第二章

字母表

定义

- 字母表：就是符号表。通常用`Σ`表示。是个集合。最常见的字母表是二元字母表{0,1}。
- 我们写(Σ^*)来指示在字母表(Σ)上的所有**任意长**有限字符串（包括空串）的集合
- 我们写(Σ^+)来指示在字母表(Σ)上的所有**长大于等于一**有限字符串的集合
- 我们写(Σ^∞)(偶尔(Σ^N)或(Σ^ω))来指示在字母表上的所有无限序列的集合。

> 这里*表示Kleene 星号，或称 Kleene（克林） 闭包（自反传递闭包），在数学上是一种适用于字符串或符号及字元的集合的一元运算。当 Kleene 星号被应用在一个集合V时，写法是V*。它被广泛用于正则表达式。

运算

- 字母表的乘积：前者的元素拼接后者的元素的集合
- 字母表的克林闭包的元素叫字母表Σ上的 字(word) 行(line) 串(string)
- 空串以`ε`或`λ`來標記
- 字符串的 连接===并置

形式语言

定义

- 形式语言是一个字母表上的某些有限长字符串的集合。
- 两语言（分别是Σ1和Σ2上的）的乘积仍是语言（Σ1∪Σ2上的），做法是两个语言的元素连接。

语言的运算

- 字符串集合的交并补等运算。
- 连接运算：L1L2 = { xy | x 属于L1并且 y 属于L2 }。
- 幂运算：Ln = L … L （共 n 个 L 连接在一起），L0 = {ε}。
- 闭包运算：$L* = L0 \cup L1 \cup \ldots \cup Ln \cup \ldots$
- 右商运算：L1/L2 = {x | 存在 y 属于L2使得 xy 属于L1}。
- 设 S ⊆ Σ* 是一个语言，S 的补语言定义为集合 {ω | ω ∈ Σ* 且 ω ∉ S}

文法

概念

符号	意思
→	可以是
↑	幂
G	形式文法
T	终极符非空有穷集
P	产生式非空有穷集
S	初始符号
V	变量

- 语言有很多表示方法：
  - 形式文法
  - 正则表达式
  - 枚举法
  - 自动机
- **形式文法**是**描述语言的一种方法**。形式文法描述形式语言的基本想法是，从一个特殊的初始符号出发，不断的应用一些**产生式规则**，从而生成出一个字串的集合。其中产生式规则指定了某些符号组合如何被另外一些符号组合替换。
- **形式文法**G是一个四元组，V是变量非空有穷集合，T是终极符非空有穷集合，P是产生式规则非空有穷集合，S是开始符号（不是集合）。
- **终极符号**（终结符）： 对于某一文法而言，在产生式规则下无法被转化成其他符号的符号。是语言的句子（最终转化结果）中出现的符号。
- **变量**（非终结符、非终极符号、语法范畴）： 与终极符号相反。
- 一般地，我们用
  - 前排大写字母 如A,B,C... 表示语法变量
  - 前排小写字母 如a,b,c... 表示终极符号
  - 后排大写字母 如X,Y,Z... 表示语法变量或者终极符号
  - 后排小写字母 如x,y,z... 表示句子（终极符号组成的行）
  - 希腊字母　 如α,β,γ... 表示句型（混合符号行）
- 句子一定是句型，反之不然

文法的构造

约定

- 若两个文法G_1和G_2产生的形式语言相同，即$L(G_1)=L(G_2)$ 它们就等价。
- 对于一个文法的表述，我们可以列出所有的产生式，然后第一个产生式的左部就是开始符号

Σ闭包型文法的构造

- 我们要构造形如{ $w|w\in\Sigma^+$}的语言
- 只需要对Σ中的每个符号a安排产生式 $S\to a|aS$ 就可以了

回文的文法构造

- 我们要构造形如{ $ww^T|w\in\Sigma^+$}的语言 T表示逆序
- 只需要对Σ中的每个符号a安排产生式 $S\to aa|aSa$ 就可以了 这样会产生偶数的长度回文语言
- 如果对Σ中的每个符号a安排产生式 $S\to a|aSa$ 就会产生奇数的长度回文语言

归纳型文法总结

语言L	基础	归纳
$\{x^n\|n\geq 1\}$	$S\to x$	$S\to xS$
$\{x^n\|n\geq 0\}$	$S\to \varepsilon$	$S\to xS$
$\{x^ny^n\|n\geq 1\}$	$S\to xy$	$S\to xSy$
$\{x^ny^n\|n\geq 0\}$	$S\to \varepsilon$	$S\to xSy$
$\sum^+$	$S\to A$, $A\to a$, $a\in\sum$	S → AS
$\{ww^T\|\ w\in\sum^+\}$	$S\to aa$, $a\in\sum$	$S\to aSa$, $a\in\sum$
表达式	$E\to c$,　$E\to id$	$E\to E+E$, $E\to E-E$...

乔姆斯基体系

乔姆斯基体系是计算机科学中刻画形式文法表达能力的一个分类谱系，包括四个层次：

0-型文法（无限制文法或短语结构文法）

包括所有的文法。该类型的文法能够产生**所有可被图灵机识别的语言**。可被图灵机识别的语言是指能够使图灵机停机的字串，这类语言又被称为**递归可枚举语言**。注意递归可枚举语言与递归语言的区别，后者是前者的一个真子集，是能够被一个总停机的图灵机判定的语言。

1-型文法（上下文相关文法）

- 生成**上下文相关语言**。
- 这种文法的产生式规则取如 α*A*β -> αγβ 一样的形式。这里的*A* 是非终结符号，而 α, β 和 γ 是包含非终结符号与终结符号的字串；α, β 可以是空串，但 γ 必须不能是空串； 也就是说 **右边的长度大于等于左边**
- 这种文法也可以包含规则 S->ε（特殊情况），但**此时文法的任何产生式规则都不能在右侧包含 S**
- 这种文法规定的语言可以被线性有界非确定图灵机接受。

2-型文法（上下文无关文法）

- 生成**上下文无关语言**。
- 这种文法的产生式规则取如 *A* -> γ 一样的形式。这里**左侧的A 是非终结符号**，γ 是包含非终结符号与终结符号的字串。
- 这种文法规定的语言可以被非确定下推自动机接受。
- 上下文无关语言为大多数程序设计语言的语法提供了理论基础。

3-型文法（正规文法）

- 生成**正规语言**。
- 这种文法要求**产生式的左侧只能包含一个非终结符号**
- 产生式的**右侧只能是空串、一个终结符号或者一个非终结符号后随一个终结符号**；
- 如果所有产生式的右侧都不含初始符号 S ，规则 S -> ε 也允许出现。
- 这种文法规定的语言可以被有限状态自动机接受，也可以通过正则表达式a*bc*来获得。
- 正规语言通常用来定义检索模式或者程序设计语言中的词法结构。

正规文法

正规文法有多种等价的定义，我们可以用*左线性文法*或者*右线性文法*来等价地定义正规文法：

- *左线性文法*要求产生式的左侧只能包含一个非终结符号，产生式的右侧只能是空串、一个终结符号或者一个非终结符号后随一个终结符号。
- *右线性文法*要求产生式的左侧只能包含一个非终结符号，产生式的右侧只能是空串、一个终结符号或者一个终结符号后随一个非终结符号。

下面给出一个正规语言的例子，语言{ $a^n b^m | m, n > 0$ }是一个正规语言。文法*G3*包括*N*={S,A,B}, Σ={a,b}, S是起始符号，产生式规则有：

1. S -> aA
2. A -> aA
3. A -> bB
4. B -> bB
5. B -> ε