\cdots$ 表示终极符号；

英文字母表较为后面的大写字母，如 $X,Y,Z,\cdots$ 表示该符号是语法变量或者终极符号；

英文字母表较为后面的小写字母，如 $x,y,z,\cdots$ 表示由终极符号组成的行；

希腊字母 $\alpha,\beta,\gamma,\cdots$，表示由语法变量和终极符号组成的行。

(3) 在以上的约定下，当写一个文法时，只写出它的产生式集合也可以。如 $S \to 0A1 \mid 10, 0A \to 00A1, A \to 1$，就表示该文法是

$$G = (\{S,A\},\ \{0,1\},\ \{S \to 0A1,\ S \to 10,\ 0A \to 00A1,\ A \to 1\},\ S)$$

3. 推导定义

如果 $\alpha v\beta$ 是 $(V \cup T)^*$ 的串，且 $v \to u$ 是 P 中的产生式，用 u 替换 v 可由 $\alpha v\beta$ 得到 $\alpha u\beta$，记为 $\alpha v\beta \Rightarrow \alpha u\beta$，其中"$\Rightarrow$"称为"直接推导"，简称"推导"，也称"派生"。

如果 $\alpha_1 \Rightarrow \alpha_2, \alpha_2 \Rightarrow \alpha_3, \cdots, \alpha_{n-1} \Rightarrow \alpha_n$，该推导可记为 $\alpha_1 \Rightarrow \alpha_2 \Rightarrow \alpha_3, \cdots, \Rightarrow \alpha_n$，或 $\alpha_1 \Rightarrow^* \alpha_n$。

例 2-2　设 $G = (\{S,A,B\}, \{a,b,c\}, \{S \to aABc, ABc \to cA, A \to cB, cB \to b\}, S)$

该文法的推导如下：

$S \Rightarrow aABc$ 使用产生式 $S \to aABc$

$\Rightarrow acA$　使用产生式 $ABc \to cA$

$\Rightarrow accB$ 使用产生式 $A \to cB$

$\Rightarrow acb$　使用产生式 $cB \to b$

所以 $S \Rightarrow^* acb$。

4. 文法产生的语言

由文法 $G = (V,T,P,S)$ 生成的语言 $L(G)$ 是从该文法的开始符号进行推导的可推导终极串集合：

$$L(G) = \{w \mid w \in T^* \text{ 且 } S \Rightarrow^* w\}$$

例 2-3　设文法 G：

①$S \to ACaB$；②$Ca \to aaC$；③$CB \to DB$；④$CB \to E$；⑤$aD \to Da$；⑥$AD \to AC$；⑦$aE \to Ea$；⑧$AE \to \varepsilon$

该文法的一个典型推导如下（约定"$\Rightarrow^{①}$"表示用①产生式推导）：

$S \Rightarrow^{①} ACaB \Rightarrow^{②} AaaCB \Rightarrow^{④} AaaE \Rightarrow^{⑦} AaEa \Rightarrow^{⑦} AEaa \Rightarrow^{⑧} aa$

该文法的另一个典型推导如下（约定"$\Rightarrow^{①\times n}$"表示用 n 次①产生式推导）：

$S \Rightarrow^{①} ACaB \Rightarrow^{②} AaaCB \Rightarrow^{③} AaaDB \Rightarrow^{⑤\times 2} ADaaB \Rightarrow^{⑥} ACaaB$

$\Rightarrow^{②\times 2} AaaaaCB \Rightarrow^{④} AaaaaE \Rightarrow^{⑦\times 4} AEaaaa \Rightarrow^{⑧} aaaa$

可以看出该文法生成的语言为：

$$L(G)=\{a^{2^n} \mid n\geqslant 1\}$$

如果$\forall w \in L(G)$，w称为G产生的一个句子。句子w是从S开始，在G中可以推导出来的终极符号行，它不含语法变量。

对于$\forall \alpha \in (V \cup T)^*$，如果$S \Rightarrow^* \alpha$，则称$\alpha$是$G$产生的一个句型。句型$\alpha$是从$S$开始，在$G$中可以推导出来的符号行，它可能含有语法变量。

2.2 文法的乔姆斯基分类

对于文法$G=(V,T,P,S)$按以下方法分为四类：

(1)若P中的产生式不加另外的限制，则G称为0型文法，或短语结构文法(PSG)。

$L(G)$叫做0型语言，也可以叫做短语结构语言(PSL)。例2-2与例2-3都是0型文法，生成的语言都是0型语言。

(2)若P中每个产生式$\alpha \to \beta$都满足条件$|\alpha| \leqslant |\beta|$，则$G$称为1型文法，或上下文有关文法(CSG)。$L(G)$叫做1型语言或者上下文有关语言(CSL)。

(3)若P中每个产生式都具有如下形式：

$$A \to \beta, \beta \in (V \cup T)^*, A \in V$$

则称G为2型文法，或上下文无关文法(CFG)。

$L(G)$叫做2型语言或者上下文无关语言(CFL)。

(4)若P中每个产生式都具有如下形式：

$$A \to a \text{ 或 } A \to aB, a \in T \cup \{\varepsilon\}, A, B \in V$$

则称G为3型文法，或正则文法(RG)或正规文法。

$L(G)$叫做3型语言，也可称为正则语言或者正规语言(RL)。

例2-4 设文法G：

①$S \to aSAc$；②$S \to abc$；③$cA \to Ac$；④$bA \to bb$

该文法的产生式右边的长度大于等于左边的长度，因此此文法为1型文法，该文法的其中一个推导过程为

$$S \Rightarrow^{①} aSAc \Rightarrow^{①} aaSAcAc \Rightarrow^{③} aaSAAcc \Rightarrow^{②} aaabcAAcc \Rightarrow^{③} aaabAcAcc$$
$$\Rightarrow^{③} aaabAAccc \Rightarrow^{④} aaabbAccc \Rightarrow^{④} aaabbbccc$$

生成的语言为

$$L(G)=\{a^n b^n c^n \mid n \geqslant 1\}$$

是一个1型语言。

例2-5 设文法G：

①$S \to AB$；②$A \to aAb$；③$A \to ab$；④$B \to cB$；⑤$B \to c$

该文法为2型文法或上下文无关文法，其中的一个推导过程为