



# 编译原理

## 第二章 高级语言及其语法描述

## 第二章 高级语言及其语法描述

- 程序语言的定义
- 高级语言的一般特性
- 程序语言的语法描述

# 第二章 高级语言及其语法描述

- 程序语言的定义
  - 语法
  - 语义
  - 程序语言的功能
- 高级语言的一般特性
  - 高级语言的分类
  - 程序结构
  - 数据结构与操作
  - 语句与控制结构

## 第二章 高级语言及其语法描述

- 程序语言的定义
- 高级语言的一般特性
- 程序语言的语法描述

## 2.3 程序语言的语法描述

### ■ 几个概念

- 考虑一个有穷 **字母表**  $\Sigma$  字符集
- 其中每一个元素称为一个 **字符**
- $\Sigma$  上的 **字** (也叫 **字符串**) 是指由  $\Sigma$  中的字符所构成的一个有穷序列
- 不包含任何字符的序列称为 **空字**, 记为  $\varepsilon$
- 用  $\Sigma^*$  表示  $\Sigma$  上的所有 **字的全体**, 包含空字  $\varepsilon$
- 例如: 设  $\Sigma = \{a, b\}$ , 则
$$\Sigma^* = \{\varepsilon, a, b, aa, ab, ba, bb, aaa, \dots\}$$

- $\Sigma^*$  的子集  $U$  和  $V$  的连接（积）定义为

$$UV = \{ \alpha\beta \mid \alpha \in U \ \& \ \beta \in V \}$$

- 设：

$$U = \{ a, aa \}$$

$$V = \{ b, bb \}$$

- 那么：

$$UV = \{ ab, abb, aab, aabb \}$$

- $\Sigma^*$  的子集  $U$  和  $V$  的**连接**（**积**）定义为

$$UV = \{ \alpha\beta \mid \alpha \in U \ \& \ \beta \in V \}$$

- $V$  自身的  $n$  次积记为

$$V^n = \underbrace{V \ V \ \dots \ V}_{n \uparrow}$$

- 规定  $V^0 = \{\varepsilon\}$

- 令

$$V^* = V^0 \cup V^1 \cup V^2 \cup V^3 \cup \dots$$

称  $V^*$  是  $V$  的**闭包**

- 记  $V^+ = V V^*$ ，称  $V^+$  是  $V$  的**正规闭包**

■ 设：

$$U = \{ a, aa \}$$

■ 那么：

$$U^* = \{ \varepsilon, a, aa, aaa, aaaa, \dots \}$$

$$U^+ = \{ a, aa, aaa, aaaa, \dots \}$$



# 上下文无关文法

- **文法**：描述语言的语法结构的形式规则
- He gave me a book.

< 句子 > → < 主语 > < 谓语 > < 间接宾语 > < 直接宾语 >

< 主语 > → < 代词 >

< 谓语 > → < 动词 >

< 间接宾语 > → < 代词 >

< 直接宾语 > → < 冠词 > < 名词 >

< 代词 > → He

< 代词 > → me

< 名词 > → book

< 冠词 > → a

< 动词 > → gave

< 句子 > → < 主语 > < 谓语 > < 间接宾语 > < 直接宾语 >

< 主语 > → < 代词 >

< 谓语 > → < 动词 >

< 间接宾语 > → < 代词 >

< 直接宾语 > → < 冠词 >

< 代词 > → He

< 代词 > → me

< 名词 > → book

< 冠词 > → a

< 动词 > → gave

< 句子 >

⇒ < 主语 > < 谓语 > < 间接宾语 > < 直接宾语 >

⇒ < 代词 > < 谓语 > < 间接宾语 > < 直接宾语 >

⇒ He < 谓语 > < 间接宾语 > < 直接宾语 >

⇒ He < 动词 > < 间接宾语 > < 直接宾语 >

⇒ He gave < 间接宾语 > < 直接宾语 >

⇒ He gave < 代词 > < 直接宾语 >

⇒ He gave me < 直接宾语 >

⇒ He gave me < 冠词 > < 名词 >

⇒ He gave me a < 名词 >

⇒ He gave me a book

# 上下文无关文法

< 句子 >  $\rightarrow$  < 主语 > < 谓语 > < 间接宾语 > < 直接宾语 >  
< 主语 >  $\rightarrow$  < 代词 >  
< 谓语 >  $\rightarrow$  < 动词 >  
< 间接宾语 >  $\rightarrow$  < 代词 >  
< 直接宾语 >  $\rightarrow$  < 冠词 > < 名词 >  
< 代词 >  $\rightarrow$  He  
< 代词 >  $\rightarrow$  me  
< 名词 >  $\rightarrow$  book  
< 冠词 >  $\rightarrow$  a  
< 动词 >  $\rightarrow$  gave

- 一个上下文无关文法  $G$  是一个四元式

$G=(V_T, V_N, S, P)$ ，其中

- $V_T$ ：终结符集合（非空）
- $V_N$ ：非终结符集合（非空），且  $V_T \cap V_N = \emptyset$
- $S$ ：文法的开始符号， $S \in V_N$
- $P$ ：产生式集合（有限），每个产生式形式为
  - $P \rightarrow \alpha$ ， $P \in V_N$ ， $\alpha \in (V_T \cup V_N)^*$
- 开始符  $S$  至少必须在某个产生式的左部出现一次

# 上下文无关文法

- 例，定义只含 + , \* 的算术表达式的文法

$G = \langle \{i, +, *, (, )\}, \{E\}, E, P \rangle$ , 其中,  $P$  由下列产生式组成:

$E \rightarrow i$

$E \rightarrow E + E$

$E \rightarrow E * E$

$E \rightarrow (E)$

# 巴科斯范式 (BNF)

- “ $\rightarrow$ ” 也可以用 “ $::=$ ” 表示，这种表示称为巴科斯范式 (BNF)



For profound, influential, and lasting contributions to the design of practical high-level programming systems, notably through his work on FORTRAN, and for seminal publication of formal procedures for the specification of programming languages.

# 巴科斯范式 (BNF)



John W. Backus

首次在 ALGOL 58 中使用这种记号系统描述语法



Peter Naur

在 ALGOL 60 中发展并简化  
命名 Backus Normal Form



Donald E. Knuth

主张称为巴斯科 - 诺尔范式 (Backus – Naur Form)  
认为它不算是一种正规形式 (Normal Form)

■ 例，定义只含 + , \* 的算术表达式的文法

$G = \langle \{i, +, *, (, )\}, \{E\}, E, P \rangle$   
，其中， $P$  由下列产生式组成：

$E \rightarrow i$

$E \rightarrow E + E$

$E \rightarrow E * E$

$E \rightarrow (E)$

□ 表示一个文法时，通常只给出开始符号和产生式，如上例，可表示为：

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

- 定义：称  $\alpha A\beta$  **直接推出**  $\alpha\gamma\beta$ ，即

$$\alpha A\beta \Rightarrow \alpha\gamma\beta$$

仅当  $A \rightarrow \gamma$  是一个产生式，

且  $\alpha, \beta \in$

- 如果  $\alpha_1 \Rightarrow \alpha_n$ ，  
列是从  $\alpha_1$  到  
从  $\alpha_1$  到  $\alpha_n$  的  
。

- 对文法  $G(E)$

< 句子 >

$\Rightarrow$  < 主语 > < 谓语 > < 间接宾语 > < 直接宾语 >

$\Rightarrow$  < 代词 > < 谓语 > < 间接宾语 > < 直接宾语 >

$\Rightarrow$  He < 谓语 > < 间接宾语 > < 直接宾语 >

$\Rightarrow$  He < 动词 > < 间接宾语 > < 直接宾语 >

$\Rightarrow$  He gave < 间接宾语 > < 直接宾语 >

$\Rightarrow$  He gave < 代词 > < 直接宾语 >

$\Rightarrow$  He gave me < 直接宾语 >

$\Rightarrow$  He gave me < 冠词 > < 名词 >



■ 通常，用  
或若干步<sub>\*</sub>

用  $\alpha_1 \Rightarrow \alpha_n$  表  
干步，可以

所以：

□ 定义：假定

如果  $S \xRightarrow{*}$

的句型是一个句子。文法  $G$  所产生的句子的全体  
是一个语言，将它记为  $L(G)$ 。

$$L(G) = \{\alpha \mid S \xRightarrow{+} \alpha, \alpha \in V_T^*\}$$

< 句子 >

$\Rightarrow$  < 主语 > < 谓语 > < 间接宾语 > < 直接宾语 >

$\Rightarrow$  < 代词 > < 谓语 > < 间接宾语 > < 直接宾语 >

$\Rightarrow$  He < 谓语 > < 间接宾语 > < 直接宾语 >

$\Rightarrow$  He < 动词 > < 间接宾语 > < 直接宾语 >

$\Rightarrow$  He gave < 间接宾语 > < 直接宾语 >

$\Rightarrow$  He gave < 代词 > < 直接宾语 >

$\Rightarrow$  He gave me < 直接宾语 >

$\Rightarrow$  He gave me < 冠词 > < 名词 >

$\Rightarrow$  He gave me a < 名词 >

$\Rightarrow$  He gave me a book

# 小结

- 上下文无关文法的定义
- 基本概念
  - 推导、句型、句子、语言